Preface

This volume contains the papers presented at CCKS2016: China Conference on Knowledge Graph and Semantic Computing held on September 19-22, 2016 in Beijing.

CCKS is organized by the Technical Committee on Language and Knowledge Computing of CIPS (Chinese Information Processing Society of China). CCKS2016 is merged from two premier relevant forums held previously: the Chinese Knowledge Graph Symposium (KGS), and the Chinese Semantic Web and Web Science Conference (CSWS). KGS was firstly held in Beijing in 2013, and then in Nanjing in 2014, at Yichang 2015. CSWS was firstly held in Beijing in 2006, and has continually been the main forum for research on the Semantic (Web) Technologies in China for nearly ten years. The new conference CCKS brings together researchers from both forums and covers wider fields including the Knowledge Graph, the Semantic Web, Linked Data, NLP, knowledge representation, graph databases etc. It aims to become the top forum on Knowledge Graph and Semantic Technologies for Chinese researchers and practitioners from academia, industry, and government.

The theme of this year is "Semantic, Knowledge and Linked Big Data".

In sumarry, there were 82 submissions. Each submission was reviewed by at least 2, and on the average 2.5, program committee members. The committee decided to accept 21 full papers and 8 short papers. The program also includes 4 invited keynotes, 4 tuorials, 4 shared tasks, 1 panel and 1 industrial forum. This year's talks were given by Prof. Ian Horrocks from Oxford University, Prof. Gerhard Weikum from Max-Planck-Institut für Informatik, Dr. Haixun Wang from Facebook, and Prof. Heyan Huang from Beijing Institute of Technology. The tutorials were given by Dekang Lin from Sigularity.io, Jie Bao from MemeCT, Jeff. Pan from Aberdeen University, Tong Ruan from East China University of Science and Technology, Haixun Wang from Facebook, Zhongyuan Wang from Microsoft Research Asia, Wei Hu and Gong Cheng from Nanjing University.

The hard work and close collabration of a number of people have contributed to the success of this conference. We would like to thank the members of the Organizing Committee and Program Committee for their support; and the authors and participants who are the primary reason for the success of this conference.

Finally, we would like to apprieciate the sponsorships from TRS and Unisound as golden sponsors, Baidu, Fujitsu, and Puhui Finance as silver sponsors.

September 19, 2016
Beijing

Conference General Chairs: Le SUN, Haixun WANG
Program Committee Chairs: Huajun CHEN, Heng JI
Tutorial Chairs: Jiaoyan ZHU, Wei HU
Industry Forum Chairs: Haofen WANG, Jie BAO
Evaluation Chairs: Kang LIU, Zhichun WANG
Poster/Demo Chairs: Yuan NI, Qi ZHANG
Local Chairs: Xianpei HAN, Yiqun LIU
Sponsorship Chairs: Jinguang GU
Publication Chairs: Tieyun QIAN, Tong RUAN
Publicity Chairs: Honghan WU, Xiangwen LIAO

Program Committee

Lidong Bing	CMU
Yixin Cao	Tsinghua University
Huajun Chen	Zhejiang University
Liwei Chen	Peking University
Gong Cheng	Nanjing University
Jingwei Cheng	Northeastern University
Jianfeng Du	Guangdong University of Foreign Studies
Yanking Feng	Peking University
Wu Gang	Northeast University
Tao Ge	Peking University
Saisai Gong	Nanjing University
Shu Guo	Chinese Academy of Sciences
Yu Hong	Suzhou University
Songfang Huang	IBM Research
Heng Ji	RPI
Yanyan Jia	
Juanzi Li	Tsinghua University
Yuan-Fang Li	Monash University
Yankai Lin	Tsinghua
Kang Liu	Chinese Academy of Sciences
Zhiyuan Liu	Tsinghua University
Jie Lu	IBM
Bingfeng Luo	Peking University
Xiaogang Ma	RPI
Gerard De Melo	Tsinghua University
Yao Meng	Fujitsu
Yuan Ni	IBM
Jeff Pan	Aberdeen University
Xian Pei	Chinese Academy of Science
Guilin Qi	Southeast University
Bin Qin	Harbin Institute of Technology
Xipeng Qiu	Fudan University
Yuming Shen	Guangdong University of Foreign Studies
Yuping Shen	Sun Yat-sen University
He Shizhu	Chinese Academy of Sciences
Dezhao Song	Thomson Reuters
Chengjie Sun	Harbin Institute of Technology
Hai Wan	Sun Yat-sen University
Juan Wang	Chinese Academy of Sciences
Junhu Wang	Griffith University
Linlin Wang	Tsinghua University
Xin Wang	Tianjin University
Yafang Wang	Shandong University

Zhe Wang	Griffith University
Zhigang Wang	Tsinghua University
Gang Wu	Northeastern University
Ruobing Xie	Tsinghua Universityhina
Wang Xin	Tianjing University
Kun Xu	Peking University
Ran Yu	L3S
Pingpeng Yuan	Huazhong University of Science and Technology
Fu Zhang	Northeastern University
Heng Zhang	Huazhong University of Science and Technology
Qi Zhang	Fudan University
Xiaowang Zhang	Tianjin University, China
Ziqi Zhang	University of Sheffield
Jun Zhao	China Academy of Science
Ganggao Zhu	Universidad Politécnica de Madrid
Bowei Zou	Soochow University

Table of Contents

Full Papers

Short Papers

Evaluation papers

Chinese Papers

Object Clustering in Linked Data using Centrality

Xiang Zhang[1], Yulian Lv[2], Erjing Lin[1]

[1]School of Computer Science and Engineering, Southeast University, Nanjing, China
{x.zhang, linerjing}@seu.edu.cn
[2]College of Software Engineering (Suzhou), Southeast University, Suzhou, China
lvyulian@seu.edu.cn

Abstract. Large-scale linked data is becoming a challenge to many Semantic Web tasks. While clustering of graphs has been deeply researched in network science and machine learning, not many researches are carried on clustering in linked data. To identify meta-structures in large-scale linked data, the scalability of clustering should be considered. In this paper, we propose a scalable approach of centrality-based clustering, which works on a model of Object Graph derived from RDF graph. Centrality of objects is calculated as indicators for clustering. Both relational and linguistic closeness between objects are considered in clustering to produce coherent clusters.

1 Introduction

The great volume of linked data is becoming a challenge for many Semantic Web tasks. These tasks vary from semantic query [1] to semantic mining [2]. The scale of linked data demands new methods to discover knowledge from the links or linguistics in linked data. A promising approach is to decompose linked data into clusters, which are sets of densely inter-connected objects. The identification of these clusters is of crucial importance as they may help to scale down the problem when exploring linked data, or may help researchers to understand the meta-structure of the linked data.

Clustering approaches have been deeply researched in the modern science of networks and machine learning. While clustering approaches like K-means or spectral clustering are commonly used and effective in small or medium dataset, they can be hardly adapted to the scale of linked data. To the best of our knowledge, clustering or community detection in linked data is still a research area not being deeply explored. There are two major problems facing this area: (1) A near-linear clustering approach is needed to efficiently decompose massive linked data; (2) How to effectively utilize relations and linguistic information of objects, which are both abundant in linked data.

We propose a centrality-based clustering in this paper, which is efficient for clustering large-scale linked data. We introduce Object Graph as the graph model. The closeness between two objects is measured both relationally and linguistically. The notion of Virtual Document is used to measure linguistic closeness between objects. For each object in linked data, a set of graph centralities is assessed and *k* centroids are selected using a distance-maximization strategy. An LPA-based clustering will decompose linked data into *k* clusters.

adfa, p. 1, 2011.

2 Models and Architecture

In this section, we propose Object Graph as the graph model for clustering. A Virtual Document is built for each object in Object Graph to capture its linguistic information. The architecture of our approach is also discussed.

2.1 Object Graph and Virtual Document

RDF model of linked data is multi-mode and multi-dimensional with multiple types of nodes (classes, properties, objects or literals) and multiple types of relations. It is not suitable for object clustering. We propose a single-mode and single-dimensional graph model, called Object Graph, as the graph model for object clustering.

Definition1 (**Object Graph**): Given a Linked Data ℓ, its Object Graph $\mathcal{G}(\ell) = \langle \mathcal{O}, \mathcal{W}, \mathcal{VD}_{\mathrm{n}} \rangle$ is a directed graph. $\mathcal{O}$ is the node set, which comprises all the named objects defined or referred in ℓ; $\mathcal{W}$ is a weighting scheme of edges. Given $o_{\mathrm{i}}, o_{\mathrm{j}} \in \mathcal{O}$, if $\mathcal{W}(o_{\mathrm{i}}, o_{\mathrm{j}}) > 0$, there is a weighted edge from o_{i} to o_{j} in ℓ. $\mathcal{W}(o_{\mathrm{i}}, o_{\mathrm{j}})$ equals to the closeness from o_{i} to o_{j}. $\mathcal{VD}_{\mathrm{n}}$ is a labeling function of $\mathcal{G}(\ell)$. For each $o_{\mathrm{i}} \in \mathcal{O}$, $\mathcal{VD}_n(o_{\mathrm{i}})$ is called *n*-step virtual document of o_{i}, which is a bag of words capturing linguistic information of o_{i} in ℓ.

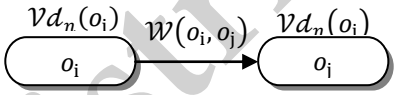


Fig. 1. The Model of Object Graph

Shown in Fig.1, each node in Object Graph represents a named object, and there is an edge from one object to another when 1) there is a direct relation between them in RDF model; 2) or there is a directed path between them, and all intermediate objects are blank nodes. Thus, Object Graph captures all direct relations between named objects, and also captures indirect relations formed by blank nodes. The edges are weighted by closeness between objects.

Definition 2 (Object Description): Given an object o_{i} in linked data ℓ, the object description of o_{i} in ℓ is a bag of words defined by Equation (1):

$$d(o_{\mathrm{i}}) = \cup \{d_{\mathrm{uri}}(o_{\mathrm{i}}), d_{\mathrm{labl}}(o_{\mathrm{i}}), d_{\mathrm{comm}}(o_{\mathrm{i}}), d_{\mathrm{anno}}(o_{\mathrm{i}})\} \quad (1)$$

In Eq.(1), $d_{\mathrm{uri}}(o_{\mathrm{i}})$ contains words in the URI of o_{i}; $d_{\mathrm{labl}}(o_{\mathrm{i}})$ and $d_{\mathrm{comm}}(o_{\mathrm{i}})$ are words occurred in *rdfs*:*label* and *rdfs*:*comment* properties of o_{i} respectively; $d_{\mathrm{anno}}(o_{\mathrm{i}})$ is the words from other annotation properties of o_{i}. $\cup$ is the operation of merging bags of words.

Definition 2 (**Virtual Document**): A virtual document $\mathcal{V}d_n(o_i)$ is a bag of words encapsulating the linguistic information of object o_i and its *n*-step surrounding neighbors. The 0-step Virtual Document of o_i $\mathcal{V}d_0(o_i) = d(o_i)$.

$$neighbor_n(o_i) = \overrightarrow{neighbor_n}(o_i) \cup \overleftarrow{neighbor_n}(o_i) \quad (2)$$

$$\mathcal{V}d_n(o_i) = \cup_{o_j \in neighbor_n(o_i)} d(o_j) \quad (3)$$

In Eq.(2-3), $\overrightarrow{neighbor_n}(o_i)$ and $\overleftarrow{neighbor_n}(o_i)$ represent the set of objects that o_i can access through a forward or backward *n*-step links. $\mathcal{V}d_n(o_i)$ is the virtual document of o_i comprising all object descriptions of itself and its *n*-step neighbors.

The notion of virtual document is originated from [3], which aimed at capturing linguistic information for ontology matching. While an object description provides firsthand but limited information about the semantics of an object, a virtual document is a comprehensive and abundant corpus to characterize the object.

2.2 Architecture

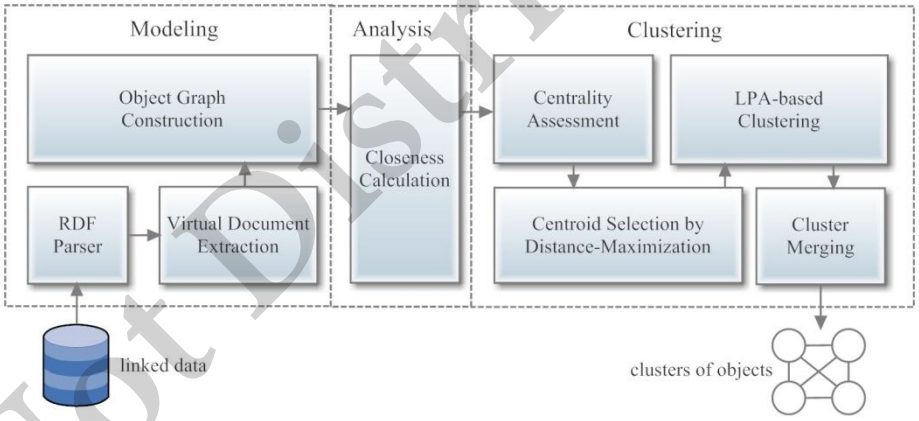


Fig. 2. Architecture of Centrality-based Clustering

As shown in Fig. 2, our approach of clustering is architected into three layers. The Modeling Layer uses an RDF parser to get the RDF model of a linked data as input. Then virtual document of each object is then extracted, and the Object Graph is constructed from RDF model. Derived Object Graph will be passed to Analysis Layer, whose major task is to calculate the relational and linguistic closeness between objects, or in other words, to refine the edge weights of Object Graph. The last layer, Clustering Layer, will first assess the centrality of each object in Object Graph, then utilize the centrality as an indicator to produce a set of important object as centroid

candidates. *k* centroids are selected using a distance-maximization strategy. For each centroid, an LPA-based clustering will be carried to produce clusters. Finally, isolated objects and sub-graphs will be merged into *k* clusters.

3 Closeness Calculation

In linked data, two objects are deemed to be close in two ways: (1) They are close if there is an explicit statement that they have a relation. For example: a student who knows another student. (2) They are similar in semantics, which can be captured in their linguistic information, even if they don't have a direct relation. For example, two researchers can be semantically close when there is no co-authorship, but the textual descriptions of them indicate that they are quite similar in research interests.

In addition to relations, linguistic similarity in linked data is an important indicator for clustering of objects. Some Semantic Web tasks rely on the analysis of object descriptions, such as entity linking from unstructured text to semantic objects. These tasks will benefit if linguistically close objects can be grouped together. Besides, objects with similar descriptions are possible to develop a potential relation in the future, such as the two researchers with same research interests. In our approach, linguistic closeness will affect the clustering in three aspects: the weighting of edges in Object Graph, the LPA-based clustering of objects and the merge of isolated objects and sub-graphs into clusters.

The relational part of closeness $\mathcal{W}_{\mathrm{r}}(o_{\mathrm{i}}, o_{\mathrm{j}})$ is calculated by Rule 1 and 2. The linguistic part of closeness $\mathcal{W}_{\mathrm{l}}(o_{\mathrm{i}}, o_{\mathrm{j}})$ is calculated by Eq.(4). Finally, edge weights in Object Graph is calculated as the multiply of the two parts as shown in Eq.(5). Given linked data ℓ:

Rule 1: For each $o_{\mathrm{j}} \in \overrightarrow{neighbor_1}(o_{\mathrm{i}})$ or $o_{\mathrm{k}} \in \overleftarrow{neighbor_1}(o_{\mathrm{i}})$ in ℓ, there is a directed edge from o_{i} to o_{j} or from o_{k} to o_{i} in $\mathcal{G}(\ell)$. $\mathcal{W}_{\mathrm{r}}(o_{\mathrm{i}}, o_{\mathrm{j}})$ or $\mathcal{W}_{\mathrm{r}}(o_{\mathrm{k}}, o_{\mathrm{i}})$ equals to the number of distinct relations from o_{i} to o_{j} or from o_{k} to o_{i} respectively.

Rule 2: For each $o_{\mathrm{j}} \in \overrightarrow{neighbor_n}(o_{\mathrm{i}})$ or $o_{\mathrm{k}} \in \overleftarrow{neighbor_n}(o_{\mathrm{i}})$ in ℓ, if all intermediate nodes lie on the *n*-step path from o_{i} to o_{j} or from o_{k} to o_{i} are blank nodes, $\mathcal{W}_{\mathrm{r}}(o_{\mathrm{i}}, o_{\mathrm{j}}) = 1/\mathrm{n}$ or $\mathcal{W}_{\mathrm{r}}(o_{\mathrm{k}}, o_{\mathrm{i}}) = 1/\mathrm{n}$ respectively.

$$\mathcal{W}_{\mathrm{l}}(o_{\mathrm{i}}, o_{\mathrm{j}}) = cos\theta = \frac{\overrightarrow{\mathcal{VD}_{\mathrm{n}}(o_{\mathrm{i}})} \cdot \overrightarrow{\mathcal{VD}_{\mathrm{n}}(o_{\mathrm{j}})}}{\|\mathcal{VD}_{\mathrm{n}}(o_{\mathrm{i}})\| \cdot \|\mathcal{VD}_{\mathrm{n}}(o_{\mathrm{j}})\|} \tag{4}$$

$$\mathcal{W}(o_{\mathrm{i}}, o_{\mathrm{j}}) = \mathcal{W}_{\mathrm{r}}(o_{\mathrm{i}}, o_{\mathrm{j}}) \times \mathcal{W}_{\mathrm{l}}(o_{\mathrm{i}}, o_{\mathrm{j}}) \tag{5}$$

In Eq.(6), $\overrightarrow{\mathcal{VD}_n(o_{\mathrm{i}})}$ is the term vector of *n*-step virtual document of o_{i}, and $\|\mathcal{VD}_n(o_{\mathrm{i}})\|$ is the document length.

4 Centrality Assessment

The centrality measurements are to find the potential of objects to be centroids of clusters. Heuristically, objects with high centrality are more likely and adequate to be the center of a cluster, comparing to ones with low centrality.

Various notions of centrality and their measurements have been proposed in literals. They can be classified into three categories: Degree centrality, Shortest-Path-based centrality and Eigenvector centrality.

Degree is a simple yet powerful measurement of objects' centrality in Object Graph. Relations between objects can be seen as conferral of importance. Objects with high degree centrality are intuitively important in the graph since they receive many conferral of importance from others. In our approach, degree centrality of object o_{i} is noted as $C_D(i)$.

Shortest-Path-based centrality is a set of notions based on shortest paths linking pairs of vertices, such as the Betweenness Centrality [4] measured by the ratio of shortest paths across it in Object Graph. The calculation of Shortest-Path-based centralities usually has a high computational complexity, which makes it difficult to adapt to big data, such as linked data. Besides, this category of centralities doesn't outperform degree centrality in some Semantic Web tasks, such as stated in [5]. Considering the scalability, Shortest-Path-based centrality is not adopted in our approach.

The calculation of eigenvector centrality is based on finding the eigenvector of the adjacency matrix encoding a graph. Two well-known measurements of eigenvector centrality on the Web are PageRank [6] and HITS [7]. PageRank is used by the Google search engine for ranking web pages. The authority of a page is computed recursively as a function of the authorities of the pages that link to it. HITS computes two values related to topological properties of the Web pages, the "authority" and the "hubness". In our approach of clustering, three weighted variations of PageRank and HITS are used to define the eigenvector centrality of objects in linked data.

In Eq.(6-9), $C_{PR}(i)$ is the original PageRank centrality. $C_{WPR}(i)$ is an weighted extension to $C_{PR}(i)$. $C_{HITS-A}(i)$ and $C_{HITS-H}(i)$ are the weighed extension of the authority and hubness in HITS algorithm. In the calculation of weighted HITS, the symbol $\|x\|$ means the normalization of *x* after each iteration.

$$C_{PR}(i) = \frac{1-d}{|\mathcal{O}|} + d \times \sum_{j \in \overrightarrow{neighbor_1}(i)} \frac{C_{PR}(j)}{|\overrightarrow{neighbor_1}(j)|} \tag{7}$$

$$C_{WPR}(i) = \frac{1-d}{|\mathcal{O}|} + d \times \sum_{j \in \overrightarrow{neighbor_1}(i)} \frac{\mathcal{W}(o_{\mathrm{j}},o_{\mathrm{i}}) \times C_p(j)}{\sum_{j \in \overrightarrow{neighbor_1}(\mathrm{i})} \mathcal{W}(o_{\mathrm{j}},o_{\mathrm{i}})} \tag{8}$$

$$C_{HITS-A}(i) = \left\| \sum_{j \in \overrightarrow{neighbor_1}(i)} \mathcal{W}(o_{\mathrm{i}}, o_{\mathrm{j}}) \times C_{HITS-H}(j) \right\| \tag{9}$$

$$C_{HITS-H}(i) = \left\| \sum_{j \in \overrightarrow{neighbor_1}(i)} \mathcal{W}(o_{\mathrm{j}}, o_{\mathrm{i}}) \times C_{HITS-A}(j) \right\| \tag{10}$$

5 Centroid Selection and Clustering

Centrality of objects indicates their topological and topical importance in linked data. An object with high centrality is usually a center object surrounded by a set of close-neighboring objects. With a set of selected centroids, the huge amount of objects in a given linked data can be clustered based on the distance between centroids and non-centroids, which is the basic idea of many clustering algorithms, such as the commonly used K-means clustering.

A naïve strategy to find centroids is to simply select top-ranked objects according to their centralities. Given k as an expected cluster numbers, top-k objects with high centralities will be chosen as centroids. However, there is a well-known TKC (Tightly-Knit Community) effect stated in [8], which could make the centrality-based clustering problematic. Objects in a tightly-knit community will mutually reinforce their centralities and dominate the set of top-k selected centroids. A clustering based on these centroids will result in a poor coverage on the whole dataset. In our approach, a set of $10k$ of candidate centroids will be selected beforehand according to their centrality. This enlarged candidate set contains all possible centroids to be further selected. A distance-maximization strategy is proposed in Algorithm 1, in which k centroids are selected one by one considering their distance to pre-selected centroids. The goal of this strategy is to maximize the mutual distance among centroids in linked data, to fulfill a well-covered clustering of objects.

Algorithm 1 : Distance-maximization Strategy for Centroid Selection

Input: a set of objects $\mathcal{O}$ with centrality values $\mathcal{C}$, parameter k as the expected number of clusters.

1. Set the set of centroid $\mathcal{O}_c$ to an empty set;
2. Rank the set of objects $\mathcal{O}$ in descending order according to $\mathcal{C}$;
3. Select top $10k$ objects in $\mathcal{O}$ to form a set of centroid candidates: $\mathcal{O}_{c'}$;
4. $\mathcal{O}_c \leftarrow \{ o_i \mid o_i \in \mathcal{O}_{c'}$ and o_i has top centrality in $\mathcal{O}_{c'}\}$;
5. $\mathcal{O}_{c'} \leftarrow \mathcal{O}_{c'}/\{o_i\}$;
6. Repeat, until $|\mathcal{O}_c| = k$:
   a) Find o_i in $\mathcal{O}_{c'}$, $o_i = argmax \sum_{o_j \in \mathcal{O}_c} d(o_i, o_j)$
   b) $\mathcal{O}_c \leftarrow \mathcal{O}_c \cup \{o_i\}$;
   c) $\mathcal{O}_{c'} \leftarrow \mathcal{O}_{c'}/\{o_i\}$;

Output: the set of centroids $\mathcal{O}_c$

In Algorithm 1, $d(o_i, o_j)$ represents the distance between o_i and o_j. Its calculation is shown in Eq.(10), in which $o_k, o_l \in \rho(o_i, o_j)$ means o_k, o_l lie on a shortest path $\rho(o_i, o_j)$ between o_i and o_j:

$$d(o_i, o_j) = \sum_{o_k, o_l \in \rho(o_i, o_j)} 1 - \mathcal{W}(o_k, o_l) \tag{11}$$

After centroid selection, all non-centroids will be grouped into *k* clusters. An LPA-based (Label Propagation Algorithm) clustering is proposed in Algorithm 2. Each centroid will propagate its cluster label to neighboring objects iteratively until no more objects can be reached. Different with the original LPA, when a non-centroid object is propagated with multiple labels during the iteration, its label will be judged to the cluster whose centroid has the greatest linguistic closeness to it.

Algorithm 2 : LPA-based Clustering

Input: the set of centroids $\mathcal{O}_c = \{o_1, o_2, \dots, o_k\}$

1. Initially set $\mathcal{O}_1 \leftarrow \{o_1\}, \mathcal{O}_2 \leftarrow \{o_2\}, \dots, \mathcal{O}_k \leftarrow \{o_k\}$
2. Repeat, until no more object can be merged into $\mathcal{O}_1$ to $\mathcal{O}_k$:
   a) For each $\mathcal{O}_p \in \{o_1, \dots, o_k\}$,
      i. For each $o_i \in \mathcal{O}_p$, find $\mathcal{O}_p' = \mathcal{O}_1 \cup \overleftarrow{neighbor_1}(o_i) \cup \overrightarrow{neighbor_1}(o_i)$;
      ii. For each $o_j \in \mathcal{O}_p'$, label o_j with a cluster id: p
   b) For each non-centroid object o_j, and o_j has been labeled with multiple cluster ids, re-label its cluster id with the cluster whose centroid has the greatest $\mathcal{W}_1$ to o_j.
   c) For those labeled non-centroid objects, merge them into corresponding clusters according to their cluster ids.
3. For each remaining non-centroid objects (isolated objects or sub-graphs, etc.), classify them into k clusters according to $\mathcal{W}_1$.

Output: A clustering of $\mathcal{O}$ into k clusters: $\mathcal{O}_1, \mathcal{O}_2, \dots, \mathcal{O}_k$

Considering there may be isolated objects or sub-graphs remained after clustering, the step 3 of Algorithm 2 will finally merge them into the k clusters. The merging of remaining objects is basically a text classification problem, which utilize the linguistic closeness between each remaining object and *k* centroids. We omit the details of merging for the sake of conciseness.

6 Evaluation

In this section, we first analysis the datasets, then evaluate the performance of different centrality measurements and the final clustering. We carried out these evaluations on our server with Intel Xeon E3 V2 processors and 16G RAM.

6.1 Datasets

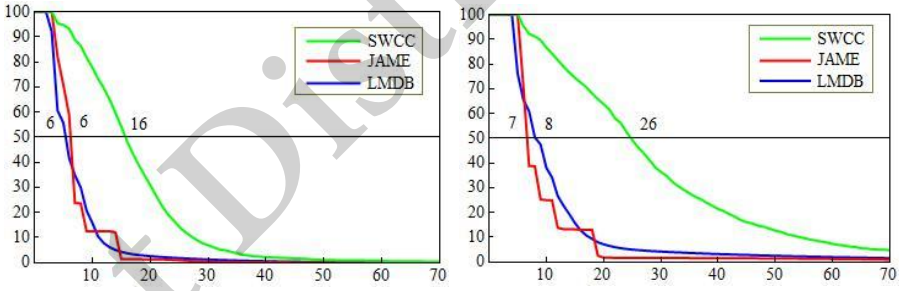
Three linked data are selected as the dataset of experiments, i.e., 1) Semantic Web

Conference Corpus (SWCC in short)[1], which is a data on Semantic Web Conference; 2) Jamendo (JAME in short)[2], which is a data on licensed music; 3) LinkedMDB (LMDB in short)[3], which is a data for movies; In Table 1, the statistics of each dataset is presented. #triple is the total number of triples; #object is the number of objects; #class and #properties represent the number of classes and properties that the dataset used as vocabulary; #relation is the number of object links, which is also the number of edges in Object Graph.

Table 1. Statistics of each linked data

Data	#triple	#object	#class	#property	#relation
SWCC	20,802	3,089	20	77	10,589
JAME	1,049,647	412,565	12	26	505,961
LMDB	6,247,909	1,326,885	53	222	2,069,454

Fig.3 shows the abundance of linguistic information in each dataset. In Fig.3(a), and Fig.3(b), the X axis respectively represents the number of unique words in a certain object's 1-step virtual document, and the document length of virtual document. In both figures, the Y axis represents the percentage of objects whose linguistic information is equal to or more abundant than a given value. A median line is drawn to illustrate the average linguistic abundance in each dataset. From both figures we can observe that the SWCC has the most abundant linguistic information, while the JAME has the least.



(a) (b)

Fig. 3. Statistics of linguistic abundance on (a) unique word (b) virtual document length

6.2 Evaluations on centrality assessment

To evaluate which measurement will produce the most reasonable candidate set, a prior ground true of human judgment should be generated, and the agreement among

[1] SWCC: http://data.semanticweb.org/
[2] JAME: http://dbtune.org/jamendo/
[3] LMDB: http://linkedmdb.org/

human-generated and machine-generated candidate sets will be calculated to find the best measurement, as stated in [5]. However, for the evaluation on large-scale linked data, the generation of ground true by human is impossible. Instead, we use the agreement among five machine-generated centralities, as well as their time performances, as selectors to filter out three measurements for the final clustering.

We use Kendall's tau statistic [9] to calculate the correlation among ranked candidate sets produced by degree centrality (*DE* in short), PageRank centrality (*PR*), Weighted PageRank (*WPR*), HITS-authority (*HA*) and HITS-hubness (*HH*). The calculation is shown in Eq.(12), where the correlation τ is the odds that two objects are ranked concordantly against discordantly in two candidate sets. The agreements among five centralities are shown in Table 2.

We use Gephi 0.9.1[4] as our tool for centrality assessment. The time performance of each measurement is shown in Table 3. From the results in both tables, we select *DE*, *WPR* and *HA* as the final measurements to produce centroid candidates. *DE* is selected because its simpleness and efficiency in calculation. *WPR* is selected because it concerns linguistic information in centrality assessment and shows a difference with non-weighted PageRank. *HA* is selected because it shows a good correlation with *DE* on two datasets, and also has a sound time performance.

$$\tau = \frac{\#concordant - \#discordant}{n(n-1)/2} \tag{12}$$

Table 2. Agreement between various centralities

SWCC						JAME						LMDB					
	DE	PR	WPR	HA	HH		DE	PR	WPR	HA	HH		DE	PR	WPR	HA	HH
DE	1	-0.08	-0.21	-0.10	-0.23	DE	1	-0.21	-0.31	1	-0.19	DE	1	-0.11	-0.05	0.35	-0.05
PR	-0.08	1	-0.16	-0.23	-0.29	PR	-0.21	1	0.13	-0.21	-0.15	PR	-0.11	1	0.19	-0.03	-0.25
WPR	-0.21	-0.16	1	-0.38	-0.44	WPR	-0.31	0.13	1	-0.31	-0.18	WPR	-0.05	0.19	1	-0.02	-0.25
HA	-0.10	-0.23	-0.38	1	-0.06	HA	1	-0.21	-0.31	1	-0.11	HA	0.35	-0.03	-0.02	1	-0.10
HH	-0.23	-0.29	-0.44	-0.06	1	HH	-0.19	-0.15	-0.18	-0.11	1	HH	-0.05	-0.25	-0.25	-0.10	1

Table 3. Time consumption of centrality assessment (ms)

	DE	PR	WPR	HA	HH
SWCC	36.8	118	111.3	33.8	33.8
JAME	1,237.5	2,435	3,086.5	1,333.3	1,333.3
LMDB	4,765	24,160	39,045.2	5,557.7	5,557.7

6.3 Evaluations on clustering

After the generation of centroid candidates, k centroids will be selected and the dataset will be decomposed into k clusters. To evaluate the performance of clustering, we use K-means as the baseline clustering algorithm. Weka 3 is used as our tool for

[4] Gephi: https://gephi.org/users/download/

K-means clustering. We use *Connectedness* defined in [10] as the indicator for the quality of clustering, which is commonly used in the evaluation of ontology modularization. The calculation of *Connectedness* is shown in Eq.(13), where $|E_x|$ is the number of shared edges in between clusters, and $|E|$ is the number of all edges.

$$connectedness = \frac{|E_x|}{|E|} \tag{13}$$

Table 4 shows the resulted quality of clustering. Both *DE*, *WPR* and *HA* produce high-quality clusters with our LPA-based clustering algorithm. The average performance on all datasets indicates that *WPR* is the best choice comparing to other two measurements, and it produces clustering with less than 5 percents of shared edges in between clusters. As we expected, K-means failed to decompose JAME and LMDB because of its computational complexity and the data volume. K-means only successfully decomposed SWCC with a *connectedness* of 0.203, which indicates a much lower quality of clustering comparing to our approach.

Table 4. Quality evalution of different clusterings

	K-means	DE	WPR	HA
SWCC	0.203	0.122	0.120	0.113
JAME	--------	0.021	0.023	0.021
LMDB	--------	0.014	0.010	0.056
Avg.	--------	0.052	0.048	0.063

7 Related Works

To the best of our knowledge, clustering or community detection in linked data is still a research area not being deeply explored. Grimnes et al. presented in [11] several ways to extract instances from RDF graph and computing the distance between them. The challenge surrounding the application of clustering algorithms to Semantic Web data was also discussed. Yan proposed RDF graph partitioning in [12], in which large RDF graph would be partitioned into sub-graphs and stored individually. In [13], Aluc proposed RDF clustering for RDF data management. They kept track of RDF records in DB that are co-accessed by queries in the workload and physically clustered them. These works differs with our approach that their goal of portioning RDF graph is to fulfill a self-adaptive RDF management to improve the efficiency of SPARQL query, while our approach aims at discovering meta-structure of linked data for diverse Semantic Web tasks.

Although object clustering hasn't been fully discussed in Semantic Web research community, centrality-based clustering on large-scale graphs has been discussed in the research of network science. Tabrizi proposed in [14] a personalized PageRank

clustering based on random walks, which has a linear time and space complexity. The basic idea of this work is similar to ours. Since the dataset of this work is web pages, our work differs with it in many aspects: the graph model, the calculation of closeness, the centroid selection strategy and the clustering algorithm. However, it motivates us and proves that centrality-based clustering in large-scale linked data is feasible.

8 Conclusion and Future work

The identification of object clusters in linked data is of crucial importance as they may help to scale down the problem when exploring linked data, or may help researchers to understand the meta-structure of the linked data. We propose an efficient centrality-based object clustering in this paper. Object Graph is introduced as the graph model of clustering. The closeness between two objects is measured in both relational and linguistic manner. A distance-maximization strategy is used to select centroids from candidates with high centrality. An LPA-based clustering decomposes linked data into *k* clusters. Our experiments show that our approach is feasible in large-scale linked data.

In our future work, we will explore the possibility of a guided clustering, in which object clustering will be guided by ontology modularization. The modules in TBox may provide information about how different types of objects are related. We will also try to performance our clustering on larger linked data, such as DBpedia. A visualized system of object clusters will be constructed for better human understanding.

Acknowledgement

The work was supported by the National High-Tech Research and Development (863) Program of China (No.2015AA015406) and the Open Project of Jiangsu Key Laboratory of Data Engineering and Knowledge Service (No. DEKS2014KT002).

Reference

1. Hartig, O., Bizer, C., Freytag, J.C.: Executing SPARQL queries over the Web of Linked Data. In: Proccedings of 8th International Semantic Web Conference (ISWC 2009). pp. 293–309 (2009).
2. Paulheim, H.: Exploiting Linked Open Data as Background Knowledge in Data Mining. In: Proceedings of the International Workshop on Data Mining on Linked Data, with Linked Data Mining Challenge collocated with ECMLPKDD 2013. pp. 1–10 (2013).
3. Qu, Y., Hu, W., Cheng, G.: Constructing Virtual Documents for Ontology Matching. In: Proceedings of the 15th international conference on World Wide Web (WWW2006). pp. 23–31 (2006).
4. Newman, M.E.J.: A measure of betweenness centrality based on random walks. Soc. Networks. 27, 39–54 (2005).
5. Zhang, X., Cheng, G., Qu, Y.: Ontology Summarization Based on RDF Sentence Graph. In: Proceedings of the 16th international conference on World Wide Web - WWW '07. p. 707 (2007).

6. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking:bringing order to the web. Tech. report, Stanford Digit. Libr. Technol. Proj. (1998).
7. Kleinberg, J.M.: Authoritative Sources in a Hyperlinked Environment. J. ACM. 46, 668–677 (1999).
8. Lempel, R., Moran, S.: Stochastic approach for link-structure analysis (SALSA) and the TKC effect. Comput. Networks. 33, 387–401 (2000).
9. Sheskin, D.J.: Handbook of parametric and nonparametric statistical procedures. Technometrics. 46, 1193 (2004).
10. Schlicht, A., Stuckenschmidt, H.: Towards structural criteria for ontology modularization. In: CEUR Workshop Proceedings (2006).
11. Grimnes, G.Aa., Edwards, P., Preece, A.: Instance based clustering of semantic web resources. In: Proceedings of the 5th European semantic web conference on The semantic web: research and applications. pp. 303–317 (2008).
12. Yan, Y., Wang, C., Zhou, A., Qian, W., Ma, L., Pan, Y.: Efficient indices using graph partitioning in RDF triple stores. In: Proceedings - International Conference on Data Engineering. pp. 1263–1266 (2009).
13. Aluç, G., Özsu, M.T., Daudjee, K.: Clustering RDF Databases Using Tunable-LSH. CoRR, abs/1504.02523. 1–13 (2015).
14. Tabrizi, S.A., Shakery, A., Asadpour, M., Abbasi, M., Tavallaie, M.A.: Personalized PageRank Clustering: A graph clustering algorithm based on random walks. Phys. A Stat. Mech. its Appl. 392, 5772–5785 (2013).

Boosting to Build a Large-scale Cross-lingual Ontology

Zhigang Wang, Liangming Pan, Juanzi Li, Shuangjie Li, Mingyang Li, and Jie Tang

Department of Computer Science and Technology,
Tsinghua University, Beijing 100084, P.R. China
{wzhigang,plm,ljz,lsj,lmy,tangjie}@keg.cs.tsinghua.edu.cn

Abstract The global knowledge sharing makes large-scale multi-lingual knowledge bases an extremely valuable resource in the Big Data era. However, current mainstream Wikipedia-based multi-lingual ontologies still face the following problems: the scarcity of non-English knowledge, the noise in the multi-lingual ontology schema relations and the limited coverage of cross-lingual `owl:sameAs` relations. Building a cross-lingual ontology based on other large-scale heterogenous online wikis is a promising solution for those problems. In this paper, we propose a cross-lingually boosting approach to iteratively reinforce the performance of ontology building and instance matching. Experiments output an ontology containing over 3,520,000 English instances, 800,000 Chinese instances, and over 150,000 cross-lingual instance alignments. The F1-measure improvement of Chinese `instanceOf` prediction achieve the highest 32%.

Keywords: Ontology Building, Instance Matching, Cross-lingual

1 Introduction

As the Web is evolving to a highly globalized information space, sharing knowledge across different languages is attracting increasing attentions. Multilingual ontologies, in which the cross-lingual equivalent concepts or relationships are linked together using `owl:sameAs`, are important sources for harvesting cross-lingual knowledge from the Web and have significant applications such as multi-lingual information retrieval, machine translation and deep question answering. DBpedia[1], by extracting structured information from Wikipedia in 111 different languages, is a multi-lingual multi-domain knowledge base and becomes the nucleus of LOD. Obtained from WordNet and Wikipedia, YAGO, MENTA, and BabelNet are other famous large multi-lingual ontologies [6,7,12].

Though lots of researches have been done, there are still some problems to be solved. Firstly, the imbalance of different Wikipedia language versions leads to the highly unbalanced knowledge distribution in different languages. Figure 1 shows a simplified long tail distribution of the number of articles on six major Wikipedia language versions. Most non-English knowledge in these ontologies is pretty scarce. Secondly, the noise of the large category system in Wikipedia leads to the incorrect semantic relations in these ontologies. For example, "Wikipedia-books-on-people is the `subCategoryOf` People" will lead

to the wrong "Wikipedia-books-on-people is `subClassOf` People" in DBpedia's SKOS schema. And the relatively precise WordNet only cover some aspects of domains in English. Finally, because those ontologies are integrated directly by Wikipedia's cross-lingual links, the coverage of cross-lingual `owl:sameAs` relations in those ontologies is limited by the number of existing cross-lingual links.

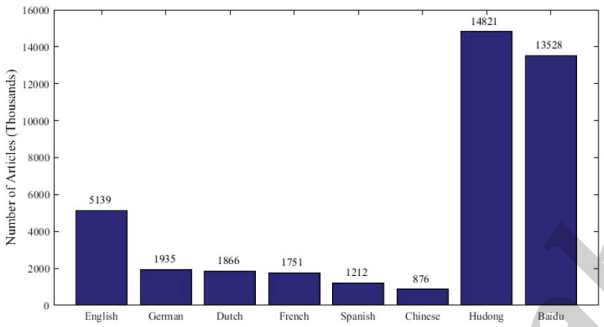


Figure 1. Number of Articles on Major Wikipedias, Hudong Baike and Baidu Baike

On the other hand, there are more and more similar large-scale non-English online wikis in big data era. For example, the Chinese Hudong Baike and Baidu Baike, both containing more than 6 million articles, are even larger than the English Wikipedia (the largest Wikipedia language version). If multi-lingual ontology could be established between two large online wikis, such as English Wikipedia and Chinese Hudong Baike, multi-lingual ontologies with much higher coverage can be constructed.

In this paper, we try to build a large-scale cross-lingual ontology based on two heterogeneous online wikis in different languages. To our best of knowledge, we are the first to combine the processes of mono-lingual ontology building and cross-lingual instance matching together to build a cross-lingual ontology. Our work is motivated by two observations on the multilingual knowledge distributions. ***Cross-lingual Knowledge Consistency***. A lot of facts are considered as correct all over the world, e.g. the facts about Science. Mining consistency across different languages not only helps to match equivalent cross-lingual knowledge, but also assists to improve the performance of mono-lingual ontology building each other. ***Cross-lingual Knowledge Discordance***. The facts people concern or believe are quite different. E.g. the Chinese instance "China" is more linked to the Chinese locations but the English instance "China" is more linked to the counties in the world. Consideration of this problem in depth can help avoid incorrect matching.

This non-trivial task poses the challenges as follows, how to build two large-scale mono-lingual ontologies with correct semantic relations? How to construct an effective and efficient language-independent instance matching model? And how to boost the building of the cross-lingual ontology iteratively? Driven by these challenges, we propose a unified boosting framework to iteratively build a cross-lingual ontology. Our contributions are as follows.

1. We propose a binary classification-based method for large-scale mono-lingual ontology building, and a language-independent instance matching method. The ontology building method is able to eliminate the noise inside the wikis by predicting the correct `subClassOf` and `instanceOf` relations. The ontology matching method works for two highly heterogenous cross-lingual ontologies effectively and efficiently.
2. We propose a cross-lingually boosting method to reinforce the processes of ontology building and instance matching. The cross-lingual knowledge consistency and discordance are analyzed in depth. We iteratively expand the volume of labeled data for ontology building and expand the cross-lingual alignments for instance matching to improve the quality of built ontology simultaneously.
3. We conduct an experiment using the English Wikipedia and Hudong Baike data sets. Experimental results show that our boosting method outperforms the non-iterative method. The F1-measure of ontology building functions has an improvement of above 6%. In particular, the performance of Chinese `instanceOf` function get a high 32% improvement for F1-measure. A large ontology containing 3,520,000 English instances and 800,000 Chinese instances is built. Over 150,000 cross-lingual instance alignments are constructed.

2 Preliminaries

Basic Concepts. Given two online wikis in different languages and an initial aligment set, our target is to build two mono-lingual ontologies and find the equivalent alignments between them.

Definition 1. An ***online wiki*** is a graph containing a set of entities and a set of links between two entities. It can be formally represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $v \in \mathcal{V}$ denotes an *entity* and has an related *document*. We have $\mathcal{E} = \mathcal{V} \times \mathcal{V}$, and $e_{ij} \in \mathcal{E}$ indicate whether there exists a `subCategoryOf` or `articleOf`[1] relation from v_i to v_j (1 for yes, 0 for no).

Definition 2. An ***ontology*** is defined as a 2-tuple of the set of entities and the set of semantic relations. It can be formally represented as $\mathcal{O} = (\mathcal{X}, \mathcal{Y})$, where $x \in \mathcal{X}$ denotes a *concept* in the schema-level or an *instance* in the instance-level. We have $\mathcal{Y} = \mathcal{X} \times \mathcal{X}$ and $y_{ij} \in \mathcal{Y}$ indicate whether there exists a legal semantic relation from y_i to y_j (1 for yes, 0 for no). We only consider two kinds of semantic relations, which are `subClassOf` between two concepts and `instanceOf` from one instance to one concept.

Definition 3. The ***alignment set*** is the set of equivalent instances between two ontologies. It can be formally represented as $\mathcal{A} = \{a_i\}$, where $a_i = (x, x')$ denotes the equivalent instances between two ontologies respectively.

Problem Formulation. Given two online wikis $\mathcal{G}_1 = (\mathcal{V}, \mathcal{E})$, $\mathcal{G}_2 = (\mathcal{V}', \mathcal{E}')$ and an initial alignment set $\mathcal{A} = \{a_i\}_{i=1}^{m}$, we aim at constructing two mono-lingual ontologies $\mathcal{O}_1 = (\mathcal{X}, \mathcal{Y})$, $\mathcal{O}_2 = (\mathcal{X}', \mathcal{Y}')$ and a cross-lingual alignment set

[1] We use category and article to denote the concept and instance in the online wiki respectively.

$\mathcal{A}' = \{a_i\}_{i=1}^{n}$. We have $n > m$, and $\mathcal{G}_1$, $\mathcal{G}_2$ are in two different languages[2]. The entities of the constructed ontologies are from the entities of online wikis, where $\mathcal{X} \subseteq \mathcal{V}$ and $\mathcal{X}' \subseteq \mathcal{V}'$. Thus, our major issue is to predict three kinds of relations, which are `subClassOf` between two concepts in each ontology, `instanceOf` from one instance to one concept in each ontology, and `equalTo` between two instances from two ontologies.

We formalize this problem as multiple binary classification problems. More formally, we are to learn two kinds of classification functions with a confidence output as follows.

- ***Instance Matching Function*** $f : \mathcal{X} \times \mathcal{X}' \mapsto [0, 1]$ to predict the probability to be `equalTo` relation between two instances x and x' from $\mathcal{O}_1$ and $\mathcal{O}_2$ respectively.
- ***Ontology Building Function*** $g_1 : \mathcal{V} \times \mathcal{V} \mapsto [0, 1]$ to predict the probability to be `subClassOf` or `instanceOf` relation between two entities v_i and v_j in $\mathcal{G}_1$, or $g_2 : \mathcal{V}' \times \mathcal{V}' \mapsto [0, 1]$ in $\mathcal{G}_2$.

To improve the performance of the isolated functions, we boost to mutually reinforce the learning of the building and matching functions.

3 Approach

As shown in Figure 2, our approach is a boosting method. In each iteration we use the results of ontology building g_1, g_2 and instance matching f to reinforce the learning performance in the next iteration.

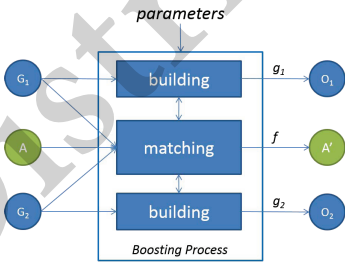


Figure 2. Overview of the Proposed Approach

3.1 Mono-lingual Ontology Building

We take the entities of $\mathcal{V}$, $\mathcal{V}'$ in the online wikis $\mathcal{G}_1$ and $\mathcal{G}_2$ as the entities of $\mathcal{X}$, $\mathcal{X}'$ in the ontoloties $\mathcal{O}_1$ and $\mathcal{O}_2$. Concretely, we take the categories in wikis as the concepts, and take the articles as the instances. Hence, our task is to learn the ontology building functions g_1 and g_2 to predict the correct `subClassOf` or `instanceOf` relations between two entities. We view both the correct `subClassOf` relation between two concepts and the correct `instanceOf` relation from an instance to a concept as an `is-a` relation. Table 1 shows some examples about the semantic relations generated from the online wikis.

Table 1. Examples of Semantic Relations

Entity 1	Relation	Entity 2	Right or Wrong
European Microstates	`instanceOf`	Microstates	Right
European Microstates	`instanceOf`	Europe	Wrong
教育人物(Educational Person)	`subClassOf`	人物(Person)	Right
教育人物(Educational Person)	`subClassOf`	教育(Education)	Wrong

In this paper, we are to learn two series of functions $g_1 : \mathcal{V} \times \mathcal{V} \mapsto [0, 1]$ and $g_2 : \mathcal{V}' \times \mathcal{V}' \mapsto [0, 1]$ to predict the probabilities to be an `is-a` relation between two entities (1 for completely positive, 0 for completely negative). Notice that, we actually train four functions which are English `subClassOf`, English `instanceOf`, Chinese `subClassOf` an Chinese `instanceOf`, but we uniformly represent the ontology building functions of `subClassOf` and `instanceOf` in one language the same. The unique difference between them is that the input entities of `subClassOf` are two concepts but the input entities of `instanceOf` are one instance and one concept.

By manually labeling some training examples, we can learn the Logistic Regression models to get the ontology building functions g_1 and g_2. Table 2 shows the feature definition of g_1 function. The 10th feature is calculated as follows. We firstly list all of the sub-categories of current super-category. Then we calculate the frequency of each word in all of the sub-categories. The score of current sub-category is the sum of the frequency of each word in current sub-category. This feature is similar to a voting process, in which the more frequent words denote a higher probability. Similar as the 11th feature is.

Table 2. Feature Definition for g_1

ID	Feature	Range
1	Is the head word of super-category plural?	$\{0, 1\}$
2	Is the head word of sub-category plural?	$\{0, 1\}$
3	Word length of super-category	Integer
4	Word length of sub-category	Integer
5	Word length of head words of super-category	Integer
6	Word length of head words of sub-category	Integer
7	Relation between the head words of super-category and sub-category	$\{\equiv, \subseteq, \supseteq, \perp, \triangle\}$
8	Does the non-head words of sub-category contain the head words of super-category?	$\{0, 1\}$
9	Does the non-head words of super-category contain the head words of sub-category?	$\{0, 1\}$
10	Score of sub-category	Numeric
11	Score of super-category	Numeric

$\equiv$ equivalent, $\subseteq$ smaller, $\supseteq$ larger, $\perp$ disjoint, $\triangle$ otherwise.

[2] We use $\mathcal{G}_1$ to represent the English online wiki, and use $\mathcal{G}_2$ to represent the Chinese online wiki.

The features in Table 2 are for learning the `subClassOf` predictor of g_1. The `instanceOf` features are similar, in which we replace the super-category into category and replace the sub-category into article. The head words can be extracted using a NLP parser. Note that, for features of g_2, we revise the 1st and 2nd features into "Is the sub-category starting with super-category" and "Is the sub-category ending with super-category" respectively. Besides, the basic unit for g_2 is one Chinese character but not a word. E.g. the 3rd feature is "the length of super-category characters".

3.2 Cross-lingual Instance Matching

Given the initial alignment set $\mathcal{A} = \{a_i\}_{i=1}^{m}$, cross-lingual instance matching is to generate a much larger alignment set $\mathcal{A}' = \{a_i\}_{i=1}^{n}$ $(n >> m)$ between $\mathcal{O}_1$ and $\mathcal{O}_2$. We are to learn the function $f : \mathcal{X} \times \mathcal{X}' \mapsto [0, 1]$ to predict the probability to be `equalTo` relation between two instances x and x'.

By automatically sampling a part of alignments from $\mathcal{A}$ as the training examples, we can learn the Logistic Regression model to get the function f. We firstly present the features for instance matching, and then introduce two preprocessing methods, namely maximum clique pruning and link annotation. Finally, we present the post-processing method.

Feature Definition. The features used in f are designed by the observation of cross-lingual knowledge consistency. Both the lexical similarities and link-based structural similarities are defined. We use the following *Set Similarity* as the basic metric for structural similarities, which has been proven to be quite effective in [15]. Given two instances a and b, let S_a and S_b be their related sets of entities, the ***Set Similarity*** between a and b is calculated as

$$s(a, b) = \frac{2 \cdot |\phi_{1\to 2}(S_a \cap S_b)|}{|\phi_{1\to 2}(S(a))| + |S(b)|} \tag{1}$$

where $\phi_{1\to 2}(\cdot)$ maps the set of entities in $\mathcal{G}_1$ (or $\mathcal{O}_1$) to their equivalent entities in $\mathcal{G}_2$ (or $\mathcal{O}_2$) if the alignment exists.

Table 3 shows the feature definition of f. As we can see, both the structural similarities in the online wikis and in the ontologies are used.

Table 3. Feature Definition for f

Type	ID	Feature	Description
Lexical	1	Edit-distance of titles without translation	Return 0 if there are no common characters.
	2	Difference in word length	$\lvert English\_Word\_Length - Chinese\_Character\_Length\rvert$.
Structural	3	*Set Similarity* of categories	Calculated between $\mathcal{G}_1$ and $\mathcal{G}_2$
	4	*Set Similarity* of outlinks	Calculated between $\mathcal{G}_1$ and $\mathcal{G}_2$
	5	*Set Similarity* of inlinks	Calculated between $\mathcal{G}_1$ and $\mathcal{G}_2$
	6	*Set Similarity* of concepts	Calculated between $\mathcal{O}_1$ and $\mathcal{O}_2$

To overcome the link sparseness, we use a smoothing method in our experiments when computing those structural features.

Maximum Clique Pruning. Due to the cross-lingual knowledge discordance, the knowledge distributions across different languages differs a lot. Our feature definition is apt to choose the correspondences sharing more common related entities. However, we observe that a lot of neighbor entities are not very related in online wikis. E.g. in Hudong Baike, the article "1月1日" (1st, Jan.) is linked to many dates without much relatedness. This will lead to some erroneous correspondences such as "1月1日" (1st, Jan.) `equalTo` "3rd, May". We propose a maximum clique pruning to remove those structurally high linked but semantically low related structures. For each article in $\mathcal{G}_1$ or $\mathcal{G}_2$, we construct a local graph using this article and its linked articles. Then we calculate the maximum clique of this local graph. If the size of the maximum clique is larger 5, we prune the links between any two articles in the clique. In this way, lots of noise can be pruned from the online wikis. We add the similarities on the pruned network as new features for instance matching.

Link Annotation. Due to the link sparseness, the structural similarities across two heterogenous online wikis are quite sparse. To overcome this problem, we conduct a n-gram link annotation process to mine more links. The precision of link annotation is not sensitive, because we use the annotated links as new features for instance matching.

Heuristic Post-processing. Based on our observations, we propose the following rules to filter out some unreliable matching results: (1) *Multiple Correspondence*. If one English instance has been aligned to more than one Chinese instance, we remove all of those correspondences. (2) *Digits or Letters Co-occurrence*. If the Chinese instance's title contains a substring of more than two continuous digits or upper-case letters, we remove the correspondence if the English instance's title doesn't contain the same substring.

3.3 Boosting to Build a Large-scale Ontology

To boost a large-scale cross-lingual ontology, we iteratively learn the ontology building functions and the instance matching function. Figure 3 shows the overview of our boosting method in the iteration of t. Our boosting strategies are different for the building and matching functions.

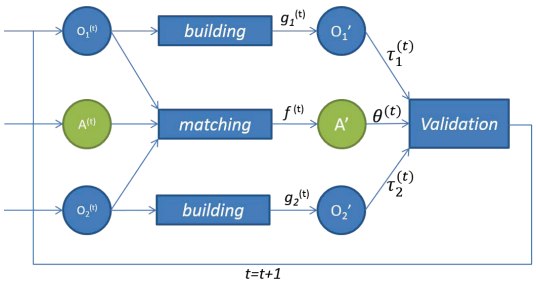


Figure 3. Overview of Boosting Process in the Iteration of t

Boosting the ontology building process. The performance of ontology building functions is related to the volume of manually labeled data sets. Our idea is to expand the training data sets automatically after each iteration by using a cross-lingual semantic validation method. The detailed strategies are as follows.

- Train the ontology building functions $g_1^{(t)}$, $g_2^{(t)}$ using current training data sets.
- Predict the unlabeled data sets using the learned $g_1^{(t)}$, $g_2^{(t)}$.
- Validate the predicted data using current cross-lingual alignments as follows: if $f^{(t)}(x_1, x_1^{'}) > \theta^{(t)}$ and $f^{(t)}(x_2, x_2') > \theta^{(t)}$, then we have $g_1^{(t)}(x_1, x_2) = g_2^{(t)}(x_1', x_2') = 1$ if $g_1^{(t)}(x_1, x_2) + g_2^{(t)}(x_1', x_2') > (\tau_1^{(t)} + \tau_2^{(t)})$, and $g_1^{(t)}(x_1, x_2) = g_2^{(t)}(x_1', x_2') = 0$ if $g_1^{(t)}(x_1, x_2) + g_2^{(t)}(x_1', x_2') < (\tau_1^{(t)} + \tau_2^{(t)})$ (we experimentally set $\theta^{(t)}$, $\tau_1^{(t)}$ and $\tau_2^{(t)}$ to be 0.9, 0.5 and 0.5 respectively. A higher parameter value generates a stricter validation result).
- Expand the training data sets using the cross-lingually validated data.
- Iteratively repeat this process for the next iteration.

Boosting the instance matching process. The structural features of instance matching process are calculated based on the initial alignment set. More alignments help to harvest more precise features. Thus, our idea is to expand the alignment set automatically after each iteration. The detailed strategies are as follows.

- Train the instance matching function $f^{(t)}$ using current alignments.
- Predict the unlabeled data sets using $f^{(t)}$.
- Validate the predicted data sets as follows: if $f^{(t)}(x, x') > \theta^{(t)}$, then we have $f^{(t)}(x, x') = 1$ (we experimentally set $\theta^{(t)}$ to be 0.9).
- Expand the alignment set using the validated alignments.
- Iteratively repeat this process for the next iteration.

4 Experiments

We conduct the experiments using English Wikipedia and Hudong Baike. The English Wikipedia dump is archived in August 2012, and the Hudong Baike dump is crawled from Huong Baike's website in May 2012. We remove all those entities in English Wikipedia, whose titles contain the following strings: *wikipedia*, *wikiprojects*, *lists*, *mediawiki*, *template*, *user*, *portal*, *categories*, *articles*, *pages*, *by*. We also remove the articles in Hudong Baike, which do not belong to any categories of Hudong. Table 4 shows the statistics of the cleaned online wikis.

Using the cross-lingual links between English and Chinese Wikipedias, we get an initial alignment set containing 126,221 alignments between English Wikipedia and Hudong Baike. We use Stanford Parser [2] for extracting the head words and use the Weka [3] toolkit for implementing the learning algorithms. We first evaluate the effectiveness of proposed mono-lingual ontology building and cross-lingual instance matching methods respectively, and then evaluate the proposed boosting approach as a whole.

Table 4. Statistics of Cleaned Data Sets

Online Wiki	#Categories	#Articles	#Links	#Links/#Articles
English Wikipedia	561,819	3,711,928	63,504,926	17.1
Hudong Baike	28,933	980,411	23,294,390	23.8

4.1 Mono-lingual Ontology Building

For the evaluation of mono-lingual ontology building, we randomly selected 3,000 English `subClassOf`, 1,500 Chinese `subClassOf`, 3,000 English `instanceOf`, and 1,500 Chinese `instanceOf` examples. We ask 5 graduate students of Tsinghua University to help us manually label those examples. The examples consented by more than 3 students are kept. Table 5 shows the detail of our labeled examples.

Table 5. Labeled Data for Mono-lingual Ontology Building.

Examples	`subClassOf` **en**	`subClassOf` **zh**	`instanceOf` **en**	`instanceOf` **zh**
Positive	2,123	780	2,097	638
Negative	787	263	381	518

en: English, **zh**: Chinese.

We conduct our experiments with a 5-fold cross-validation, and compare our Logistic Regression (LR) model with two baselines, namely Naïve Bayes (NB) and Support Vector Machines (SVM), using the same features defined in Section 3.1. As shown in Table 6, LR outperforms NB a lot and achieves comparative performance as the SVM (in most cases also outperforms SVM on F1-measure). In consideration of computation cost of the boosting process, our LR method is a good choice owing to its excellent learning efficiency.

Table 6. Results of Mono-lingual Ontology Building. (%)

	`subClassOf` **en**			`subClassOf` **zh**			`instanceOf` **en**			`instanceOf` **zh**		
Methods	P	R	F1	P	R	F1	P	R	F1	P	R	F1
NB	87.1	62.5	72.8	87.1	85.9	86.5	95.8	42.7	59.1	60.2	55.7	57.9
SVM	80.8	86.7	83.6	83.8	98.6	**90.6**	84.5	100	91.6	53.1	82.1	64.5
LR	80.6	87.1	**83.7**	84.0	97.7	90.3	87.4	98.4	**92.6**	56.5	80.1	**66.3**

P: precision, R: recall, F1: F1-measure, **en**: English, **zh**: Chinese.

Table 6 also shows the cross-lingual performance comparison of `subClassOf` and `instanceOf` respectively. We find that English `instanceOf` performs better than Chinese `instanceOf`, but Chinese `subClassOf` is better than English `subClassOf`. This is because the 2nd and 3rd features in learning the building functions are linguistic related. The features are quite effective in learning English `instanceOf` and Chinese `subClassOf` respectively. That indicates the possibility to mutually improve the performance by the boosting process.

4.2 Cross-lingual Instance Matching

In order to evaluate the cross-lingual instance matching method, we randomly select 3,000 initial alignments as the ground truth. We also automatically sample 10,000 random positive and 25,000 random negative alignments as the training

data sets. In the experiments, we aim to investigate how the instance matching method performs before and after the heuristic post-processing (**HP**), and how the instance matching performs with different numbers of alignments. Therefore, we conduct four groups of experiments, each of which uses different number of alignments. In each group, we also compare the performance of our method before and after the heuristic post-processing. Table 7 shows the detailed results. The precision of our method is relatively high but the recall is rather low. We think this still works for our boosting method because the recalled alignments can be enriched iteratively even the recall is relatively low. However, a low precise alignment results will deteriorate the boosting process rapidly.

Table 7. Results of Cross-lingual Instance Matching. (%)

#Alignments	**Before HP**			**After HP**		
	Precision	Recall	F1-measure	Precision	Recall	F1-measure
0.03 Mil.	81.5	5.6	10.5	91.4	5.6	10.6
0.06 Mil.	86.4	6.0	11.3	91.9	6.0	11.3
0.09 Mil.	**89.7**	6.5	12.0	**93.9**	6.5	12.2
0.12 Mil.	86.5	6.8	**12.6**	88.9	6.8	**12.6**

As we can see from Table 7, in each group of the experiments, our method always performs better after the heuristic post-processing (especially for the precision). It shows the heuristic post-processing method can effectively filter out the unreliable matching results. On the other side, the F1-measure of our approach always increases when more alignments are used. Therefore, expanding the initial alignment set iteratively is important for improving the instance matching performance.

4.3 Boosting to Building a Large-scale Ontology

At last, we evaluate our approach as a whole. For ontology building, we use the same labeled data sets and iteratively boost our approach. Table 8 shows that the performance of the four ontology building functions increases in each iteration. In particular, the precision and recall of Chinese `instanceOf` function goes from 65.0% and 63.0% to 96.7% and 96.9% respectively. As we can see, the performance after three iterations is excellent.

Table 8. Results of Boosting to Build the Ontology. (%)

Iteration	**`subClassOf` en**			**`subClassOf` zh**			**`instanceOf` en**			**`instanceOf` zh**		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Iteration 1.	80.8	88.2	84.4	82.0	100	90.1	87.4	97.1	92.0	65.0	63.0	64.0
Iteration 2.	87.3	91.8	89.5	91.8	98.6	95.1	93.3	98.4	95.8	91.4	89.1	90.2
Iteration 3.	87.7	93.4	**90.5**	94.8	99.3	**97.0**	97.3	99.6	**98.4**	96.7	97.0	**96.9**

P: precision, R: recall, F1: F1-measure, **en**: English, **zh**: Chinese.

In our experiments, we stop after the third iteration and successfully get two ontologies as shown in Table 9. For ontology matching, we use the same training data sets and all of the 126,221 alignments as the initial alignment set. We iteratively repeat the boosting process and 31,108 new alignments are found after 100 iterations. Due to the high computation cost, more iterations are still ongoing to find more alignments.

Table 9. Results of Built Ontology

	#Concepts	#Instances	#subClassOf	#instanceOf
English	479,040	3,520,765	751,154	11,339,698
Chinese	24,243	803,278	29,655	2,144,000

5 Related Work

Multi-lingual Ontology Building. Ontology building is to generate an ontology concerning some specific domains in the form of Resource Description Framework. Current ontology building strategies can be grouped into three categories, namely manual construction, crowdsourcing based approach [13] and open Web extraction approach. The costly manual constructed ontologies, such as WordNet, HowNet and Cyc, are relatively high-quality but usually only cover parts of facts and are costly to maintain. Crowdsourcing based approach is becoming a prevalent method for building a large-scale and regularly updated ontology. DBpedia, by making the Wikipedia machine-readable, is a representative of this approach [1]. YAGO [12], MENTA [6] and BabelNet [7] are other multi-lingual ontologies based on WordNet and Wikipedia. Zhishi.me [8] is a Chinese knowledge base by integrating Hudong Baike, Baidu Baike and Chinese Wikipedia. XLORE [16] is a multilingual ontology generated from Hudong Baike, Baidu Baike, Chinese Wikipedia and English Wikipedia. Ponzetto and Strube have proposed some methods based on connectivity in the network and lexico-syntactic matching to derive a taxonomy from Wikipedia [9]. The open Web extraction approach aims to find a wider range of knowledge in the Web. This method gives us more opportunities to harvest more knowledge, but involves more noise and need to build an ontology from scratch. Probase [17] and TextRunner [18] are representatives of open Web extraction approach. Our proposed approach is a crowdsourcing based cross-lingual ontology building method.

Cross-lingual Ontology Matching. Ontology matching is to find equivalent correspondences between semantically related entities of ontologies [4,11]. Current ontology matching strategies can be grouped into two categoies, namely heuristic-based approach and machine learning-based approach. By manually defining some weights or threshold values, such heuristic-based approaches as similarity flooding and similarity aggregation can resolve the ontology matching problem quite efficiently and effectively. RiMOM [5] is a multi-strategy ontology alignment framework. The machine learning-based approach is to learn the weights and threshold values automatically. Rong et al. have proposed a transfer learning-based binary classification approach for instance matching [10]. Wang et al. have proposed a linkage factor graph model to match the instances across heterogenous wiki knowledge bases [15]. Current cross-lingual ontology matching approaches usually employ a generic two-step method, where ontology labels are translated into the target natural language first and monolingual matching techniques are applied next [5] [14]. Wang et al. proposed a language-independent linkage factor graph model for instance matching [15]. Our proposed approach is a classification-based language-independent boosting method.

6 Conclusion and Future Work

In this paper, we propose a boosting method to build a large-scale cross-lingual ontology. The performance of ontology building and instance matching is reinforced iteratively. In particular, the performance of Chinese `instanceOf` function get a high 32% improvement for F1-measure. In our future work, we will iteratively find more cross-lingual instance alignments and crawl more Hudong Baike articles to enrich the Chinese instances. We will also improve our cross-lingual instance matching model to improve the recall, which is relatively low currently.

References

1. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: a nucleus for a web of open data. ISWC (2007)
2. Green, S., de Marneffe, M.C., Bauer, J., Manning, C.D.: Multiword expression identification with tree substitution grammars: a parsing tour de force with french. EMNLP (2011)
3. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD (2009)
4. Jean-Mary, Y.R., Shironoshita, E.P., Kabuka, M.R.: Ontology matching with semantic verification. Web Semant. (2009)
5. Li, J., Tang, J., Li, Y., Luo, Q.: Rimom: A dynamic multistrategy ontology alignment framework. TKDE (2009)
6. de Melo, G., Weikum, G.: Menta: inducing multilingual taxonomies from wikipedia. CIKM (2010)
7. Navigli, R., Ponzetto, S.P.: Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artif. Intell. (2012)
8. Niu, X., Sun, X., Wang, H., Rong, S., Qi, G., Yu, Y.: Zhishi.me: weaving chinese linking open data. ISWC (2011)
9. Ponzetto, S.P., Strube, M.: Deriving a large scale taxonomy from wikipedia. AAAI (2007)
10. Rong, S., Niu, X., Xiang, E.W., Wang, H., Yang, Q., Yu, Y.: A machine learning approach for instance matching based on similarity metrics. ISWC (2012)
11. Shvaiko, P., Euzenat, J.: Ontology matching: State of the art and future challenges. TKDE (2013)
12. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: a core of semantic knowledge. WWW (2007)
13. Tang, J., Leung, H.f., Luo, Q., Chen, D., Gong, J.: Towards ontology learning from folksonomies. IJCAI (2009)
14. Trojahn, C., Quaresma, P., Vieira, R.: A framework for multilingual ontology mapping. LREC (2008)
15. Wang, Z., Li, J., Wang, Z., Tang, J.: Cross-lingual knowledge linking across wiki knowledge bases. WWW (2012)
16. Wang, Z., Li, J., Wang, Z., Li, S., Li, M., Zhang, D., Shi, Y., Liu, Y., Zhang, P., Tang, J.: Xlore: A large-scale english-chinese bilingual knowledge graph. ISWC (2013)
17. Wu, W., Li, H., Wang, H., Zhu, K.Q.: Probase: a probabilistic taxonomy for text understanding. SIGMOD (2012)
18. Yates, A., Cafarella, M., Banko, M., Etzioni, O., Broadhead, M., Soderland, S.: Textrunner: open information extraction on the web. NAACL-Demonstrations (2007)

A Joint Embedding Method for Entity Alignment of Knowledge Bases

Yanchao Hao, Yuanzhe Zhang, Shizhu He, Kang Liu, and Jun Zhao

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
{yanchao.hao,yuanzhe.zhang,shizhu.he,kang.liu,jun.zhao}@nlpr.ia.ac.cn

Abstract. We propose a model which jointly learns the embeddings of multiple knowledge bases(KBs) in a uniform vector space to align entities in KBs. Instead of using content similarity based methods, we think the structure information of KBs is also important for KB alignment. When facing the cross-linguistic or different encoding situation, what we can leverage are only the structure information of two KBs. We utilize seed entity alignments whose embeddings are ensured the same in the joint learning process. We perform experiments on two datasets including a subset of Freebase comprising 15 thousand selected entities, and a dataset we construct from real-world large scale KBs – Freebase and DBpedia. The results show that the proposed approach which only utilize the structure information of KBs also works well.

Keywords: embeddings, multiple knowledge bases, structure information, Freebase, DBpedia

1 Introduction

As the amount of knowledge bases (KBs) accumulated rapidly on the web, the problem of how to reuse these KBs has gained more and more attention. In the real-world scenarios, many KBs describe the same entities in different ways, because KBs are distributional heterogeneous resources created by different individuals or organizations. For example, president *Barack Hussein Obama* is denoted by *m.02mjmr* in Freebase [3], while *Barack_Obama* in DBpedia [2]. Aligning such same entities could help people acquire knowledge more conveniently, as they no longer need to look up multiple KBs to obtain the full information of an entity. However, knowledge base alignment is not a trivial task, and the alignment system is often complex [8, 15]. Many traditional KB matching pipeline systems including [22, 20, 11, 7] are based on content similarity calculation and propagation.

There are some standard benchmark datasets from the Ontology Alignment Evaluation Initiative(OAEI), on which several alignment systems perform alignment algorithms. The datasets don't contain many relationships and two KBs to be aligned have common relation and property strings, which can be used

to compute content similarity to assist instances alignment. The statistics of the author-disambiguation dataset from OAEI2015 Instance Matching are as Table 1. Think about a real case, we have an entity named *m.02mjmr* refering to president *Barack Hussein Obama*, How do we align it with the entity named *Barack_Obama* in another KB with all of the relations and properties in two different encoding system? When facing the cross-linguistic or different encoding situation, what we can leverage are only the structure information of two KBs. Content information is important to KB alignment, but we think the structure information of KBs is also significant. Based on the observation above, we create two datasets including a subset of Freebase comprising 15 thousand selected entities (FB15K) and a dataset we construct from real-world large scale KBs: Freebase and DBpedia. What we try to do is to construct datasets with abundant relations and rich structure information, regardless of the content.

instance class	author-instance	relation	property
2	854	6	6

Table 1. Statistics of author-dis sandbox from OAEI2015. The relations and properties are shared in two KBs.

In this paper, we perform the KB entity alignment task by leveraging the embeddings of the KBs which are learned via the structure of KBs no matter what the content is. In previous work, KB embeddings[4, 5, 17, 6, 21, 9] are learned in order to complete the KB, and they aim at single KB. If the embedding learning method is applied on two KBs, we will obtain two independent embeddings in two different vector spaces. To represent two KBs in a uniform embedding vector space, we give some initial alignments, called seed entity alignments. In the learning process, we ensure the embeddings of the seed entities try to maintain the same. In this way, we could jointly learn the embeddings of the two KBs in a uniform embedding vector space, with two KBs connected by the seed entities "bridge". The seed alignments help learn potential alignments of the two KBs in the uniform expressive vector space via the network of the triplets. Entities with similar learned embeddings could be considered as the same entities. Thus we could find more alignments. The proposed method does not depend on manually designed rules and features, and we do not need to be aware of the content of the KBs. As a result, the proposed approach is more adaptive, could be easily utilized to large scale applications.

We conduct extensive large scale experiments on two datasets including a subset of Freebase comprising 15 thousand selected entities, denoted FB15K[5], and a dataset we construct from real-world large scale KBs – Freebase and DBpedia. The results indicate that the proposed method could achieve promising performance, and the joint embedding method only utilize the structure information of KBs, which may be a efficient supplement for KB alignment pipeline systems.

To the best of our knowledge, this is the first work to deal with the KB alignment problem using an end to end joint embedding model only utilizing the structure information of KBs. In summary, the contributions of this paper are as follows.

(1) We propose a novel model which jointly learns the embeddings of multiple KBs in a uniform vector space to align entities in KBs, only using the structure information of KBs.

(2) We construct two datasets for KB alignment task based on real-world large scale KBs: FB15K datasets and DBpedia-Freebase datasets, which have abundant relationships and rich structure information.

(3) We conduct experiments on the datasets, and the experimental results show that our approach works well.

The remainder of this paper is organized as follows. We first introduce our task in detail and overview of the related work. Then, we present the proposed method in the following section. Finally, we show the experimental results and conclude this paper.

2 Background

2.1 Task Description

Entity alignment on KBs, which is to align the entities that referring to the same real-world things, has been a hot research topic in recent years. For example, we should align the entity *m.02mjmr* in Freebase with the entity *Barack_Obama* in DBpedia. The goal of the KB alignment is to link multiple KBs effectively and create a large scale and unified KB from the top-level to enrich the KBs, which can be used to help machines understand the data and build more intelligent applications.

KBs usually use Resource Description Framework Schema(RDFS) or Ontology Web Language(OWL)or triples to describe ontology, defining elements such as"class","relation","property","instance" and so on. The research of KB alignment starts from ontology matching[23–25], mainly focusing on the semantic similarity at early time.

2.2 Related Work

Over the years, various methods have been proposed for KB alignment. Akbari et al.[1] and Suna et al.[19] utilize string-matching based methods which are quite straightforward but fail when two entity mentions are crossing languages or significantly different in literal. Joslyn et al.[10] consider the aligning problem as a graph homomorphism problem, [16, 14]exploit Instance-based techniques to align KBs, and some take the KB alignment as combinatorial optimization problems [13].

In pairs-wise alignment methods, some supervised learning methods compare vectors via property to judge an entity pair whether should be aligned

or not. This kind of technology contains decision tree[26], Support Vector Machine(SVM)[27], ensemble learning[28] and so on. Some clustering based methods[29] learns how to cluster similar entities better.

In collective alignment methods, [18]present a PARIS system based on probabilistic method to align KBs without tuning parameters and training data, but PARIS cannot handle structural heterogeneity. Lacoste et al.[12] propose SiGMa algorithm to propagate similarity via viewing the task of KB alignment as a greedy optimation problem of global match score objective function.

All of them are based on content similarity calculation and propagation, and many ontology matching pipeline systems including[22, 20, 11, 7] which participate in the OAEI 2015 Instance Matching track need to calculate content similarity. Some of them use local structure information to propagate similarity, but from another point of view, we think that the global structure information of KBs is also important. Our proposed models are based on global structure information of KBs, regardless of what the content exactly is.

3 Datasets

Because of the lack of suitable data for our task which is under the cross-linguistic or different encoding situation, we construct two datasets based on real-world large scale datasets. Firstly we present a dataset generated from FB15K, which is extracted from Freebase comprising 15 thousand selected entities. Then we illustrate the DBpedia-Freebase dataset(DB-FB), which are extracted from DBpedia and Freebase.

3.1 FB15K dataset

FB15K There are more than 2.4 billion triplets and 80 million entities in Freebase[1]. The base dataset we choose should not be too small to acquire enough overlapping part, and should not be too large to cause computational bottlenecks. As a tradeoff, we choose FB15K containing 592,213 triplets with 14,951 entities and 1,345 relationships. We randomly split them into two KBs, i.e., $kb1$ and $kb2$, with a large amount of overlapping part. Given a ratio number, i.e., the parameter *splitRatio*, we split the intersecting entities into two parts. The first part remains identical entity mention forms in two KBs, denoted as remaining part (seed alignment part). The second part keeps the entity mention forms unchanged in $kb1$, and changes the entity mention forms in $kb2$ by suffixing a certain string like "_#NEW#" to create the different entities, denoted by changing part (target alignment part), which is used for evaluation. Fig.1 indicates the splitting process of our datasets. There are two advantages of our proposed dataset. First, since they origin from the same FB15K dataset, we can control the overlapping part conveniently. Second, the gold entity alignment is known, so the evaluation is more accurate.

[1] https://developers.google.com/freebase/

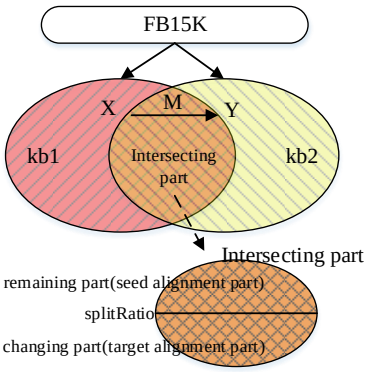


Fig. 1. The process of splitting FB15K.

3.2 DB-FB dataset

DB-FB There are more than 3 billion factual triples in DBpedia[2] and 2.4 billion in Freebase. DBpedia also provide datasets which contain triples linking DBpedia to many other datasets. Based on the given entity alignments with Freebase released on the DBpedia website[3], we can build a DBpedia-Freebase alignment dataset. Following the original intention, we intend to construct a dataset with abundant relationships and rich structure information. The dataset we construct should not be too small to contain enough structure information, and too large to cause computational bottlenecks. The steps of constructing DB-FB dataset are as follows.

step1 As we know, Freebase triples have some Compound Value Types(CVTs) to represent data where each entry consists of multiple fields. Firstly, we need to convert the triples in Freebase which contain CVT to factual triples by reducing the CVT in the preprocessing step.

step2 Then we find the triples in DBpedia and Freebase whose head and tail entity both show up in the given alignments.

step3 In the selected triples, we count the frequencies of the entity alignment pairs (take the Napierian logarithm of the product of each entity's frequency in a pair) and rank the frequencies of the entity pairs.

step4 Based on the top 10 thousand most frequently showing up entity alignment pairs, we select the triples whose head entity or tail entity are among the top 10 thousand entity alignment pairs in the picked out triples in step2.

step5 Then we make a filter to reduce the triples whose entity frequency are less than 7 in DBpedia and 35 in Freebase.[4]

The statistics of the DB-FB dataset are as Table 2.

[2] http://wiki.dbpedia.org/Downloads2015-10
[3] http://downloads.dbpedia.org/2015-10/links/freebase_links.nt.bz2
[4] In step5, 7 and 35 are empirical values chosen in experiments.

	triples	entities	relations	align_pairs
DB	515,937	57,076	373	13,932
FB	724,894	19,166	1,219	

Table 2. statistics of DB-FB dataset.

4 Methodology

Given two KBs, denoted by $kb1$ and $kb2$ respectively. The facts in both KBs are represented by triplets (h, r, t), where $h \in E$ (the set of entities) is the head entity, $t \in E$ is the tail entity, and $r \in R$ (the set of relationships) is the relationship. For example, *(Obama, president_of, USA)* is a fact. Different from previous KB embedding learning methods, our model learns the joint embeddings of the entities and the relations of two KBs. In detail, we firstly generate several entity alignments using simple strategies which leverage some extra information or other measures. As shown in Fig.2, the entities in the same color are the entity alignments, i.e., the selected seed entities. In this way, the seed entity alignments could serve as bridges between $kb1$ and $kb2$, thus we can learn the joint embeddings of both KBs in a uniform framework.

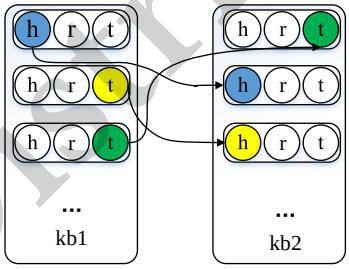


Fig. 2. Selecting seed entities in two KBs.

A KB is embedded into a low-dimensional continuous vector space while certain properties of it are preserved. Generally, each entity is represented as a point in that space while each relation is interpreted as an operation over entity embeddings. For instance, TransE[5] interprets a relation as a translation from the head entity to the tail entity. Following the energy-based framework in TransE, the energy of a triplet is equal to $d(h+r, t)$ for some dissimilarity measure d, which we take to be either the L_1 or L_2-norm. To learn such embeddings, we

minimize the margin-based objective function over the training set:

$$L = \sum_{(h,r,t)\in S} \sum_{(h',r,t')\in S'_{(h,r,t)}} \{[\gamma + d(h+r,t) - d(h'+r,t')]_+ + \lambda_1 \sum_{y\in\{h,h',r,t,t'\}} |||y||_2 - 1|\} + \lambda_2 \sum_{(e_i,e'_i)\in A} ||e_i - e'_i||_2 \quad (1)$$

where $[x]_+$ denotes the positive part of x, $\gamma > 0$ is a margin hyper-parameter, λ_1, λ_2 are ratio hyper-parameters, A is the selected seed alignments whose entities are represented by e_i in $kb1$ and e'_i in $kb2$, and

$$S'_{(h,r,t)} = \{(h',r,t)|h' \in E\} \cup \{(h,r,t')|t' \in E\} \quad (2)$$

The set of corrupted triplets, constructed according to Equation (2), is composed of training triplets with either the head or tail replaced by a random entity (but not both at the same time). The objective function is optimized by stochastic gradient descent (SGD) with mini-batch strategy . The soft constraints of the entities and relations (the λ_1 part in Equation (1)) is important because they are meaningful in preventing the training process to trivially minimize the loss function by increasing the embedding norms and shaping the embeddings[5]. The alignment part(the λ_2 part in Equation (1)) helps learn the alignment information between KBs.

Following the projection transformation idea, we can fix Equation (1) by adding a projection transformation matrix M_d:

$$L = \sum_{(h,r,t)\in S} \sum_{(h',r,t')\in S'_{(h,r,t)}} \{[\gamma + d(h+r,t) - d(h'+r,t')]_+ + \lambda_1 \sum_{y\in\{h,h',r,t,t'\}} |||y||_2 - 1|\} + \lambda_2 \sum_{(e_i,e'_i)\in A} ||M_d e_i - e'_i||_2 \quad (3)$$

The projection matrix M_d serves as the transformation of different KB vector spaces. It is more reasonable to transfer one KB vector space to another when we want to connect two KBs.

In the learning process, the embeddings of the entities in $kb1$ could become more and more similar with the same factual world entities in $kb2$ through seed entities. So the jointly learned embeddings can help improve entity alignment between the two KBs. The key of our model is to align two KBs using embeddings in a uniform space that jointly learned via the overlapping parts between the two KBs.

5 Experimental Evaluations

5.1 Baseline

Given the two KBs generated from FB15K, we suffix all the intersecting entities in $kb2$ to make $kb2$ totally different from $kb1$. Then we learn the embeddings of

the entities and relations in the two KBs in two vector space individually following TransE[5]. Since the intersecting entities are split into two parts, we use the remaining part to learn the projection transformation matrix M, representing transformation of the same entities from one vector space to the other using the following equations:

$$Y^T = MX^T \tag{4}$$

$$M = Y^T X(X^T X)^{-1} \tag{5}$$

Where X denotes the embedding matrix of the remaining part of $kb1$, Y denotes the embedding matrix of the remaining part of $kb2$, and M denotes the projection transformation matrix. Let len denote the number of entities in the remaining part, and dim denotes the dimension of the embeddings. So the matrixes X and Y are $\mathbb{R}^{len \times dim}$, while the matrix M is $\mathbb{R}^{dim \times dim}$.

As for the changing part, we could obtain the projection embeddings of the entities of $kb1$ Y in the vector space of $kb2$, using equation (4). In other words, the function of matrix M is to transform the embeddings in $kb1$'s vector space to $kb2$'s vector space in order to find the degree of similarity between the projected embeddings and the true embeddings.

In DB-FB dataset, we can directly use the Equation(4),(5) without changing the forms of the entities.

5.2 Implementation

For our model, we regard the remaining part as the seed alignment part. Some hyper-parameters in two models were just set empirically. For experiments settings, when we learn the embeddings, we choose the margin γ as 1, the dimension k as 100, the λ_1 in loss function as 0.1, the λ_2 in loss function as 1, the epoch for training as 2000. The dissimilarity measure d is L_2 distance. The embeddings of entities and relations are initialized in the range of [-0.01, 0.01] with uniform distribution. Table 3 shows the comparison of overall results where there are 7,365 entities in the target entity part for evaluation and 14,825 entities in $kb2$ totally under the parameters setting $splitRatio = 0.5$. Every entity in the target entity part could have rank value from 1 to 14,285. In this table, Mean_Rank represents the mean rank value of the target entities part, and Hits@n means the ratio number of entities that rank at top n.

Models	Mean_Rank	Hits@1	Hits@10	Hits@100
Baseline	95.97	23.96%	54.96%	83.22%
JE	94.76	29.73%	56.36%	81.91%
JEwP	**88.51**	**29.88%**	**59.21%**	**84.97%**

Table 3. Overall results of FB15K. *JE* denotes our joint Embedding model in Equation (1), and *JEwP* denotes as our joint Embedding model with projection matrix in Equation (3).

Our model improves the performance significantly compared with the baseline approach. We believe that the good performance of our model is due to jointly embedding two KBs into a uniform vector space via seed entities "bridge" connecting two KBs. The seed alignments help learn potential alignments of the two KBs in the uniform expressive vector space via the triplets' network, while in the baseline model, we can only utilize the projection transformation matrix learned from the seed alignment part with no extended alignment information on the whole.

Models	splitRatio	Mean_Rank	Hits@1	Hits@10	Hits@100
Baseline	0.1	91.79	25.10%	56.52%	83.84%
	0.3	92.71	23.34%	54.25%	82.95%
	0.5	95.97	23.96%	54.96%	83.22%
	0.7	94.44	25.12%	55.66%	83.10%
JE	0.1	352.00	10.25%	20.19%	47.18%
	0.3	239.56	15.47%	31.63%	63.30%
	0.5	94.76	29.11%	56.62%	81.91%
	0.7	97.85	29.73%	56.36%	81.48%
JEwP	0.1	205.74	17.59%	42.34%	66.67%
	0.3	123.28	25.63%	55.35%	78.60%
	0.5	88.51	29.88%	59.21%	84.97%
	0.7	**86.83**	**30.38%**	**60.70%**	**85.14%**

Table 4. Effect of *splitRatio* on FB15K.

We also explore the effect of *splitRatio*, i.e., the number of seed entities, on our models. As shown in Table 4, along with the ascending order of *splitRatio*, the Mean_Rank value of our model decreases and the Hits@n increases, indicating the performance of our model getting better because of more seed entities. While the baseline model shows much more placid when the *splitRatio* increases, as shown in Figure3. The impression of the baseline model is that the performance should be increasing along with the ascending order of *splitRatio* because there are more and more data to learn the projection transformation matrix M well. But the result is almost placid. The reason in further analysis shows that when $splitRatio = 0.1$ the categories of the entities in the remaining part to learn are already covered enough and the projection transformation based method cannot depict the influence of different relations to the entity alignment. While our joint embedding method learns the different representations of different relations which help improve the performance of alignment. For example, the relation "son_of" is more important than the relation "nationality" in judging whether two entities are the same or not.

We conduct experiments on the DB-FB dataset, and the results are as Table 5. The baseline model has better $Mean\_Rank$, and our joint embedding projection model has better performance at $Hits@n$ when we have a certain number of seed Alignments. The reason may be that the baseline model learns the pro-

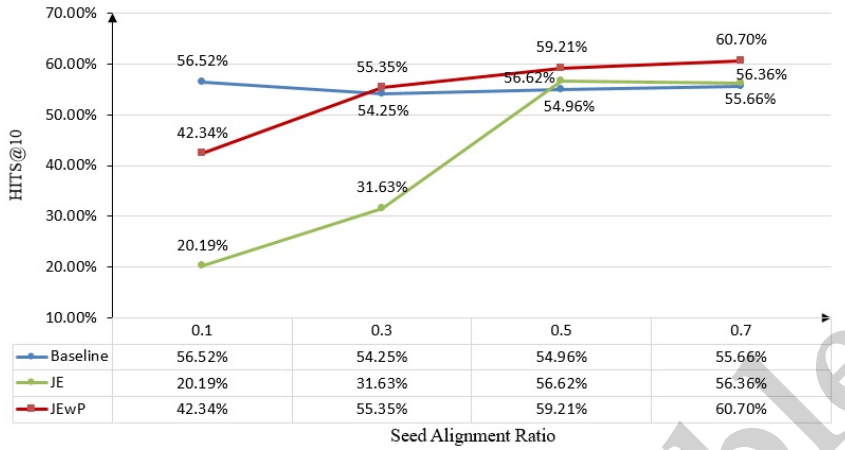


Fig. 3. The performance of our models on FB15K along with the ascending *splitRatio*.

jection transformation matrix from a global perspective, while our models learn the embeddings of KBs and projection matrix M_d (especially the JEwP model) in the iterative optimization process. The DB-FB dataset is relatively large and the selected DBpedia set which has 515,937 triples and 57,076 entities is more sparse than the selected Freebase set which has 724,894 triples and 19,166 entities. So on the DB-FB dataset, it may be more difficult to capture the global accurate alignment information for our models in the learning process. Note that our models only utilize the structure information of KBs to align entities, not the accurate content information. When we are faced with actual KB alignment task, our model may be an efficient supplement to the alignment pipeline systems.

Models	SeedAlignments_Ratio	Mean_Rank	Hits@1	Hits@10	Hits@100
Baseline	0.1	554.43	2.20%	14.56%	45.81%
	0.3	485.46	2.00%	14.85%	46.46%
	0.5	490.23	2.18%	14.76%	48.11%
	0.7	**476.25**	2.20%	14.81%	49.13%
JE	0.1	1019.98	0.66%	4.46%	27.06%
	0.3	785.63	1.25%	8.55%	35.67%
	0.5	723.18	1.57%	11.35%	38.68%
	0.7	700.00	1.87%	13.15%	41.40%
JEwP	0.1	639.89	1.91%	9.86%	41.72%
	0.3	605.39	2.38%	15.18%	45.65%
	0.5	524.74	3.91%	19.39%	53.34%
	0.7	510.18	**4.64%**	**19.90%**	**54.89%**

Table 5. Results on the DB-FB dataset.

6 Conclusions

We propose a model which jointly learns the embeddings of KBs in a uniform vector space via seed entity alignments to align KBs. Generally, our model with projection matrix has better performance than our model without projection matrix, which is reasonable for that projection matrix indicates transformation of KBs, and projection matrix should be added when we associate one vector space with another. To utilize structure information of KBs, we construct two datasets including FB15K and DB-FB based on real-world large scale KB. The experimental results show that the proposed approach which only utilize the structure information of KBs also works well, and may be an efficient supplement for KB alignment pipeline systems.

7 Acknowledgement

This work was supported by the Natural Science Foundation of China (No. 61533018), the National Basic Research Program of China (No. 2014CB340503) and the National Natural Science Foundation of China (No. 61272332). And this work was also supported by Google through focused research awards program.

References

1. Ismail Akbari, Mohammad Fathian, and Kambiz Badie. 2009. An improved mlma+ and its application in ontology matching. In Innovative technologies in intelligent systems and industrial applications, 2009. CITISIA 2009, pages: 56 – 60. IEEE.
2. Soren Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. Springer.
3. Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim S- turge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD international conference on Management of data, pages: 1247 – 1250. ACM.
4. Antoine Bordes, Jason Weston, Ronan Collobert, and Yoshua Bengio. 2011. Learning structured embeddings of knowledge bases. In Conference on Artificial Intelligence, number EPFL-CONF-192344.
5. Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. InAdvancesinNeuralInformationProcessingSystems, pages: 2787 – 2795.
6. Kai-Wei Chang, Wen-tau Yih, and Christopher Meek. 2013. Multi-relational latent semantic analysis. In EMNLP, pages: 1602 – 1612.
7. Syrine Damak, Hazem Souid, Marouen Kachroudi, and Sami Zghal. 2015. Exona results for oaei 2015.
8. Chaitanya Gokhale, Sanjib Das, AnHai Doan, Jeffrey F Naughton, Narasimhan Rampalli, Jude Shavlik, and Xiaojin Zhu. 2014. Corleone: Hands-off crowdsourcing for entity matching. In Proceedings of the 2014 ACM SIGMOD international conference on Management of data, pages: 601 – 612. ACM.
9. Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge graph embedding via dynamic mapping matrix. In Proceedings of ACL, pages: 687 – 696.

10. Cliff A Joslyn, Patrick Paulson, Amanda White, and Sinan al Saffar. 2009. Measuring the structural preservation of semantic hierarchy alignments. In Proceedings of the 4th International Workshop on Ontology Matching. CEUR Workshop Proceedings, volume 551, pages: 61 – 72. Citeseer.
11. Abderrahmane Khiat and Moussa Benaissa. 2015. Insmt+ results for oaei 2015 instance matching.
12. Simon Lacoste-Julien, Konstantina Palla, Alex Davies, Gjergji Kasneci, Thore Graepel, and Zoubin Ghahramani. 2013. Sigma: Simple greedy matching for aligning large knowledge bases. In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, pages: 572 – 580. ACM.
13. Natalia Prytkova, Gerhard Weikum, and Marc Spaniol. 2015. Aligning multicultural knowledge taxonomies by combinatorial optimization. In Proceedings of the 24th International Conference on World Wide Web Companion, pages: 93 – 94. International World Wide Web Conferences Steering Committee.
14. R Pushpakumar, Tiruchirappalli Srirangam, India Dr M Sai Baba, N Madurai Meenachi, and P Balasubra- manian. 2016. Instance based matching system for nuclear ontologies.
15. Francois Scharffe, Ondrej Zamazal, and Dieter Fensel. 2014. Ontology alignment design patterns. Knowl- edge and information systems, 40(1): 1 – 28.
16. Md Seddiqui, Rudra Pratap Deb Nath, Masaki Aono, et al. 2015. An efficient metric of automatic weight generation for properties in instance matching technique. arXiv preprint arXiv:1502.03556.
17. Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In Advances in Neural Information Processing Systems, pages: 926 – 934.
18. Fabian M Suchanek, Serge Abiteboul, and Pierre Senel- lart. 2011. Paris: Probabilistic alignment of relations, instances, and schema. Proceedings of the VLDB Endowment, 5(3): 157 – 168.
19. Yufei Suna, Liangli Maa, and Shuang Wangb. 2015. A comparative evaluation of string similarity metrics for ontology alignment.
20. Wenyu Wang and Peng Wang. 2015. Lily results for oaei 2015.
21. Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In AAAI, pages: 1112 – 1119. Citeseer.
22. Yan Zhang and Juanzi Li. 2015. Rimom results for oaei 2015.
23. Shvaiko P, Euzenat J. Ten Challenges for Ontology Matching[C]. Proceedings of the Move to Meaningful Internet Systems. Berlin: Springer,2008: 1164 – 1182.
24. Berstein P A, Madhavan J, Rahm E. Generic schema matching, ten years later[J]. Proceedings of the VLDB Endowment, 2011, 4(11): 695 – 701.
25. Shvaiko P, Euzenat J. Ontology matching: State of the art and future challenges[J]. IEEE Trans on Knowledge & Data Engineering, 2013,25(1): 158 – 176.
26. Han J W, Kambe M. Data Mining: Concepts and Techniques[M]. San Francisco, CA: Morgan Kaufmann,2006.
27. Vapnik V. The Nature of Statistical Learning Theory[M]. Berlin: Springer, 2000.
28. Kantardzic M. Data Mining[M]. Hoboken, NJ: John Wiley & Sons, 2011: 235 – 248.
29. Cohen W W, Richman J. Learning to match and cluster large high-dimensional data sets for data integration[C]. Proceddings of Advances in Neural Information Processing Systems. Cambridge, MA: MIT Press, 2005: 905 – 912.

LD2LD: Integrating, Enriching and Republishing Library Data as Linked Data

Qingliang Miao[1], Ruiyu Fang[1], Lu Fang[1], Yao Meng[1],
Chenying Li[2], Mingjie Han[2], Yong Zhao[2]

[1] Fujitsu R&D Center CO., LTD.
100027, Chaoyang District, Beijing P. R. China
{qingliang.miao, fangruiyu, fanglu,
mengyao}@cn.fujitsu.com
[2] China Agricultural University,
100083, Haidian District, Beijing P. R. China
{licy, hanmj,zhaoyong}@cau.edu.cn

Abstract. The development of digital library increases the need of integrating, enriching and republishing library data as Linked Data. Linked library data could provide high quality and more tailored service for library management agencies as well as for the public. However, even though there are many data sets containing metadata about publications and researchers, it is cumbersome to integrate and analyze them, since the collection is still a manual process and the sources are not connected to each other upfront. In this paper, we present an approach for integrating, enriching and republishing library data as Linked Data. In particular, we first adopt duplication detection and disambiguation techniques to reconcile researcher data, and then we connect researcher data with publication data such as papers, patents and monograph using entity linking methods. After that, we use simple reasoning to predict missing values and enrich the library data with external data. Finally, we republish the integrated and enriched library data as Linked Data.

1 Introduction

Libraries are experiencing a time of huge, tumultuous change. With the rapid development of digital libraries, library management agencies and users are faced with an increasing amount of publications. The huge amount and not interconnected nature of publications challenges library management agencies and users on managing and accessing scientific information. On the one hand, users demand intelligent search services to discover interested publications. On the other hand, library management agencies need to incorporate semantic information to better organize their digital assets and make publications more discoverable. For example, many libraries maintain data on researchers, papers and other materials, and separate search systems are built for each of these data sets [1]. As data is so distributed and heterogeneous, there is not a single search engine that can effectively retrieve a comprehensive set of the re-

sources, e.g. find all the papers related to a given author within a given time period. Libraries have all been exploring new approaches to dramatically improve the discovery experience for users seeking scholarly information resources, such as traditional monograph and journal publications, archival materials, web archives, and much more [2].

Moreover, researchers are duplicated and ambiguous. One researcher may have different mentions (names) that distributed in different data sets, while different researchers may have the same name. Therefore, disambiguating and detecting duplicated researchers are necessary. If we can detect duplicated and ambiguous researchers, library management agencies and users can use the library data more efficiently. Since library data covers many elements such as papers, patents, discipline and organizations, it contains a large ration of missing values in its data sets. The impact of missing values is even aggravated when combining different data sets. The missing values makes library data harder to integrate and link. Consequently, missing value complement and data enrichment are important.

The Semantic Web in general and the Linked Data[1] initiative in particular encourage institutions to publish, share and interlink their data. This has considerable potential for libraries, which can complement their data by linking it to other external data sources. The Linked Data technology meets the need of connecting distributed data silos across the web. The Linked Data is based on a set of principles created by W3C[2]. The primary data model of Linked Data is the Resource Description Framework (RDF)[3], under which each resource in Linked Data space is identified by a unique HTTP dereferenceable Uniform Resource Identifier (URI) and the relations of resources are described with simple subject-predicate-object triples. Based on these principles, resources are linked by relations, and sophisticated networks of Linked Data can be built.

In this paper, we present the first effort to work on integrating, enriching and republishing library data as Linked Data. More specifically, we adopt Linked Data technology to integrate library data that wasn't previously linked. We first use hierarchical clustering method to conduct duplicated detection and disambiguation for researchers. And then, we link researchers with other library data such as monograph, journal publications, archival materials, research results, images and recordings. After that, we enrich library data by predicting missing values and republish library data as Linked Data. Our contributions are:

- We analyze and integrate several data sources including library data, DBpedia, Zhishi.me.
- We provide a system architecture for transforming library data into Linked Data including data cleaning, data integration, data enrichment and republishing.
- We use reasoning method to predict missing values and enrich the library data with external data.

[1] http://linkeddata.org/home

[2] http://www.w3.org/

[3] http://www.w3.org/RDF/

- We develop a system[4] providing semantic search, statistical analysis and visualization based on linked library data.

The remainder of the paper is organized as follows. In the next section we review the related literature on linked library data. In the third section, we introduce the Chinese Agriculture University (CAU) Library data. We introduce the approach in detail and present the results in the fourth section. Last, we conclude the paper with a summary of our work and point out future directions.

2 Related Work

There are three related research field to our work. They are person disambiguation, entity linking and property alignment in the following subsections respectively.

2.1 Person Disambiguation

Previous work usually uses clustering techniques to solve person disambiguation issues. Christof Monz and Wouter Weerkamp [3] introduce a clustering approach to person name disambiguation. Minoru Yoshida et al., [4] propose to use a two-stage clustering algorithm by bootstrapping to improve person disambiguation performance, and they use named entities, compound key words, and URLs as features for similarity calculation. Jian Xu et al., [5] present a new key-phrased clustering method combined with a classification to improve cluster performance. Silviu Cucerzan [6] proposes a name entity disambiguation method through a process of maximizing the agreement between the contextual information extracted from Wikipedia and the context of a document, as well as the agreement among the category tags associated with the candidate entities. More recently, researchers combine traditional disambiguation methods with Linked Data knowledge for entity disambiguation. For example, Danica Damljanovic and Kalina Bontcheva [7] combine a state-of-the-art entity disambiguation tool with novel Linked Data-based similarity measures and show that the combined algorithm can improve disambiguation accuracy. Ricardo Usbeck et al., [8] propose a novel knowledge-base-agnostic approach for named entity disambiguation. Their approach combines the Hypertext-Induced Topic Search (HITS) algorithm with label expansion strategies and string similarity measures.

2.2 Entity Linking

Entity linking has attracted more and more attentions from both academia and industry. For example, Mihalcea and Csomai [9] propose Wikify system to annotate text using Wikipedia. Milne and Witten [10] implement a similar system called Wikipedia Miner, which adopts supervised disambiguation approach using Wikipedia hyperlinks as training data. Han and Sun [11] leverage entity popularity and context knowledge for

[4] http://36.110.45.42:3333/

entity linking. In practical applications, TagMe [12] system adopts a collective disambiguation approach, which computes agreement score of all possible bindings, and uses heuristics to select best target. DBpedia Spotlight [13] is a system for automatically annotating text with DBpedia. One important feature of the system is that it allows users to configure the annotations through the DBpedia ontology and quality measures such as prominence, topical pertinence, contextual ambiguity and disambiguation confidence. The disambiguation model of Illinois Wikifier [14] is based on weighted sum of features such as textual similarity and link structure. AIDA [15] is a robust system based on collective disambiguation exploiting the prominence of entities, context similarity between the mention and its candidates, and the coherence among candidate entities for all mentions.

2.3 Property Alignment

Since different data sets may use different properties, property alignment should be conducted. Property alignment is related to schema matching and ontology matching. Falcon-AO [16], Logmap [17] RiMOM [18], and PARIS [19] are ontology matching tools for the automatic alignment of instances, properties and classes from different ontologies. These tools reach satisfactory results in the recent OAEI evaluation. Different from traditional ontology alignment settings, in this study, domains and ranges of properties are not provided. Worse still, some object values are missing. Lack of such ontological knowledge, these tools fail to conduct property alignments.

3 Data Sources

In this study, we use CAU library data. The CAU library data set contains data ranging from 1980 to 2015 and it contains 108340 entities in 10 isolated data sets. The statistics of CAU library data is shown in Table 1. Our goal is to integrate these 10 isolated data sets, enrich these data semantically, and republish them as Linked Data.

Table 1. The statistics of CAU library data

Data set	#Instance	#Property
Researcher	5863	51
SCI Indexed Journal Paper	11934	42
Chinese Journal Paper	48449	52
Thesis	32755	41
Patent	2389	24
Project	3941	31
Monograph	1572	61
Research results	536	30
Curriculum	410	42
Organization	312	46
Discipline	179	52

CAU library data has a large proportion of missing values. Due to the page limit, we only shows the statistics of instances missing discipline and affiliation values in Table 2.

Table 2. The statistics of missing value in CAU library data

Data set	# Instance	#NoDiscipline	#NoAffiliation
SCI Indexed Journal Paper	11934	48	959
Chinese Journal Paper	48449	27104	12407
Thesis	32755	673	5691
Patent	2389	1	2389
Project	3941	1677	2751
Monograph	1572	1150	1572
Research results	536	3	536
Curriculum	410	10	55

In this study, we use simple reasoning method to predict missing values as detailed in section 4.5. Besides missing values, we enrich CAU library data by linking it with external knowledge base e.g. DBpedia [20] and Zhishi.me [21] as well.

DBpedia, initially released in 2007, is an effort to extract structured data from Wikipedia and publish the data as Linked Data. Zhishi.me is the first effort to publish large scale Chinese semantic data and link them together as a Chinese LOD (CLOD). Zhishi.me derives important structural features in three largest Chinese encyclopedia sites (i.e., Baidu Baike, Hudong Baike, and Chinese Wikipedia) and proposes several data-level mapping strategies for automatic link discovery. At present, the CLOD has more than 5 million distinct entities.

DBpedia and Zhishi.me could supply more information for instances in CAU library data. For example, when linking researcher with DBpedia and Zhishi.me, more information can be obtained such as nationality, birthday, birthplace, research field and awards. When linking organization instance with DBpedia and Zhishi.me entity, more information can be obtained, such as past name, launch date, longitude, latitude, homepage. Moreover, linking research topic with DBpedia and Zhishi.me entity, we can obtain category information by "dc:subject" relation and other mentions by DBpedia redirection relation.

4 The Approach

In this section, we will first illustrate the system architecture of the proposed approach, and then introduce how to integrate and link these data silos into Linked Data, and how to enrich the Linked Data with external knowledge base.

4.1 System Architecture

Figure 1 shows the system architecture of the proposed approach. The inputs are structured data in CSV or XML format and unstructured text and html data, and the outputs are linked library data. The approach includes five main modules: (1) duplication detection and disambiguation; (2) data linkage; (3) ontology design; (4) data enrichment and (5) data republish.

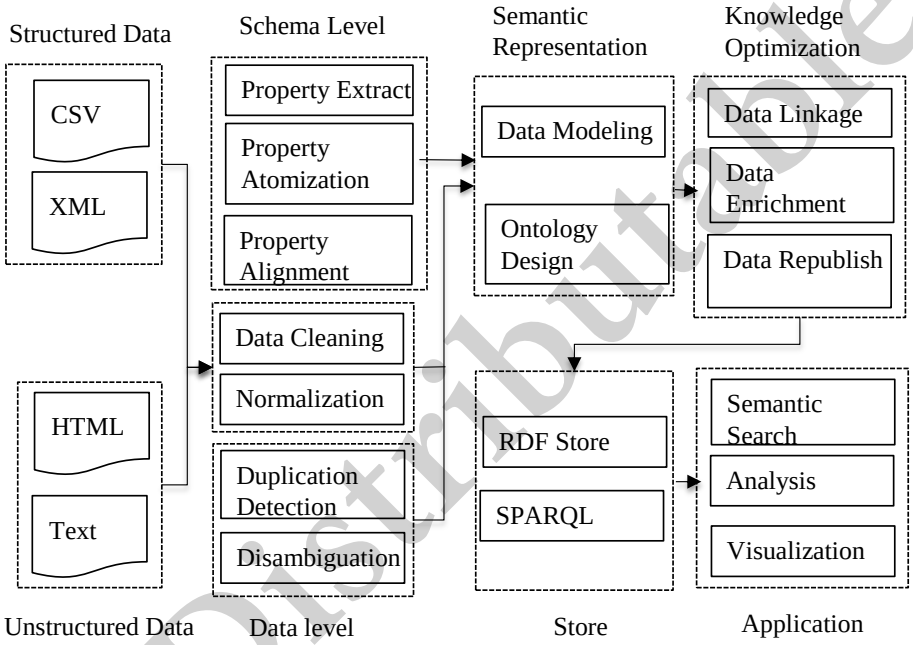


Fig. 1. LD2LD system architecture

Firstly, the input data is preprocessed at both schema and data level. Schema lever preprocessing includes property extraction, atomization and alignment. Some properties in original data is non-atomized, for example, some properties indicate time period information, therefore, they should be separated into two properties indicating starting date and ending date respectively. Some properties including time modifier, e.g. "2010 PhD entrance examination subjects" should be separated as well. Data level preprocessing includes data cleaning and normalization. For example, there are more than 30 different time expressions in CAU library data, therefore, we prepare specific normalization rules for each time expression. Besides time expression, we develop normalization rules for currency as well. If a string value contains any delimiter, the value is segmented into different parts by the delimiter and each segment will be assigned a type. For example, "BEIJING AGR UNIV, COLL ANIM SCI & TECHNOL, BEIJING 100094, PEOPLES R CHINA" will be segmented as "BEIJING AGR

UNIV", "COLL ANIM SCI & TECHNOL", "BEIJING 100094", "PEOPLES R CHINA" and assigned types "University", "College", "Address", "Country". Since CAU library data was created and managed by different agents, they may use different properties to represent the same thing. For example, property "学科专业" is used in SCI Indexed Journal Paper data set, while "相关一级学科" and "相关二级学科" are used in Chinese Journal Paper data set. Therefore, we need to conduct property alignment.

After preprocessing the input data, we conduct duplication detection and disambiguation for researchers and assign a URI for each researcher. This URI is essential for the integration and enables to link researcher with other publication data such as journals, patents and monograph in data linkage module; Based on data linkage results, we design ontology to represent the integrated data. After that, we use simple reasoning to predict missing values and enrich the library data with external data in data enrichment module. More specifically, we link researcher, organization, keywords with DBpedia and Zhishi.me. Finally, we republish the library data into Linked Data. Following sections will introduce each module in detail.

4.2 Duplication detection and disambiguation

We treat researcher disambiguation as a clustering problem. We select several features to disambiguate those researchers with same name. The similarity score of two feature vectors are calculated using VSM (vector space model). We adopt hierarchical clustering method to do researcher disambiguation.

The features as follows:

(1). Affiliations of researcher, include college and department of researcher.

(2). Research field of researcher, which can be derived from discipline, curriculum and specialty.

(3). Graduate school of the researcher.

We give different feature weights to those features based on their discriminating degrees for disambiguation. More specifically, we treat affiliation feature contributes more to disambiguate two researchers with same name than other types of feature. If several same named researchers hold the same affiliations, we prefer identifying them as the same person. And features are combined using pre-defined weights, we try different groups of feature weights and select the one with best performance. Given two feature vectors $P_1 = (a_{11}, a_{12}, ..., a_{1n})$ and $P_2 = (a_{21}, a_{22}, ..., a_{2n})$, a_{ij} takes the value of 0 or 1, which stands for whether the feature condition is met. Meanwhile We define a group of feature weights $W = (w_1, w_2, ..., w_n)$, $\sum_{i=1}^{n} w_i = 1$. And the similarity score of two researcher vectors is computed using formula (1):

$$sim(P_1, P_2) = \sum_i w_i a_{1i} a_{2i} \quad \textbf{(1)}$$

During hierarchical clustering, to decide which clusters should be combined, we adopt the average linkage criterion as in formula (2).

$$csim(c_1, c_2) = \frac{1}{|c_1||c_2|} \sum_{p_i \in c_1} \sum_{p_j \in c_2} sim(p_i, p_j) \quad (2)$$

There are 5863 researchers in original CAU library data, after disambiguation and duplicated detection, we get 5583 researchers. We find 297 different researchers with same name and 130 duplicated records of 65 researchers. We conduct a preliminary experiment to evaluate the duplicated detection and disambiguation results, and the accuracy of hierarchical clustering method is 98%.

4.3 Data Linkage

After researcher disambiguation and duplicated detection, we link researchers to their archive i.e. SCI indexed journal papers, Chinese journal papers, theses, monographs, curriculums, patents, projects and research results. Data linkage, however, can be non-trivial due to the researcher ambiguity and name variation issues. The researcher ambiguity issue means that a mention could refer to multiple researchers in different data sets. Name variation indicates that an entity may be mentioned in different ways such as official name, nickname, aliases, abbreviation or even misspellings. For example, researcher names of SCI papers are usually written in abbreviated form. Therefore, cross-lingual data linkage is more complicated due to the cross-lingual ambiguity. To solve these issues, we extract rich features from both researcher profiles and their archive, and compute the similarity of two feature sets, and link two resources if their similarity score is greater than a threshold.

Since SCI papers are written in English, meanwhile the researcher profiles are in Chinese form. To solve the cross-lingual linking issues, we develop a cascaded linking method. More specifically, we first link resources (researcher profiles and archive) in the same language. Then we enrich the researcher profile feature sets by adding new features extracted from the linking results obtained in the first step. As a result, researcher profile feature set is enriched. Then we translate the enriched feature set into English: 1. we translate the coauthor names into English, in both complement and abbreviation forms. 2. We translate the publication titles and keywords into English. After feature set translation, we conduct mono-lingual linking using the method described above. We also use a self-training strategy by iteratively adding confident features into the researcher feature sets during linking.

To evaluate the data linkage performance, we manually annotate 10 researchers and their archive as the test data. Table 3 shows the experiment results.

Table 3. The experiment result of data linkage

Data set	Precision	Recall	F1-measure
SCI Indexed Journal Paper	0.989	1.0	0.994
Chinese Journal Paper	1.0	1.0	1.0
Thesis	0.994	1.0	0.997
Patent	1.0	1.0	1.0
Project	1.0	1.0	1.0
Research results	1.0	1.0	1.0

4.4 Ontology Design

Selecting established ontologies as the basis for data modeling is strongly suggested in the semantic web community, since it makes the published data easier to share and exchange. Consequently, we aimed to do that as well. In practice however, we had to realize that existing ontologies are only partially suitable to model our data. Individual properties had definitions that did not match our data sets, so that no single ontology was found acceptable. Instead, we had to meticulously determine a set of ontologies whose parts would together cover most of our data. For the remaining portions we defined our own properties, with the intent to register the resulting ontology in the future.

The data modeling for the representation of researcher and publication utilizes several existing ontologies like the FOAF vocabulary and the Relationship Vocabulary. For subject headings the data modeling is based on the use of the Simple Knowledge Organization System (SKOS) and Dublin Core elements. We use 34 established properties and defined 250 properties ourselves. Table 4 lists the established ontologies we used.

Table 4. The established ontologies we used

Ontology	namespace
dbo	http://dbpedia.org/ontology/
dcterms	http://purl.org/dc/terms/
foaf	http://xmlns.com/foaf/0.1/
iscover	http://i-scover.ieice.org/terms/iscover#
prism	http://prismstandard.org/namespaces/basic/2.0/
schema	http://schema.org/
skos	http://www.w3.org/2004/02/skos/core#
swrc	http://swrc.ontoware.org/ontology#
vcard	http://www.w3.org/2006/vcard/ns#

4.5 Data Enrichment

Data enrichment includes two steps, one is predicting missing values and the other one is link researcher, organization and keywords with DBpedia and Zhishi.me. For missing value prediction, we use a simple reasoning based method. More specifically, we use following rules to predict discipline and affiliation values. If the author of publication P is R, and author R's affiliation is A, then publication's affiliation is A. If the author of publication P is R, and author R's discipline is D, then publication's discipline is D. Table 5 shows the discipline and affiliation value complement results.

<P author R> $\cap$ <R affiliation A> $\Rightarrow$ <P affiliation A>

<P author R> $\cap$ <R discipline D> $\Rightarrow$ <P affiliation D>

Table 5. The results of value complement in CAU library data

Data set	#Instance	#AddDiscipline	#AddAffiliation
SCI Indexed Journal Paper	11934	9	875
Chinese Journal Paper	48449	12376	10927
Thesis	32755	567	5342
Patent	2389	1	936
Project	3941	756	2251
Monograph	1572	372	1238
Research results	536	2	498
Curriculum	410	8	52

For researcher, organization and keyword linkage with DBpedia and Zhishi.me, we first conducts character and punctuations normalization, and then use normalized entity name as query to retrieval all the candidates from DBPedia and Zhishi.me. In order to obtain more accurate candidates, we conduct link analysis for each candidate. Specifically, if a candidate A has a redirect entity B, we add entity B into candidate set. If a candidate A is ambiguous, we add all the entities that candidate A may refer to into candidate set. After that, we use a ranking model that combines lexical and semantic similarity to determine which candidate should be linked. Specifically, we computes the string similarity between entity and each candidate using Levenshtein and Jaccard similarity. Semantic similarity is computed using semantic profiles. For organization, we use type and location information. For researcher, we use type, affiliation and research field. For keyword linkage, we use related keywords.

4.6 Republishing as Linked Data

The resources and properties in the library data namespace are published according to the Linked Data principles. The ontology contains all library data properties and class descriptions. Each resource is assigned a dereferenceable URI. The CAU linked library data includes 106109 resources in 10 classes, and 5826579 triples. We provide SPARQL endpoint at http://36.110.45.46:8890/sparql.

5 Conclusions and Future Work

In this paper we have presented an approach for integrating, enriching and republishing library data as Linked Data from several data sources including CAU library data, DBpedia and Zhishi.me. We have developed several components including a data cleaning, duplication detection and disambiguation, entity linkage and missing value prediction module. The linked library data includes 106109 resources in 10 classes, and 5826579 triples. A system with semantic search, statistic and visualization function is developed as well. We also conduct preliminary experiments and the results indicate the approach is effective.

Our future work include extensions of the presented data sets, methods, and the system itself. We plan to predict more missing values based on more sophisticated semantic reasoning methods. Cross-lingual data integration, e.g. linking English papers with researchers is another research direction.

References

1. Nobuyuki Igata, Fumihito Nishino, Terunobu Kume and Takahide Matsutsuka.: Information Integration and Utilization Technology using Linked Data. FUJITSU Sci. Tech. J., Vol. 50, No. 1, pp. 3--8 (2014)
2. Dean B. Krafft.: Linked Data for Libraries: A Project Update. In: 14th International Semantic Web Conference, United States of America, Bethlehem, pp.11--15 (2015)
3. Christof Monz, Wouter Weerkamp.: A Comparison of Retrieval-based Hierarchical Clustering Approaches to Person Name Disambiguation. In: 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 65--651 (2009)
4. Minoru Yoshida, Masaki Ikeda, Shingo Ono, Issei Sato, Hiroshi Nakagawa.: Person Name Disambiguation by Bootstrapping, In: 33th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 10--17 (2010)
5. Jian Xu, Qin Lu,Zhengzhong Liu.: Combining Classification with Clustering for Web Person Disambiguation, In: 21st International Conference on World Wide Web, pp. 637--638 (2012)
6. Silviu Cucerzan.: Large-Scale Named Entity Disambiguation Based on Wikipedia Data, In: 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 708--716 (2007)
7. Danica Damljanovic and Kalina Bontcheva.: Named Entity Disambiguation using Linked Data, In: 9th Extended Semantic Web Conference, (2012)
8. Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, Michael Röder, Daniel Gerber, SandroAthaide Coelho, Sören Auer, and Andreas Both.: AGDISTIS - Graph-Based Disambiguation of Named Entities Using Linked Data, In: 13th International Semantic Web Conference, (2014)
9. Mihalcea, R., and Csomai.: A. Wikify! Linking Documents to Encyclopedic Knowledge. In: 17th ACM Conference on Information and Knowledge Management, pp. 233--242 (2007)
10. Milne, D., and Witten, I. H.: Learning to Link with Wikipedia. In: 17th ACM Conference on Information and Knowledge Management, pp. 509--518 (2008)
11. Han, X. P., Sun, L.: A Generative Entity-Mention Model for Linking Entities with Knowledge Base. In: 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, pp. 945--954 (2011)
12. Ferragina, P., Scaiella, U.: TAGME: On-the-fly Annotation of Short Text Fragments. In: 19th ACM International Conference on Information and Knowledge Management, pp. 1625--1628 (2010)
13. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: DBpedia Spotlight: Shedding Light on the Web of Documents. In: 7th International Conference on Semantic Systems, pp. 1--8 (2011)
14. Ratinov, L., Roth, D.: Design Challenges and Misconceptions in Named Entity Recognition. In: 13th Conference on Computational Natural Language Learning, pp.147--155 (2009)
15. Yosef, M.A., Hoffart, J., Bordino, I., Spaniol, M., Weikum, G.: AIDA: an Online Tool for Accurate Disambiguation of Named Entities in Text and Tables. In: PVLDB'11, pp. 1450--1453 (2011)

16. Hu, W., Qu, Y., Cheng, G.: Matching Large Ontologies: A Divide-and-Conquer Approach. Data & Knowledge Engineering 67(1), pp. 140--160 (2008)
17. Jimenez-Ruiz, E., Grau, B.C., Zhou, Y.: Logmap 2.0: towards Logic-based, Scalable and Interactive Ontology Matching. In: Ontology Matching, pp. 45--46 (2011)
18. Li, Y., Li, J.Z., Zhang, D., Tang, J.: Result of Ontology Alignment with RiMOM at OAEI'06. In: Ontology Matching. (2006)
19. Suchanek, F.M., Abiteboul, S., Senellart, P.: PARIS: Probabilistic Alignment of Relations, Instances, and Schema. PVLDB 5(3), pp. 157--168 (2011)
20. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - A crystallization point for the web of data. J. Web Sem. pp. 154--165 (2009)
21. Xing Niu, Xinruo Sun, Haofen Wang, Shu Rong, Guilin Qi, Yong Yu.: Zhishi.me - Weaving Chinese Linking Open Data, In Proceedings of 10th International Semantic Web Conference, Bonn, pp. 23--27 (2011)

Large Scale Semantic Relation Discovery: Toward Establishing the Missing Link between Wikipedia and Semantic Network

Xianpei Han, Xiliang Song, Le Sun
State Key Laboratory of Computer Sciences,
Institute of Software, Chinese Academy of Sciences
100190 Beijing, China
{xianpei, xiliang, sunle}@nfs.iscas.ac.cn

Abstract. Wikipedia has been the largest knowledge repository on the Web. However, most of the semantic knowledge in Wikipedia is documented in natural language, which is mostly only human readable and incomprehensible for computer processing. To establish the missing link from Wikipedia to semantic network, this paper proposes a relation discovery method, which can: 1) discover and characterize a large collection of relations from Wikipedia by exploiting the *relation pattern regularity*, the *relation distribution regularity* and the *relation instance redundancy*; and 2) annotate the hyperlinks between Wikipedia articles with the discovered semantic relations. Finally we discover *14,299* relations, *105,661* relation patterns and *5,214,175* relation instances from Wikipedia, and this will be a valuable resource for many NLP and AI tasks.

Keywords: semantic network, relation discovery, knowledge acquisition

1 Introduction

A long-standing goal of natural language processing (NLP) and artificial intelligence (AI) is to build large-scale, machine-readable knowledge base (KB) which can support natural language understanding and human-like reasoning. To achieve this goal, a continuum of research, from the manual construction to the automatic information extraction, have been devoted to the knowledge base construction. In its early stages, researchers attempted to build KBs by manually collecting common sense knowledge. Several most notable examples include, *WordNet* (Miller, 1995), *FrameNet* (Baker et al., 1998) and *OpenCyc* (Matuszek et al., 2006). These manually constructed methods, however, require too much manual engineering and are not suitable for constructing high-coverage and up-to-date KBs which fit to the real world usage.

To overcome the limitations of manually constructed KBs, there have been many research efforts devoted to the fully automatic open information extraction (Open IE) techniques, which extract facts (i.e., relational tuples such as *Headquarters-In*(*Armonk*, *IBM*)) from a large corpus or web in a *Bootstrapping* or *self-learning* way. Several notable examples include *DIPRE* (Brin, 1999), *Snowball* (Agichtein, Eugene & Gravano, 2000), *KnowItAll* (Etzioni et al., 2004), *TextRunner* (Yates et al., 2007) and *NELL* (Carlson et al., 2010). These open IE methods, however, often fail in achieving

the high quality due to the limited performance of automatic IE techniques. For instance, only 1 million (9.1%) of the 11 million relation tuples extracted by *TextRunner* were concrete facts (Banko, Cararella et al. 2007).

In recent years, Wikipedia provides a *large-scale*, *structure-rich* and *up-to-date* text corpus, which contains more than 4,000,000 articles and with rich semantic structures such as *Categories*, *Links* and *Infoboxes*. Wikipedia provides a new opportunity for knowledge base construction. Unfortunately, the task of harvesting semantic knowledge from Wikipedia (and other knowledge sharing sites) is challenging. That is, in spite of its rich structure (e.g., each *company* has its own article, and has links to its *products*, *headquarter*, *founder*, *CEO*, et al.), Wikipedia contents are still mostly only human readable. Semantic knowledge, e.g., the semantic relation between concepts, is not *formally* and *explicitly* stated. For example, although the article *IBM* contains links to *Thomas J. Watson*, *Thomas Watson Jr.* and *1911*, the semantic of these links are implicitly stated in natural language sentences such as "*The company was founded in 1911 by Thomas J. Watson*". Based on the above observation, we believe that there is *a missing link* between Wikipedia and a machine processable semantic network, i.e., the meaning of links is documented in natural language only – a representation which is incomprehensible for computer processing and its meaning is unclear to computer, therefore most knowledge in Wikipedia cannot be directly used in common sense reasoning and natural language understanding.

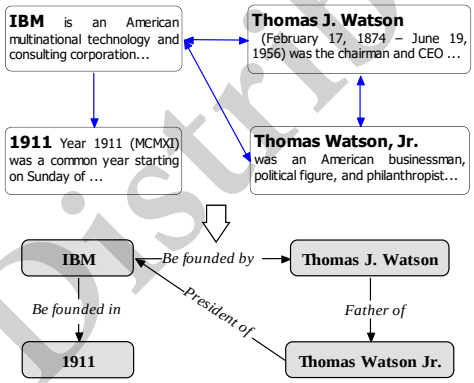


Fig. 1. Currently there are articles and links (above), our system will finally extract semantic relations from the links between these arguments (below).

In this paper, we want to establish the missing link from Wikipedia to a semantic network by providing a formalized, machine-processable semantic definition for the links between Wikipedia articles, see Figure 1 as an example. To achieve this goal, this paper proposes a semantic relation discovery method, which can:

1) Discover and characterize a large collection of semantic relations, which propose a formalized way to define the semantic of links; and
2) Annotate the links between Wikipedia articles using the above set of semantic relations.

Specifically, our method extracts relation patterns and discover semantic relations by exploiting the regularity and the redundancy of semantic relations:

1) **Regularity**: Although there is nearly unlimited ways to express a specific relation, in many cases basic principles of economy of expression and/or conventions of genre will ensure that certain systematic ways (i.e., the patterns) will be used to express a specific relation (Wang et al., 2012). For example, in most cases the *IS-A* relation will be expressed by the pattern "*Arg1 is a Arg2*", although there may exist many other ways to express it. This paper refers this regularity as *relation pattern regularity*. Based on the relation pattern regularity, we believe that the patterns of relations will be repeatedly used to the extent that it can be identified and categorized from a large corpus.
2) **Redundancy**. Due to the regularity and the large size of Wikipedia, the same relation instance will be expressed redundantly in many different ways and many times. For example, the relation *Be-Founded-In(IBM, 1911)* is expressed in many different ways in Wikipedia such as the *link between IBM and 1911*, the *Infobox of IBM*, and natural language sentences such as "*IBM was founded in 1911*". This paper refers this redundancy as *relation instance redundancy*.

Based on the above observations, we propose to exploit the above regularity and redundancy using a hierarchical Dirichlet process (HDP) model (Teh et al., 2006), where the regularity and redundancy are modeled as statistical distributions and the dependencies between them. Furthermore, the HDP can adaptively determine the number of relations underlying the relation instances, which is a challenging problem for relation discovery.

We have applied our relation discovery method to Wikipedia, and finally *14,299* relations, *105,661* relation patterns and *5,214,175* relation instances are discovered. We believe this will be a valuable resource for many NLP tasks.

This paper is organized as follows. Section 2 describes the data preprocessing step. Section 3 demonstrates how to extract relation instances from Wikipedia. Section 4 describes how to discover semantic relations using the HDP model. Section 5 presents the experiments. Section 6 reviews the related work. Section 7 concludes this paper.

2 Data Preprocessing

In this section, we describe the data preprocessing steps for Wikipedia, including Wikipedia text preprocessing and entity linking.

2.1 Wikipedia Text Preprocessing

In this paper, we use the Jan. 30, 2010 English version of Wikipedia. Given the Wikipedia data, we first segment the main content of each article into sentences, and discard the sentences which are too short (< 4 words) or too long (> 50 words). Finally we collect 26,852,307 sentences. For each sentence, we tokenize, tag and parse them using the Stanford CoreNLP Tools[1].

[1] http://nlp.stanford.edu/software/corenlp.shtml

2.2 Entity Linking

In order to discover semantic relations between entities, we need to identify all occurrences of a specific entity. Unfortunately, there are many different ways to mention a specific entity, including *name mentions*, *nominal mentions* and *pronoun mentions* (Doddington et al., 2004). For example, the company *IBM* may be mentioned by its name *IBM*, the nominal *the company* and the pronoun *it*.

To resolve the above problem, this paper links all mentions with their referent entities (i.e., *entity linking*) through the following two steps:

Linking Name Mentions to Entities. In this step, we link all name mentions to their referent entity. There have been a lot of entity linking methods, in this paper we use the entity-topic model described in (Han & Sun, 2012), which collectively links all name mentions in a document by exploiting both the mention context and the document topics.

Linking Subject Mentions to Entities. In this step we link the nominal mentions and the pronoun mentions to their referents (e.g., it → IBM). In this paper, we use the method described in Li et al. (2010), which identify the subject mentions of a Wikipedia article by finding the top 3 frequent subject noun phrases of a Wikipedia article.

2.3 Relation Instance Extraction

This section describes how to extract the relation instances from Wikipedia. Given a pair of entities in a sentence, then we describe how to: 1) extract the phrase in the sentence which expresses the relation between them; and 2) validate whether the extracted phrase is a relation pattern based on the relation instance redundancy and the relation pattern regularity.

Relation Phrase Extraction. In this paper, a relation phrase is the phrase in a sentence which expresses the relation information between two given entities. For example, the relation phrase for entities *IBM* and *1911* in sentence "*IBM was founded in 1911 by Thomas J. Watson.*" should be "*IBM was founded in 1911*".

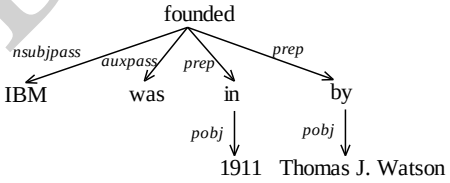


Fig. 2. A typed dependency parse tree

According to Bunescu and Mooney (2005), most of the information for identifying a relation between two entities is in the *shortest dependency path* (*SDP*) between them. Furthermore, we also observed that some modifiers of the SDP words also contain the relation information about the two entities. For example, in Figure 2, the *auxpass* modifier *was* of the SDP word *founded* is also useful for expressing the relation between (*IBM*, *1911*) and between (*IBM*, *Thomas J. Watson*).

Based on the above observation, given two entities in a sentence, this paper extracts the relation phrase of them as follows:

First, we extract all words in the SDP between the two arguments as relation phrase. For instance, in Figure 2 the SDP words "*IBM founded in 1911*" is identified for *(IBM, 1911)*;

For each SDP word, we add the selected modifiers of them to the relation phrase. Using the Stanford's typed dependencies (Marneffe & Manning, 2008), the modifiers we used for SDP words are shown in Table 1. For instance, in Figure 2 the modifier word will be added to the relation phrase for *(IBM, 1911)*, and now the relation phrase is "*IBM was founded in 1911*".

Table 1. Selected modifiers for SDP word

POS	Typed Modifiers
Verb	*aux, auxpass, cop, neg*
Noun	*cop, det, neg*

Relation Instance Extraction. Through the relation phrase extraction, we can extract many relation phrases. For example, in Figure 2 we can identify three entity pairs corresponding with their relation phrases:

- (IBM, 1911): *Arg1 be found in Arg2*
- (IBM, Thomas J. Watson): *Arg1 be found by Arg2*
- (1911,Thomas J. Watson): *be found in Arg1 by Arg2*

Unfortunately, not all relation phrases are relation patterns, e.g., the phrase "be found in Arg1 by Arg2" in above. Therefore, we need to filter out noisy relation instances. In this paper, we filter out noisy relation instances using three constraints:

Syntactic Constraint. As shown in (Etzioni and Banko, 2008) and (Chan and Roth, 2011), the relation pattern usually follows some specific syntactic patterns. Therefore we can filter out the relation phrases which are not consistent with these syntactic patterns. In this paper, we assume that all relation patterns should be consistent with the Verb patterns in (Chan and Roth, 2011), i.e., the two arguments should head in the same verb, with one argument the subject of the head verb, and the other argument the object or the preposition object of the head verb.

Link Constraint. Based on the relation instance redundancy, a relation instance should occur in many different ways. Therefore, we can filter out the relation instances which occur in only one way. In Wikipedia, a link between two articles usually indicates the existence of semantic relation between them, therefore we can filter out the relation instances with no link between their arguments. For example, if there is no link between the articles *1911* and *Thomas J. Watson*, we will filter out all relation instances whose arguments are (*1911*, *Thomas J. Watson*).

Significance Constraint. Based on the relation pattern regularity, a relation pattern will be used frequently to express a specific relation. For example, the "*Arg1 be a Arg2*" will be used many times to express the *IS-A* relation. Based on this observation, we filter out all relation phrases whose occurrences are below a specific threshold (5 times in this paper).

Using the above three constraints, our method finally identifies *105,661* relation patterns and *5,214,175* relation instances. Table 2 demonstrates the top five frequent relation patterns extracted from Wikipedia.

Table 2. The top 5 frequent relation phrases

Relation Pattern	Frequency
Arg1 be a Arg2	679,081
Arg1 be Arg2	234,081
Arg1 have Arg2	74,266
Arg1 became Arg2	39,628
Arg1 be born in Arg2	36,390

2.4 Argument Classification

Finally, we add the argument type information to the relation instance. Although Wikipedia has a category system, its categories are mostly thematic facets (Ponzetto, and Navigli, 2009) rather than categories from a well-formed taxonomy. For example, the article *IBM* is labeled with categories "*Companies listed on the New York Stock Exchange"*, "*1911 establishments in the United States*", etc. To resolve the above problem, this paper uses WordNet as the taxonomy and labels each argument with a WordNet synset using the method described in (Ponzetto, and Navigli, 2009).

Through the above relation phrase extraction, relation instance extraction and argument classification steps, we extract and represent each relation instance as a 5-tuple (*Arg1, Arg1 Type, Arg2, Arg2 Type, Relation Pattern*). For example, the relation instance *Be-Founded-In(IBM, 1911)* will be represented as (*IBM, Company, 1911, Year, Arg1 be found in Arg2*).

3 Discovering Semantic Relations using HDP Model

In this section, we describe how to discover and characterize a large collection of semantic relations from the extracted relation instances. Specifically, we address three problems in this section:

1) How many different underlying semantic relations for the extracted relation instances?
2) How to represent and characterize the discovered semantic relations?
3) For each relation instance, which semantic relation it expresses?

As described in Section 1, we resolve the above problems based on the idea that: 1) *Relation Pattern Regularity*, i.e., a certain systematic patterns will be used to express a specific relation; and 2) *Relation Distribution Regularity*, i.e., the relations for each argument type pair are usually selected from a regular and fixed set and follow a specific distribution. Based on the above idea, then we propose to model and exploit them using a hierarchical Dirichlet process model (HDP).

3.1 Document and Relation Representation

Based on the relation distribution regularity, we organize all relation instances with the same argument types into an individual document, so that the patterns in the same document will have a high likelihood to be assigned to the same relation. For example, Figure 3 shows a document for the argument type pair (*Actor*, *Actor*), corresponding with their relation patterns' count.

Doc: *Actor-Actor*	
Arg1 be a Arg2	*1,480*
Arg1 appear with Arg2	*583*
Arg1 star with Arg2	*519*
Arg1 be married to Arg2	*471*
Arg1 marry Arg2	*440*

Fig. 3. A demo of the *Actor-Actor* document

Based on the relation pattern regularity, we model each relation as a multinomial distribution of relation patterns. Figure 4 demonstrates the learned pattern distribution of the well-known *IS-A* relation.

Relation: *IS-A*	
Arg1 be a Arg2	*0.940*
Arg1 be establish as Arg2	*0.009*
Arg1 be among Arg2	*0.008*
Arg1 be consider one of Arg2	*0.005*
Arg1 be seen as Arg2	*0.005*

Fig. 4. The top 5 patterns of the *IS-A* relation

3.2 Hierarchical Dirichlet Process Model

In this section, we describe how to exploit the redundancy and the regularity using a Hierarchical Dirichlet Process (HDP) model. Specifically, the HDP model assumes that all documents are generated through the following process (Teh et al., 2006):

1. Draw the corpus level (global) relation distribution $\beta \sim GEM(\gamma)$. For example, in Figure 5 the corpus probabilities for the three relations may be drawn as β = {*Appear-With*$^{0.3}$, *IS-A*$^{0.4}$, *Be-Acquire-By*$^{0.3}$*};*
2. For each relation $z \in \{1,2,\ldots\}$, draw its relation pattern distribution $\phi_z \sim \text{Dirichlet}(\delta)$;
3. For each document d_j (i.e., a specific argument type pair), draw the document's specific relation distribution $\pi_i \sim DP(\alpha, \beta)$. For instance, in Figure 5 we may draw the relation probabilities for document *(Actor, Actor)* as {*Appear-With*$^{0.6}$, *IS-A*$^{0.4}$}, and for document (*Company, Company*) as {*IS-A*$^{0.3}$, *Be-Acquire-By*$^{0.7}$}.
4. For each relation instance x_i in a document d_j:
   a) Draw the expressed relation of the instance xi as $z_i \sim \pi_j$;
   b) Draw the relation pattern from the pattern distribution of relation z_i as $x_i \sim \phi_{z_i}$.

In HDP model, the pattern distributions of the same relation are shared across all documents, therefore the relation pattern regularity can be exploited, i.e., the same pattern distribution will be used in all documents. For example, in Figure 5 the pattern distribution of the IS-A relation will be shared across the docs (Actor, Actor) and (Company, Company). Furthermore, for each document, their relation distribution is draw from the corpus relation distribution with a concentration parameter α. Thus the HDP will put a concentrated relation distribution for each document, and the relation distribution regularity can be modeled by selecting an appropriate α. For example, although there are three global relations, only two of them will appear in doc (*Actor, Actor*).

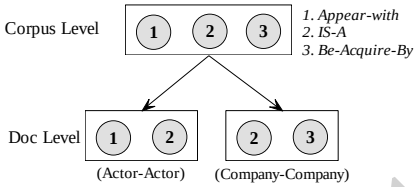


Fig. 5. A demo of the relation distribution generation of HDP

The Inference of HDP. As the same as (Teh et al., 2006), the Gibbs sampler for the HDP in this paper is as follows:

$$\begin{cases} z_{ji} = t & \propto \frac{n_{jt}^{-i}}{n_{j*}^{-i}+\alpha} f(x_{ji}, \phi_t) \\ z_{ji} = t^g & \propto \frac{\alpha}{n_{j*}^{-i}+\alpha} \frac{m_{t^g}^{-i}}{m_{*}^{-i}+\gamma} f(x_{ji}, \phi_{t^g}) \\ z_{ji} = t^{new} & \propto \frac{\alpha}{n_{j*}^{-i}+\alpha} \frac{\gamma}{m_{*}^{-i}+\gamma} f(x_{ji}, \phi_{t^{new}}) \end{cases}$$

where x_{ji} is the i^{th} instance in d_j, z_{ji} is relation assignment for x_{ji}, $z_{ji}=t$ means assign x_{ji} with a relation which has appeared in document d_j, and $z_{ji}=t^g$ means assign x_{ji} with a relation has appeared in corpus, and $z_{ji}=t^{new}$ means assign x_{ji} with a new relation, n_{jt}^{-i} and m_t^{-i} correspondingly the appearance count of relation t in document d_j and in corpus, $n_{j*}^{-i} = \sum_t n_{jt}^{-i}$, $m_*^{-i} = \sum_t m_t^{-i}$, the $f(x_{ji}, \phi_t)$ is the likelihood of generating pattern x_{ji} from relation t.

Notice that the above HDP model is a non-parametric Bayesian model, it can generate new relation when $z_{ji}=t^{new}$ is sampled, therefore it can adaptively determine the number of relations underlying the extracted relation instances. Furthermore, the above inference process can identify which semantic relation a relation instance expressed by assigning it with a relation. After the assignment, we can easily get the pattern distribution of all relations by estimating them from the final assignments.

The Hyperparameter Setting. In HDP model, the hyperparameter α controls the number of relations in a document, and α together with γ control the number of relations in a corpus. In this paper, following Escobar and West (1995), we put vague gamma priors on α and γ so that their values can be adaptively learned. Concretely, we set $\alpha \propto \text{Gamma}(1,1)$ and $\gamma \propto \text{Gamma}(1,1)$, and the final value of α and γ in our data set are correspondingly around *1790* and *27*. For σ, we set σ to a small value *0.0001* so that HDP can express a relation with a regular and fixed set of patterns.

4 Experiments

4.1 Experimental Settings

Generally, the semantic relation discovery is a process of grouping relation patterns into clusters $C=\{C_1, C_2, ..., C_n\}$, with each cluster C_i representing a semantic relation. Therefore, we can evaluate the system as a clustering system.

Data Set. Due to the size of relation patterns and relations, we evaluate the quality of discovered relations under 4 entity type pairs, including *Company-Month*, *Company-Company*, *Company-City* and *Company-People*. For each entity type pair, we manually group the salient patterns (whose appearing probability is no smaller than 5% in at least one discovered relation of the entity type pair) into relation clusters $L = \{L_1, L_2, ..., L_m\}$, with each relation cluster is a set of relation patterns indicating the same relation.

Evaluation Criteria. Given the discovered relations C and the manually clustered relations L, we evaluate the quality of the discovered semantic relations using the standard clustering metrics: *Purity* , *Inverse Purity* and *F-Measure* (Amigo et al., 2008).

Baselines. We compare our method with two baselines:

1) ***One_in_One:*** The first is *One_in_One*, which assigns each relation pattern to an individual cluster, therefore the Purity of *One_in_One* will always be *1.0*.

2) ***ET_in_One***: The second is *ET_in_One*, which assigns all relation patterns with same entity argument types into a single cluster. In our data set the Inverse Purity of *ET_in_One* will always be 1.0.

4.2 Experimental Results

In this section we demonstrate and discuss the experimental results. Table 3 shows the size of discovered relations and Table 4 shows the quality of the discovered relations.

Table 3. The size of discovered semantic relations

Relation	Relation Pattern	Relation Instance
14,299	*105,661*	*5,214,175*

Table 4. The quality of discovered relations

	Pur	Pur_Inv	F
One_in_One	*1.00*	*0.29*	*0.45*
ET_in_One	*0.40*	*1.00*	*0.57*
Our Method	***0.77***	***0.56***	***0.65***

From the Table 3 and 4, we can see that:

1) Our method can discover a large collection of relations: totally *14,299* relations, *105,661* patterns and *5,214,175* instances are discovered. We believe this will be a valuable resource for many NLP tasks;

2) Our method can discover homogeneous and complete relations: the average *Purity* and *Inverse Purity* of learned relations are about 0.77 and 0.56, and a 20% and 8% F-measure improvements are achieved over the *One_in_One* and the *ET_in_One* baselines. This means that for each resulting cluster around 77% patterns within it will

express the same relation, and for each relation there will be a cluster which can capture around 56% patterns of it.

Table 5. Some examples of learned relations

Relation	Top 5 Frequent Patterns with Prob.	
(Company, Month)#1	*Arg1 be found on Arg2*	*0.391*
	Arg1 be incorporate on Arg2	*0.126*
	Arg1 be found Arg2	*0.076*
	Arg1 be form in Arg2	*0.067*
	in Arg1, Arg2 merge to form	*0.047*
(Company, Company)#1	*Arg1 be sold to Arg2*	*0.311*
	Arg1 be acquire by Arg2	*0.248*
	Arg1 acquire in Arg2	*0.078*
	Arg1 own Arg2	*0.039*
	Arg1 be list a constituent of Arg2	*0.031*
(Company, City)#1	*Arg1 be headquarter in Arg2*	*0.280*
	Arg1 establish in Arg2	*0.139*
	Arg1 be establish in Arg2	*0.099*
	Arg1 be open in Arg2	*0.037*
	Arg1 be a company base in Arg2	*0.022*
(Company, People)#1	*Arg1 work at Arg2*	*0.288*
	Arg1 be found by Arg2	*0.233*
	Arg1 to work for Arg2	*0.089*
	Arg1 be hire by Arg2	*0.056*
	Arg1 to work at Arg2	*0.031*

Table 5 also shows the top 1 frequent relation (represented using its top 5 patterns) of the above 4 argument type pairs. From Table 5 we can see that:

1) Our method can group patterns which may implicitly express the same relation. For example, in Table 5 the pattern "*Arg1 be sold to Arg2*" can entail "*Arg1 be list a constituent of Arg2*", and the pattern "*Arg1 be headquarter in Arg2*" usually entails "*Arg1 establish in Arg2*".
2) Some relations are hard to be distinguished from each other, because they are highly coupled in different documents. For example, in the *(company, people)* document, because the founder of a company will also work at that company, it will be hard to distinguish the *Be-Found-By* relation with the *Work-At* relation.

5 Related Work

In this section, we briefly review the related work of relation discovery. Start from the Message Understanding Conferences (MUC) (Grishman & Sundheim, 1996), most relation extraction work focuses on supervised relation extraction methods, i.e., identifying and classifying relation instances within a document, given the annotated corpus and the target relation types. However, due to the large amount of manual engineering for corpus annotation and the large size of relations, recent research has focused on weakly supervised and self-supervised relation extraction, such as *DIPRE* (Brin, 1999), *Snowball* (Agichtein, Eugene & Gravano, 2000), *KnowItAll* (Etzioni et al., 2004), *TextRunner* (Yates et al., 2007) and *NELL* (Carlson et al., 2010). The idea of these weakly supervised methods is to exploit the duality between relation instances and patterns,

then a bootstrapping process can be constructed to iteratively extract new instances of the given relations.

In recent years, with the population of knowledge sharing web sites, a lot research efforts have been devoted to harvest machine-readable knowledge from Wikipedia, some projects include *Yago* (Suchanek et al., 2008), *DBpedia* (Auer et al., 2007) and *Kylin* (Wu & Weld, 2007). The shortage of these projects is that they usually only harvest knowledge from the structures whose semantics is explicitly given, mostly the *Infoboxes* in Wikipedia. There were also some other research focus on building a relation extraction system using the distant supervision methods (Mintz et al., 2009), or organize the relation pattern using argument taxonomy hierarchy (Nakashole et al., 2012).

Some other work focuses on relation discovery from single domain corpus (Chen et al., 2011; Mohamed et al., 2011). The idea of these methods is to exploit the regularity in different syntactic levels, then identify the salient syntactic patterns in a domain as discovered relations.

6 Conclusions

This paper proposes a method which can discover a large collection of semantic relations from Wikipedia by exploiting the regularity and redundancy of semantic relations, and finally *14,299* relations, *105,661* patterns and *5,214,175* instances are discovered from Wikipedia. For future work, we want to exploit the argument type hierarchy in our method, so that the relations under lower level argument types can be inherited from their ancestor argument types. For example, the *Be-Married-To* relation of *(Actor, Actor)* can be inherited from *(People, People)*.

References

1. Agichtein, E., and Gravano, L.: Snowball: Extracting relations from large plain-text collections. In: Proceedings of the fifth ACM conference on Digital Libraries, pp. 85--94. ACM, New York (2000)
2. Amigo, E., Gonzalo, J., Artiles, J. and Verdejo, F.: A comparison of extrinsic clustering evaluation metrics based on formal constraints. Identification of Common Molecular Subsequences. 12, 461--486 (2009)
3. Auer, S. and Bizer, C., et al.: DBpedia: A nucleus for a web of open data. In: The Semantic Web, vol. 4825, pp. 722--735. Springer, Heidelberg (2007)
4. Baker, C. F., Charles J. F., and John B. L.: The Berkeley Framenet project. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, pp. 86--90. Association for Computational Linguistics, Stroudsburg (1998)
5. Bunescu, R. and Mooney, R.: A shortest path dependency kernel for relation extraction. In: Proceedings of the conference on human language technology and empirical methods in natural language processing, pp. 724--731. Association for Computational Linguistics, Stroudsburg (2005)
6. Brin, S.: Extracting patterns and relations from the world wide web. In: International Workshop on The World Wide Web and Databases, pp. 172--183. (1999)

7. Carlson, A. and Betteridge, J., et al.: Toward an Architecture for Never-Ending Language Learning. In: Proceedings of the Conference on Artificial Intelligence (AAAI 2010), pp. 3. AAAI Press, Palo Alto (2010)
8. Chan, Y. S. and Roth, D.: Exploiting Syntactico-Semantic Structures for Relation Extraction. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 551--560. (2011)
9. Chen, H. and Benson, E., et al.: In-domain Relation Discovery with Meta-constraints via Posterior Regularization. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 530--540. Association for Computational Linguistics, Stroudsburg (2011)
10. Doddington, G., et al.: The automatic content extraction (ACE) program–tasks, data, and evaluation. In: Proceedings of LREC (2004)
11. Etzioni, O. and Banko, M., et al.: Open information extraction from the web. Communications of ACM. 51, 68--74 (2008)
12. Etzioni, O., et al.: Web-scale information extraction in knowitall:(preliminary results). In: Proceedings of the 13th international conference on World Wide Web, pp. 100--110. ACM, New York (2004)
13. Grishman, R. and Sundheim, B.: Message understanding conference-6: A brief history. In: Proceedings of the 16th International Conference on Computational Linguistics, pp. 466--471. (1996)
14. Han, X. and Sun, L.: An Entity-Topic Model for Entity Linking. In: Proceedings of EMNLP-CoNLL, pp. 105--115. Association for Computational Linguistics, Stroudsburg (2012)
15. Li, P., Jiang, J., et al.: Generating Templates of Entity Summaries with an Entity-Aspect Model and Pattern Mining. In Proceedings of ACL, pp. 640--649. Association for Computational Linguistics, Stroudsburg (2010)
16. Matuszek, C., Cabral, J., Witbrock, M., & DeOliveira, J.: An introduction to the syntax and content of Cyc. In: Proceedings of the 2006 AAAI spring symposium on formalizing and compiling background knowledge and its applications to knowledge representation and question answering, pp. 44--49. AAAI Press, Palo Alto (2006)
18. Miller, G. A.: WordNet: A Lexical Database for English. Communications of the ACM. 38, 39--41 (1995)
19. Mintz, M., Bills, S., Snow, R. and Jurafsky D.: Distant supervision for relation extraction without labeled data. In: Proceedings ACL-IJCNLP, pp. 1003—1011. Association for Computational Linguistics, Stroudsburg (2009)
20. Mohamed, T. P. and Hruschka, J. E. R., et al.: Discovering relations between noun categories. In Proceedings of EMNLP, pp. 1447--1455. Association for Computational Linguistics, Stroudsburg (2011)
21. Nakashole, N., Weikum, G., Suchanek, F.: PATTY: A Taxonomy of Relational Patterns with Semantic Types. In: Proceedings of EMNLP, pp. 1135--1145. (2012)
22. Ponzetto, S. P. and Navigli, R.: Large-scale taxonomy mapping for restructuring and integrating Wikipedia. In: Proceedings of the 21th IJCAI, pp. 2083--2088. AAAI Press, Palo Alto (2009)
23. Suchanek, F. M. and Kasneci, G., et al.: Yago: A large ontology from Wikipedia and Wordnet. Web Semantics: Science, Services and Agents on the World Wide Web. 6, 203--217 (2008)
24. Teh, Y. W. and Jordan, M. I., et al.: Hierarchical Dirichlet processes. Journal of the American Statistical Association. 101, 1566--1581 (2006)
25. Wang, C., Kalyanpur, A., et al.: Relation extraction and scoring in DeepQA. IBM Journal of Research and Development. 56, 9:1—9:12 (2012)
26. Wu, F. and Weld, D. S.: Autonomously semantifying wikipedia. In: Proceedings of CIKM, pp. 41--50. ACM, New York (2007)
27. Yates, A., et al.: TextRunner: Open information extraction on the web. In: Proceedings of HLT-NAACL, pp. 25--26. Association for Computational Linguistics, Stroudsburg (2007)

Research on Knowledge Fusion Connotation and Process Model

Hao Fan[1], Fei Wang[1⋆], and Mao Zheng[2]

[1] School of Information Management, Wuhan University
Wuhan, Hubei, 430072, P.R. China
[2] Department of Computer Science, University of Wisconsin-La Crosse
La Crosse WI, 54601, USA

Abstract. The emergence of big-data brings diversified structures and constant growths of knowledge. The objective of knowledge fusion (KF) research is to integrate, discover and exploit valuable knowledge from distributed, heterogeneous and autonomous knowledge sources, which is the necessary prerequisite and effective approach to implement knowledge services. In order to apply KF practice, this paper firstly discusses KF connotations in terms of analysing the relations and differences among various notions, i.e. knowledge fusion, knowledge integration, information fusion and data fusion. Then, based on the knowledge representation method using ontology, this paper investigates several KF implementation patterns and provides two types of dimensional KF process models oriented to demands of knowledge services.

Keywords: Knowledge Fusion, Knowledge Representation, Fusion Pattern, Process Mode

1 Introduction

With the development of data creating, releasing, storing and processing technologies, data is showing a rapid growth trend in all society areas. Of all the data available to the human civilization, 90% were produced in the past two years, the big data era has arrived[16]. Knowledge is awareness and understanding about people or things in the objective world, which is generated by feeling, communicating and logic inference activities in the course of practice and education and maybe facts, information or skills. The information chain, formed with "fact → data → information → knowledge → wisdom", indicates that big data contains huge amount of information, from which large knowledge can be extracted. Big data gives rise to the emergence of large scale knowledge bases. Famous knowledge base research projects, e.g. DBpedia, KnowItAll, NELL and YAGO, use information extraction techniques acquiring knowledge from high quality network data sources (e.g. Wikipedia), and automatically realize its construction and management[22]. Meanwhile, big data brings about information

⋆ Corresponding Author, Email to: feiwang@whu.edu.cn

overload and pollution too, in which knowledge presents characteristics of heterogeneity, diversity and independence. In the era of data, with rapidly increasing of information and knowledge, knowledge discovery has become the research focus in various disciplines, including data science and information science[25]. Therefore, in order to improve the efficiency and quality of knowledge service, issues of analysing and utilizing knowledge existing in big data, eliminating the inconsistency between different knowledge sources, and extracting, discovering and inducing the potential valuable connotations, have become important in knowledge management research.

The continuous formation and evolution have brought about autonomous, heterogeneous and multi-source features of knowledge. Knowledge Fusion (KF) is a process of acquiring and utilizing knowledge aiming at the problem of knowledge service. Operated by KF activities, implicate and undiscovered valuable knowledge is mined from various distributed and heterogeneous data sources. KF converts autonomous knowledge into new one with higher levels of intension and reliability, helps users to find potential associations between knowledge and fact, and improves decision-making levels by making more efficient, objective and scientific judgments. KF becomes a new growth point for knowledge service[23].

As an important part of knowledge management and engineering, KF has been widely received the attention of scholars in many fields, such as computer science, knowledge engineering and information science. Smirnov et al.[21] investigates patterns for context-based KF In the decision support systems. Dong et al.[7] analyses differences and relations between data fusion and KF, and realizes KF processes by combining knowledge extraction and traditional data fusion methods together. Tang et al.[23] discusses the requirement of big data KF and its basic framework. Liu et al.[15] defines a structure of multi-domain ontology and provides dynamic ontology based on KF demands through mappings between different domain ontologies. Xu et al.[24] designs a KF framework based on ontology, which is consists of several parts, such as constructing meta knowledge set, determining knowledge measurement indicators, designing fusion algorithm, applying fused knowledge, and so on. Qiu et al.[20] summaries the KF implementation path as four types based on semantic rules, *Bayesian* networks, *D-S* theories and knowledge mining, with which Zhou et al.[26] discusses various KF processing algorithms. Guo et al.[9] reviews and evaluates research trends and theoretical developments of KF, and indicates that, there is not yet a formed general framework for KF systems, as well as directly applicable KF algorithms and standardized KF procedures. The existing research mainly focuses on specific KF frameworks, algorithms, and practical theories.

In terms of time distribution of related literatures, KF is a new research topic which is produced with the change of knowledge service requirements and the development of knowledge management research. In order to implement KF in practice, it is necessary to correctly understand KF connotation by analysing relations and differences among various relative notions, i.e. knowledge fusion, knowledge integration, information fusion and data fusion, and analyse KF implementation patterns and its process models.

2 Knowledge Fusion Connotation

2.1 Conception of Knowledge Fusion

KF is a new concept developed on the basis of information fusion. There are many intersections between the two research areas. The early definition of KF is given by Preece in the KRAFT project[19], refers to a process locating and extracting knowledge from multiple, heterogeneous on-line sources and transforming it so that the union of the knowledge can be applied in problem-solving. The KF system in KRAFT project includes three layers of services: knowledge retrieval, transformation and fusion, in which KF is defined to associate, link and simplify the transformed distributed knowledge with a unified model, and provide solutions for the problem under specific conditions.

Smirnov et al.[21] proposes that the aim of KF is to integrate multi-source information and knowledge into a unified knowledge structure model, in order to allow decision-makers to understand and look insight into the decision-making environment and provide the needed knowledge to solve problems. Hou[11] and Xu[24] believe that KF is the process of intelligently processing distributed databases, knowledge bases and data warehouses, and acquiring new knowledge by transformation and integration procedures. It aims to realize the sharing and cooperation between different knowledge resource systems, and apply knowledge mining among knowledge bases. These definitions have carried on the inheritance and development to the Preece's KF concept, which is emphasized that fusion results are productions of new knowledge.

Guo[9] and Tang[23] propose that KF is mainly studying the transformation, integration and aggregation processes in distributed knowledge base systems in order to generate new knowledge, and investigating optimization processes of knowledge structures and contents to provide knowledge service. This definition concerns processes of knowledge innovation and knowledge optimization, indicates the KF aim as providing knowledge services, and extends the KF object from traditional resources (such as databases, knowledge bases, fact parameters acquired by sensors, etc.) to the one including rules, models, methods, and even experiences, ideas, etc. In other words, the object of KF includes not only explicit knowledge, but also tacit knowledge.

Dong et al.[8] considers KF as the issue assessing and measuring the accuracy of extracting knowledge. In the process of building a knowledge base, it is required to extract knowledge from distributed data sources, and integrate it into the base. A number of different knowledge extractors might be used during knowledge extraction, and each extractor generates its corresponding knowledge results. So, it is required to evaluate the accuracy of each extracted result to improve the correctness of knowledge bases.

Hu et al.[10] extracts and transforms sentences in Web page texts into triple semantic nets for representing knowledge. It defines KF as the process eliminating contradictions among extracted knowledge and integrating its structures in accordance with user constraints and rules, which solves problems of incomplete, fuzzy, redundant and inconsistent knowledge contained in Web page texts.

Kampis et al.[12] proposes the notation of *Collaborative KF*, and indicates that traditional KF assumes informational completeness, while collaborative KF is a version of KF where traditional fusion events are local, e.g. happen upon the meetings of individual knowledge providers, and global fusion happens due to the collective (hence "collaborative") interaction dynamics. In collaborative KF, there is no guarantee that different knowledge sources were keeping unchanged and available at any time.

To sum up, concepts of KF are different in different periods and research fields. In the field of computer science and database research, KF emphasizes on the representation, transformation, cleansing and integration of explicit knowledge, focuses on eliminating the inconsistency, incompleteness, redundancy and uncertainty of knowledge among different knowledge sources, which mainly investigates on KF algorithm design and implementation so as to improve the standardization and credibility of fused knowledge. In the field of library and information science, knowledge refers to the sum of cognition and experience in the practice of changing the world, in which both explicit knowledge and tacit knowledge are concerned. KF research is to construct theory and method systems, which emphasizes on the integration of tacit knowledge and its impact.

2.2 Knowledge Fusion and Knowledge Integration

KF and knowledge integration are both knowledge object-oriented in terms of dealing with different structure and multi-source knowledge, which have connections and differences to each other. Literally, "integration" is the process of aggregating multiple individual objects to form a whole one, while "fusion" is the process of recombining multiple individual objects, splitting and dismantling it into a complete one. Integration emphasizes on aggregation and combination, while fusion more on merging and reorganizing. After fusion process, knowledge objects are supposed to have new emerging features relative to original ones.

Scholars have given definitions of knowledge integration from various perspectives. In the field of management, library and information science, Liu et al.[13] indicates that knowledge integration refers to the process of dynamically enhancing the core competitiveness of an organization though different merging levels between knowledge and knowledge, knowledge and people, and knowledge and procedures, which aims to realize the knowledge innovation. Cai et al.[6] gives a review of knowledge integration research, and proposes that knowledge integration is a comprehensive process of technology organization and human resource management, in which the initiative and creativity of the integrated entity need to be emphasized. Knowledge integration is an essentially important step in the dynamic process of knowledge innovation.

In the field of computer science and automatic control, knowledge integration research emphasizes on handling organizable and expressible explicit knowledge. Liu et al.[14] indicates that, knowledge integration is mainly to identify, process, evaluate and reform new knowledge, to realize interactions between new knowledge and original one, and to provide users with an unified knowledge

access interface and intelligent knowledge service by integrating different knowledge structures. Bohlouli et al.[4] investigates a knowledge integration framework based on big data analysis platform, divides knowledge integration processes into acquisition, representation, evaluation, transformation, aggregation and matching of knowledge, which is to provide services for intelligent knowledge retrieval.

In the field of library and information science, relative research is gradually changing from resource integration to resource aggregation. Resource integration refers to combination of all the relative independent resources to a new organic whole, through reorganizing, coordinating, recombining and optimizing the existing status of resource portfolio, which aims to solve the problem of information redundancy, content duplication and inconsistence between primary and secondary documents, while resource aggregation is borrowed from the concept of organic chemistry and refers to fusing knowledge elements to generate new ones by using artificial intelligence technologies, which aims to discover internal semantic associations among resources. Resource aggregation constructs a multidimensional and multi-level resource system with content correlation, and forms a solid knowledge network combining concept themes, subject contents and research objects as a whole[5]. At the conceptual level, KF and resource aggregation have the similar connotations.

Therefore, this paper argues that KF is the advanced stage of knowledge integration. KF applies fusion algorithms and matching rules over the result of knowledge integration to implement deduction, discovery and innovation of knowledge. Furthermore, KF is also difference from knowledge aggregation, in which KF has no need to keep and remain all knowledge concepts, relationships and instances from the original sources, but need to construct the required objects meeting knowledge service demands.

2.3 Fusion of Data, Information and Knowledge

In practice, the term "data", "information" and "knowledge" are not strictly distinguished in statements, and can even be used interchangeably. However, there is a general consensus on distinguishing between the three concepts. A commonly held view, including minor variants is that data is raw numbers and facts without processing, information is processed data, and knowledge is the result of learning and reasoning[1].

The concept of data fusion is mostly in the field of computer science and engineering science. Bleiholder et al.[2] indicates that data fusion is the last step in a data integration process, where schemata have been matched and duplicate records have been identified. Data fusion merges duplicate records into a single representation and, at the same time, resolves existing data conflicts. Dong et al.[7] also indicates that data fusion aims at resolving conflicts from data and increasing correctness for data integration.

Information fusion is a multidisciplinary research field widely concerned by academic and industrial scientists, and in lots of literature, terms of information/data fusion and information/data integration are used interchangeably.

Typically, information fusion refers to the study on efficient methods for automatically or semi-automatically transforming information in time from different sources and different points into a representation that provides effective support for human or automated decision making[3].

Thus, generalized information fusion involves intersections of multiple disciplinary for the processing different information objects. According to application scenarios and processing objects, data/information/knowledge fusions can be regarded as the different levels of abstraction for realizing generalized information fusion. Data fusion is the process of removing noise and redundancy, reducing uncertainty and improving accuracy and reliability of original data at signal and pixel levels. Information fusion is the process of extracting features from multi-source raw data and eliminating contradictions between data contents to improve the consistency and reliability of fused information providing local supports for decision-makers. Data fusion handles raw data on the signal level, and so does information fusion on the feature level. Both of them are belonging to the low-level fusion, while the high-level KF is on the decision level, which involves processes of situation awareness and assessment, influence degree evaluation, fusion optimization, mining implicit information, reasoning and judgment of decision conditions, and so on.

3 Knowledge Representation based on Ontology

Knowledge representation is the process of symbolizing, formalizing and modeling knowledge, which is the foundation of knowledge organization and the prerequisite for realizing knowledge management. Traditional knowledge representation technologies include state-space, predicate logic, generative rule and frame methods. Along with the discipline crossing and increased complexity of knowledge, methods of neural network, fuzzy set, object-oriented and ontology are developed for knowledge representation. Different knowledge representation methods lead to heterogeneities of knowledge, which is an emerging issue addressed in the research of KF systems.

Although the expressive power and reasoning ability of ontology is less than the traditional formal methods, in order to solve the problem of heterogeneous knowledge, many researches use ontology to represent knowledge and construct knowledge bases[9]. As a structured knowledge representation method, ontology is able to abstractly express a domain as a set of concepts and relationships between the concepts, and unify the domain concepts for sharing the formal specification of the conceptual model, exchanging and reusing knowledge between human and computers.

In the Web Ontology Language, *OWL 2*[1], recommended by W3C, the basic modeling elements of ontology are *Classes*, *Properties*, and *Individuals*. All entity objects are represented as individuals, while type of entities as classes, and entity relationships as attributes. Attribute can be further refined as subattributes, such as object relationships, object features, object value ranges, and

[1] `https://www.w3.org/TR/2012/REC-owl2-primer-20121211/`

so on. Pérez[18] classifies five ontology modeling primitives: *Concepts*, *Relations*, *Functions*, *Axioms* and *Instances*. A concept can be anything including the description of a task, function, action, strategy, reasoning process, etc; Relations represent a type of interaction between concepts of the domain; Functions are a special case of relations in which the *n-th* element of the relationship is unique for the n-1 preceding elements; Axioms are used to model sentences that are always true; and instances are used to represent elements.

Based on the *OWL 2* definition and Pérez's five modeling primitives, we define a knowledge ontology as the form of five-tuple: $ontology(O) = \langle C, A, R, D, I \rangle$, where C is a set of concepts or classes with hierarchical structure; A is a set of attributes describing features of concepts, and usually defined as attributes of classes; R is a set of relationships, including functions, axioms and other constraints, representing effective associations between concepts, such as father, son and equality relationships, functional relationships and True assertions; D is a set of attribute domains, describing fields or value ranges of attributes; and I is a set of instances, containing entity objects of concept classes.

For example, if $\langle C_H, A_H, R_H, D_H, I_H \rangle$ is defined as an ontology for describing hypertension, set C_H may contain concepts such as $\langle\langle \mathsf{HBP} \rangle\rangle$, $\langle\langle \mathsf{Cause} \rangle\rangle$, $\langle\langle \mathsf{Symptom} \rangle\rangle$, $\langle\langle \mathsf{Therapy} \rangle\rangle$, $\langle\langle \mathsf{Patient} \rangle\rangle$, etc.; set A_H contains attributes of the concepts such as $\langle\langle \mathsf{HBP}, \mathsf{type} \rangle\rangle$, $\langle\langle \mathsf{HBP}, \mathsf{level} \rangle\rangle$, $\langle\langle \mathsf{Cause}, \mathsf{humoral} \rangle\rangle$, $\langle\langle \mathsf{Cause}, \mathsf{nervous} \rangle\rangle$, etc.; set R_H indicates relationships between concepts, e.g. $father(\langle\langle \mathsf{HBP} \rangle\rangle, \langle\langle \mathsf{PrimaryHBP} \rangle\rangle)$ means that $\langle\langle \mathsf{HBP} \rangle\rangle$ is the father class of $\langle\langle \mathsf{PrimaryHBP} \rangle\rangle$; and if any, D_H and I_H may contain concept value ranges and its instances.

The five-tuple form reflects the process of hierarchically modeling knowledge from entities to concepts. If only knowledge entities or concepts are separately considered to be merged, the KF process is not comprehensive and completed. In other words, all elements of the knowledge ontology form need to be handled in KF processes, which will be discussed in the next section as KF patterns.

4 Patterns of Knowledge Fusion

So far, there are not many literatures about KF patterns. Xu et al.[24] classifies KF into active and passive types. Qiu[20] and Zhou[26] discuss several kinds of KF processing algorithms. Smirnov et al.[21] proposes seven context-based KF patterns, i.e. *Simple*, *Extension*, *Configured*, *Instantiated*, *Flat*, *Historical* and *Adaptation Fusion*, which are classified upon the problem solved by each KF process for satisfying the requirement of the decision support system.

In this section, we classify KF patterns, from the perspective of knowledge representation, according to the five-tuple ontology form.

Instance Fusion is the process of removing redundancy, deducing noise, correcting error and merging content for entity objects and producing a new set, in which knowledge sources usually have the same modeling structure, or can be converted into the same one. After Instance Fusion, the modeling structure of source knowledge is totally or partly inherited into the fused target in accordance

with user definitions and requirements, where the pertinence, consistency and correctness of knowledge entities are improved. There is a substantial overlap between Instance Fusion and traditional information fusion, so that the former can be implemented by using the latter fusing methods as references.

Domain Fusion is the process of applying set operations like `UNION`, `INTERSECT`, `MINUS` and `EXCEPT` on attribute fields or value ranges of source knowledge entities, resulting in attribute definitions of fused knowledge entities. When Instance Fusion is applied, knowledge sources might be in the same modeling structure but different domains, which is required to redefine the attribute domain of fused knowledge. Domain Fusion remains the modeling structure of source knowledge, but change its attribute fields or value ranges, which is an extension and expansion of Instance Fusion.

Relationship Fusion is the process of merging relationships in source knowledge by removing redundancy and combining structures, as well as applying inductive and deductive reasoning over relationships for inferring and mining a new one. Relationships in knowledge ontology include interactions between concepts, affiliations between concepts and attributes, functions defining particular mappings, and axioms representing true assertions. Relationship Fusion explores and derives new relationships according to original ones in the source, in which modeling structures might be different from either each other, or the fused one where the new knowledge is generated.

Attribute Fusion is the process of comparing, analysing, transforming and merging attributes of knowledge concepts, in terms of classifying, selecting and reorganizing the object features according to users requirements. In the situation of Attribute Fusion, there are usually differences between modeling structures of knowledge sources, especially including complementary, contradiction and homograph differences in attribute definitions. After Attribute Fusion, new attributes appear in the fused knowledge,and new relationships are also required to correspond with them. Thus, Attribute Fusion and Relationship Fusion are two complementary and alternately iterative processes, both are important parts of knowledge discovery and innovation processes

Concept Fusion is the process of constructing new knowledge concepts, which might bring about new attributes and new relationships as well. Therefore, it is not possible to individually produce Concept Fusion separately from the other KF patterns, which have to be based on Instance Fusion, iteratively and incrementally applying Domain, Relationship and Attribute Fusions to achieve a whole fusion process. Concept Fusion is considered as the high level of the KF hierarchy, where Domain, Relationship and Attribute Fusions are middle levels between the low level Instance Fusion and the high level Concept Fusion. It is difficult to directly apply traditional information fusion methods for Concept Fusion to generate new knowledge, thus new KF approaches need to be developed, and participations of domain experts are also required for the completion of knowledge innovation.

5 Process Model of Knowledge Fusion

As discussed above, different KF patterns meet different requirements and produce different fusion results. This section proposes two types of process models to analyse the operational mechanism of KF patterns.

5.1 One-Dimension KF Process Model

Relationship, Attribute and Concept Fusions are processes of knowledge innovation, to a certain extent, by changing the original knowledge models and generating a new one; Instance Fusion changes knowledge objects in terms of consistency, correctness, validity and quantities, which is a process of manifesting and discovering knowledge; and Domain Fusion is the transitional phase from knowledge discovery to knowledge innovation, which does not change the original knowledge model but the value range of the concepts.

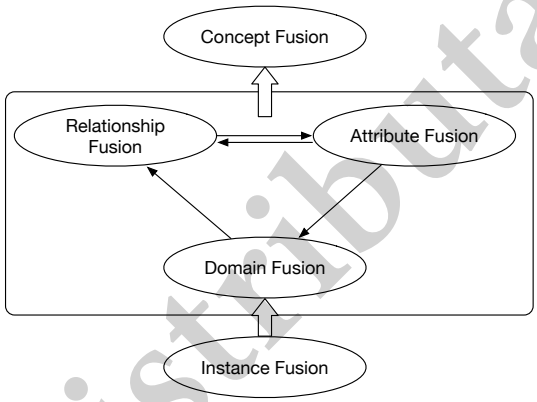


Fig. 1. One-Dimension KF Process Model

Figure 1 gives the one-dimension KF process mode to illustrate relationships among the five KF patterns. The requirement of Domain Fusion is generated on the basis of Instance Fusion. In different knowledge sources, value ranges of concepts might be different from each other, which is required to be adjusted, merged and redefined, i.e. producing Domain Fusion, to meet the demand of Instance Fusion. After changes of concept domains, relationships between the concepts may also need to change so as to affect the inferring results of Relationship Fusion. E.g. the increase or decrease of a concept value ranges is likely to affect the establishment of equal relationships between the concepts. At the same time, Relationship Fusion and Attribute Fusion are also two interactive and complementary processes. The production of new attributes might lead to the generation of new relationships, and vice versa.

Therefore, the three KF patterns, i.e. Domain Fusion, Relationship Fusion and Attribute Fusion, are performing in a way of loop iterations. In order to eventually achieve Concept Fusion, each iteration makes a further step in the

progress of generating new knowledge. Thus, KF processes could not be completed only by a single fusion pattern, nor by a stepwise linear procedure. All fusion patterns need to be comprehensively considered, and KF is realized in a way of loop iteration, incremental progression and spiral development.

5.2 Two-Dimension KF Process Model

As mentioned above, KF generates new knowledge and produces knowledge innovation, while the aim of knowledge innovation is to provide better knowledge service. Nonaka et al.[17] summarizes knowledge innovation processes into four stages: *Socialization*, *Externalization*, *Combination* and *Internalization*, as known as the SECI model, describing transformations between tacit and explicit knowledge. Socialization is the process of converting new tacit knowledge through shared experiences; Externalization is the process of articulating tacit knowledge into explicit knowledge; Combination is the process of converting explicit knowledge into more complex and systematic sets; Internalization is the process of embodying explicit knowledge into tacit knowledge.

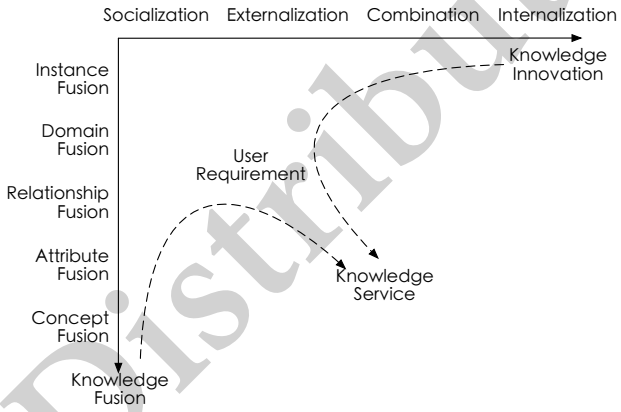


Fig. 2. Two-Dimension KF Process Model

In the SECI model, knowledge is created through a spiral by applying the four processes in a way of circular loop rather than a stepwise linear procedure, which is similar to the implementation of KF patterns. Although it is not able to directly map the KF patterns with the SECI stages, the common characteristic makes it possible to organically combine the two processes accordingly, as shown in Figure 2, in order to achieve the accurate, personalized and effective knowledge service in accordance with the user requirement. In particular, during the stages of Socialization and Externalization, methods for fusing instances and domain can be used to discover tacit knowledge objects, and methods for fusing relationships and attributes can be used to articulate it into an explicit one, while during the stages of Combination and Internalization, the fusion patterns are naturally involved since they are both supposed to handle explicit knowledge.

The two-dimensional KF process model shows relationships between the innovation stages and the fusion patterns and indicates that, although KF patterns proposed in this paper are based on the ontology representation of explicit knowledge, it have the potential to expand to tacit KF, which is one of the research issues in our future work.

6 Conclusion and Future Work

The big data era brings distributed, heterogeneous and autonomous knowledge, from which KF integrates, discovers and exploits valuable knowledge for achieving a high quality service. This paper discuss the KF connotation in terms of giving the definition of KF and analysing the relation and difference between KF and various notions, such as knowledge integration, information fusion and data fusion. Then, we introduce five KF patterns, i.e. *Instance*, *Domain*, *Relationship*, *Attribute* and *Concept Fusion*, and indicate that the KF process is implemented in a way of loop iteration, incremental progression and spiral development, rather than only by a single step, nor a stepwise linear procedure. Finally, two types of dimensional KF process models are proposed to illustrate relationships between knowledge innovation stages and KF patterns. In future, we will implement the KF patterns in a specific application domain, e.g. chronic disease domain, and extend it to handle tacit knowledge.

7 Acknowledgement

This paper is supported by the Chinese NSFC International Cooperation and Exchange Program, *Research on Intelligent Home Care Platform based on Chronic Diseases Knowledge Management* (71661167007).

References

1. Alavi, M., Leidner, D.E.: Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues. MIS Quarterly, 25:107-136, (2001)
2. Bleiholder, J., Naumann, F.: Data fusion. ACM Computing Surveys, 41(1):1-41, (2008)
3. Balazs, J.A., Velasquez, J.D.: Opinion mining and information fusion: A survey. Information Fusion, 27:95-110, (2016)
4. Bohlouli, M., Merges, F., Fathi, M.: Knowledge integration of distributed enterprises using cloud based big data analytics. In Proceedings of IEEE International Conference on Electro/Information Technology, June 5-7, pages 612-617, (2014)
5. Bi, Q.: Digital resources: from integration to aggregation. Digital Library Forum, 6, (2014)
6. Cai, Q.H., Chen, G.H.: A review of knowledge integration research. In Journal of Research and Development Management, 22(6):15-22, (2010)
7. Dong, X.L., Gabrilovich, E.: From data fusion to knowledge fusion. In Proceedings of VLDB'14, (2014)

8. Dong, X.L., Srivastava, D.: Knowledge curation and knowledge fusion. In Proceedings of VLDB, pages 2063-2066, (2015)
9. Guo, Q., Guan, X., Cao, X.Y., etc.: Research progress and trends of knowledge fusion. In Journal of China Academy of Electronics and Information Technology, 7(3), (2012)
10. Hu, S.K., Cao, Y.D.: Knowledge fusion framework based on web page texts. In Frontiers of Computer Science in China, 3(4):457-464, (2009)
11. Hou, J., Yang, J.G., Jiang, Y.L.: Knowledge fusion algorithm based on metadata and ontology. In Journal of Computer-Aided Design and Computer Graphics, 18(6):819-823, (2006)
12. Kampis, G., Lukowicz, P.: Collaborative knowledge fusion by ad-hoc information distribution in crowds. In Procedia Computer Science, 51:542-551, (2015)
13. Liu, X.C., An, X.M.: Knowledge integration research status analysis. In Information and Documentation Services, 1:9-12, (2006)
14. Liu, X.L., Ma, J.: Research progress of knowledge integration based on Ontology in Semantic Web Environment. In Journal of Modern Intelligence, 01:159-163+169, (2015)
15. Liu, J.H., Xu, W.T., Jiang, H.: Research on dynamic ontology construction method for knowledge fusion in group corporation. In Knowledge Engineering and Management, volume 278 of Advances in Intelligent Systems and Computing, pages 289-298, (2014)
16. Meng, X.F., Chi, X.: Big data management: concepts, technologies and challenges. Computer Research and Development. 50(1): 146-169, (2013)
17. Nonaka, I., Umemoto, K., Senoo, D.: From information processing to knowledge creation: A paradigm shift in business management, Technology in Society, 18(2), pp.203-218, (1996)
18. Pérez, A.G., Benjamins, V.R.: Overview of knowledge sharing and reuse components: Ontologies and problem-solving methods. In Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5), (1999)
19. Preece, K., Hui, A. Gray, etc.:Kraft: An agent architecture for knowledge fusion. In International Journal of Cooperative Information Systems. 10(1-2):171-195, (2001)
20. Qiu, J.P., Yu, H.Q.: Research progress and trends of knowledge fusion in perspectives of knowledge science. In Library and Information Service, 59(08):126-132+148, (2015)
21. Smirnov, A., Levashova, T., Shilov, N.: Patterns for context-based knowledge fusion in decision support systems. In Information Fusion, (21):114-129, (2015)
22. Suchanek, F.M., Weikum, G.: Knowledge bases in the age of big data analytics. In Proceedings of the VLDB Endowment, Volume 7: 1713-1714, (2014)
23. Tang, X.B., Wei, W.: The growth points of knowledge service in big data age. In Researches in Library Science. (05):9-14, (2015)
24. Xu, C.J., Li, A.P., Liu, X.M: Knowledge fusion architecture. In Journal of Computer-Aided Design and Computer Graphics, 22(7), (2010)
25. Ye, Y., Ma, F.C.: The rise of data science and its relation with information science. In Journal of Information Science. 34(6): 575-580, (2015)
26. Zhou, F., Wang, P.B., Han, L.Y.: Multi source knowledge fusion processing algorithm. In Journal of Beijing University of Aeronautics and Astronautics, 39(1):109-114, (2013)

A Multi-dimension Weighted Graph-based Path Planning with Avoiding Hotspots

Shuo Jiang[1,2] , Zhiyong Feng[1,2], Xiaowang Zhang[2,3], Xin Wang[2,3], Guozheng Rao[2,3]

[1]School of Computer Software, Tianjin University, Tianjin 300350, China
[2]School of Computer Science and Technology, Tianjin University, Tianjin 300350, China
[3]Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin 300350, China
{jiangshuo,zyfeng,xiaowangzhang,wangx,rgz}@tju.edu.cn

Abstract . With the development of industrialization rapidly, vehicles have become an important part of people's life. However, transportation system is becoming more and more complicated. The core problem of the complicated transportation system is how to avoid hotspots. In this paper, we present a graph model based on a multi-dimension weighted graph for path planning with avoiding hotspots. Firstly, we extend one-dimension weighted graphs to multi-dimension weighted graphs where multi-dimension weights are used to characterize more features of transportation. Secondly, we develop a framework equipped with many aggregate functions for transforming multi-dimension weighted graphs into one-dimension weighted graphs in order to converse the path planning of multi-dimension weighted graphs into the shortest path problem of one-dimension weighted graphs. Finally, we implement our proposed framework and evaluate our system in some necessary practical examples. The experiment shows that our approach can provide “optimal” paths under the consideration of avoiding hotspots.

Keywords: path planning; avoiding hotspots; multi-dimension weighted graph; shortest path problem

1 Introduction

1.1 Path Planning

Path planning is a sequential algorithm based on existing nodes, edges and weights according to a certain method. These nodes, edges and weights are data in graph model, which can represent different things in different situations, such as obstacles, hotspots and so on. Path planning technology has been applied extensively in many domains since it was proposed [1]. There are plenty of applications in frontier domains: route planning of unmanned aerial vehicles, robot path planning and space path planning of rocket launch. This technology not only speeds up the progress in frontier domains, but also becomes an integral part of our daily life. For example, GPS navigation helps us plan path while we are driving. The application of the technology in business and management domain is logistics, that is, resources dispatch in

a reasonable way. Generally speaking, the problems which can be translated into graph models can be translated into nodes, edges and weights, and we can use path planning to solve them [2].

1.2 Avoiding Hotspots

Avoiding hotspots is a way to use existing data to deal with hotspots, thus making the overall planning immune to the effects of hotspots. Avoiding hotspots is not eliminating hotspots. What we avoid is the effect and damage caused by hotspots. That is to reduce the occurrence probability of hotspots. Therefore, avoiding hotspots can be used in path planning, especially vehicle routing problem.

Vehicle routing problem (VRP) was proposed firstly in 1959. It means a distribution center which provides different numbers of cargoes to a certain number of customers in a city or an area. The most important part is to plan an optimal path, the goal of which is to reach the highest economic benefit under the precondition that the requirements of customers must be met. There are some requirements in path planning such as the shortest path, the shortest time or the least oil consumption.

The loss is not only in economy, but also in environment. Fuel consumption causes air pollution. Traffic jam happens frequently in our daily life. The probability of traffic accidents is still rising. Thus, it is of great importance to avoid hotspots.

1.3 Related Works

Since the weighted graph was proposed, plenty of applications have been generated. After researching the current situation of path query of weighted graph, we reach some conclusions as follows: [3] proposes a new optimal route search model for public transit based on directed weighted graph. This model cannot only allow users to set their ideal maximum walking distance, but also meet the requirements of personalized query by using the flexible weighting graph method, especially the strong expression ability for multi-object query. Fire brigades are always required to reach the field of fire in the shortest time. Therefore, selecting the appropriate path can effectively reduce the loss of casualties and property. [4] establishes a model of multi-stage weighted directed graph aimed at this problem. Multi-stage weighted directed graph is a common graph, which can translate a lot of practical matters, such as transportation, engineering, and management, into the shortest path problem. [5] establishes a general weighted graph for transportation. It is a math model which combines network analysis and linear programming theory. This model solves the practical matter caused by network complexity, path diversity and load capacity.

There is a query of weighted graph based on regular expression in [6]. The author characterizes the query of weighted graph and proposes the algorithm. This query can be embedded effectively in XML query language. [7] proposes a query of weighted regular expression. This query can allow users to define priority of weight and be connected naturally with link information of quantitative database. The authors also propose a distribution algorithm to calculate this query. This algorithm can also solve the multi-source shortest path problem in case that we do not know the com-

plete graph. In order to query and analyze graph database by the method of aggregate function and order, [8] extends a previous graph query language. This language can support query probability graph in that way. [13] presents a SPARQL-based querying language named pSPARQL for probabilistic RDF graphs.

We can see that there is a preliminary study on weighted graph from the research above. Not only the language is of normalization and flexibility, but also the algorithm for the weighted graph is of efficiency. However, there is still something that can be extended in the weighted graph to make it more effective than previous ones. We can see that the previous studies on weighted graph only focus on one-dimension weight, while the traffic environment is complex in the real world, which means one-dimension weight cannot describe the information exactly. As we know, many cases are composed of factors influenced with each other. Therefore, it is unreasonable to calculate weighted graph with one-dimension weight.

We can also see from the research on related works that the weighted graph systems can not meet all the requirements of customers. The problems we face every day are all in characteristics for ourselves, as a result, the previous one-dimension weighted graph models can only solve a little part of the problem. Therefore, there is a lack of model or an aggregate function which users can define for their own demands to solve problems.

1.4 Overview

The overview of the paper is as follows: we focus on complex traffic environment with multi-dimension weighted graph; then we establish a model according to the specific circumstances and requirements; and we define an aggregate function which can translate multi-dimension weighted graph into one-dimension weighted graph; finally we use Dijkstra algorithm, a classical path planning algorithm, to solve the problem and propose a good plan.

The overall structure of this paper is as follows: Section 1 mainly introduces the related work, the lack of them, and how we deal with the lack through our innovations of this paper; Section 2 introduces related concept of graph; Section 3 introduces the multi-dimension weighted graph and the aggregate function; Section 4 simulates a specific case, then we show how to use our method to solve it; Section 5 shows the whole framework, related experience and the efficiency of this framework; Section 6 concludes this paper and the future work.

2 Graph

2.1 Basic Definitions

Graph is a math object to describe the relationship among objects. Assume graph G is an ordered two tuple (V, E), and V represents a set of vertices, then we can use $V(G)$ to represent a set of nodes; E represents a set of edges. Similarly, we can use $E(G)$ to represent a set of edges. Note that E and V do not intersect. The elements of E are all two tuple, which are noted by (x, y), and $x, y \in V$ [9].

Path is a sequence from one node to another. For example, assume that a path P is v_0, $(e_1, v_1, e_2, v_2, \dots, e_k, v_k)$ and the length of this path is k. There is a pair (v_{i-1}, v_i), which is an edge from v_{i-1} to v_i. If the starting node and the ending node is the same, then we say this path is close. Otherwise, we say it is open.
Graphic model is a structure model, whose function is to describe a system. Constituted by nodes and edges, it can represent everything in the real world, so it can be used to describe the relationship among all objects. Therefore, a graphic model is a good tool for modeling, and it proposes a good way to deal with complex systems.

2.2 Directed Graph

Directed graph is a subclass of graph. Every edge is directed in directed graph. Directed graph is an ordered pair. Assume there is a directed graph D, and the ordered pair is (V, E),. Then V is a nonempty set constituted by nodes of D. The elements in V are vertexes. E is a set of edges of D constituted by V.
Every element in the edge of directed graph is an ordered pair. Assume that an ordered pair is $< u, v >$ in directed graph D, which we say is a directed edge. u represents the starting node of the edge, while v represents the ending node of the edge. Therefore, $< u_i, v_i >$ and $< v_i, u_i >$ represent two different edges.

2.3 Undirected Graph

Undirected graph is a subclass of graph, however, different from directed graph. Every edge in undirected graph is undirected, and it is represented by unordered pair. Assume that an undirected graph $G =< V, E >$. V is a nonempty set constituted by nodes. E is a set of unordered two tuple constituted by the elements in V, and it is a set of edges. Intuitively, if all edges in a graph are undirected, then the graph is undirected. Unordered pair is usually noted by round brackets. Contrary to directed graph, there are no starting node s and ending node s in undirected graph. That is, the two unordered pairs (v_i, v_j) and (v_j, v_i) present the same edge.

2.4 Weighted Graph

Weighted graph is also a subclass of graph, but it is different from the previous two graphs for the reason that every edge in weighted graph is assigned with a value. This value is the weight of this edge. Weight can take a certain value to represent other objects, such as cost, probability and so on. Broadly speaking, weight in the weighted graph is usually single.
Assume there is a weighted graph $G =< V, E, W >$. V is a nonempty set constituted by nodes, then it is a set of nodes of G. E is a set of two tuple constituted by the elements in V, then it is a set of edges. W represents weight. If E is constituted by a set of unordered two tuples, then the weighted graph G is an undirected weighted graph. Otherwise, it is directed weighted graph. The study of this paper focuses on undirected weighted graph.

3 Extension of Weighted Graph

We can see that there is usually a single weight in weighted graph from the previous research. However, objects are usually affected by more than one factor in the real world. For example, when a user needs a path to reach the destination in the shortest time, we should consider about the length, the probability of traffic jam and the degree of traffic jam. Not only that, different people will have different requirements for the problem of path planning. Some people need the shortest time to reach the destination, while some people need the shortest length to reach there. Therefore, faced with many different requirements, we cannot use the single weight to solve those problems, but need to define a multi-dimension weighted graph to create different models to solve different practical problems.

3.1 Multi-dimension Weighted Graph

Multi-dimension weighted graph is an extension of weighted graph. From the Section 2 we have already known that weighted graph G can be represented as follows:

$$G =< V, E, W > \tag{1}$$

Multi-dimension weighted graph is not a single weight on every edge. Assume a graph $G_1$is a multi-dimension weighted graph. Then it can be represented as follows:

$$G_1 =< V, E, (w_1, w_2, \dots) > \tag{2}$$

Every weight in multi-dimension weighted graph is related to path planning, since the study is based on path planning. Here shows an example based on the G_1.
Assume that G_1 represents a graph of probability of traffic jam. Then V represents a set of location in a city; E, represents a set of roads; w_1, w_2, ... represents a set of attributes of the roads. In other words, they are the factors which can affect the path planning. There are three weights w_1, w_2 and w_3, where w_1 represents the length of every road; w_2 represents the degree of traffic jam; w_3 represents the probability of traffic jam of every road.
Therefore, we can connect the related factors to solve the problem in the real world more exactly. Then we will introduce the aggregate function $f(x)$ to deal with these weights.

3.2 Aggregate Function

Aggregate function $f(x)$ can calculate several weights, and obtain the functional results. That is, we can use aggregate functions to translate several weights into one weight. There are several common aggregate functions in Excel, such as addition, subtraction, multiplication, division and averaging. For example, the addition aggregate function:

$$f(w_1, w_2, \dots) = w_1 + w_2 + \cdots \tag{3}$$

The common aggregate functions are too restricted, which can only calculate common data. Users usually face difficult situations, and these aggregate functions can not deal with them well. Therefore, we need to propose aggregate functions for users to calculate special problems for their own demands. If we have the aggregate function which is defined by ourselves, then we can translate the multi-dimension weighted graph $G_1 =< V, E, (w_1, w_2, \dots) >$ into one-dimension weighted graph $G =< V, E, w_f >$ by the aggregate function $f(w_1, w_2, \dots)$, and $w_f = f(w_1, w_2, \dots)$.

4 Application of Multi-dimension Weighted Graph

We focus on a problem of navigation based on a graph of probability of traffic jam to introduce the two concepts proposed in the third chapter in detail.

4.1 Graph of Probability of Traffic Jam

We establish a graph of probability of traffic jam in order to solve the problem of path planning for those people who are in emergency. This model can reduce the risk to meet the traffic jam. This model not only has the basic information of roads and locations, but also has the attributes which will affect the traffic jam for every road. Graph of probability of traffic jam is a multi-dimension weighted graph. We define it as follows:

$$G =< V, E, (w_l, w_h, w_p) > \tag{4}$$

- V represents the locations in the city. We note $A, B, C, \dots$ to represent them.
- E represents the roads in the city. We note $a, b, c, \dots$ to represent them.
- (w_l, w_h, w_p) represents three-dimension weights, where w_l represents length of road. We note L, and $L \in (0,+\infty)$; w_h represents the degree of traffic jam. We note H, and $H = \{1, 2, 3\}$ (1 represents a weak degree, 2 represents a common degree, 3 represents a strong degree); w_p represents the probability of traffic jam. We note P, and $P \in [0,1]$.

We study the case of traffic jam in the real world, then we define the following aggregate function $f(x)$:

$$f(w_l, w_h, w_p) = w_l \times (w_h \times w_p + 1) \tag{5}$$

Then we use the above function to calculate the three-dimension weights to obtain the result. We will establish a graph model to show how to obtain this result.

4.2 Data and Results

We establish 5 nodes and 6 edges. The detail data and the graph model are as follows:

- $V = \{A, B, C, D, E\}$,
- $E = \{a, b, c, d, e\}$,

- The set of three-dimension weights of the 6 edges is W = { (12, 1, 0.1), (11, 2, 0.5), (1, 3, 0.8), (6, 2, 0.3), (15, 2, 0.2), (5, 2, 0.6)}

Figure 1 shows the graph model which stores the above data.

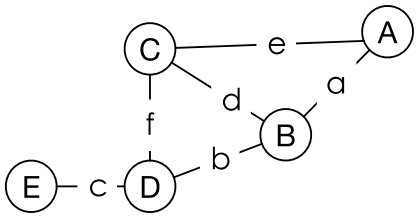


Fig. 1. The graph model

First, put W into the aggregate function $f(x)$, which we have defined before. For example, w_a = (12, 1, 0.1), then according to the $f(w_l, w_h, w_p) = w_l \times (\mathrm{w_h} \times \mathrm{w_p} + 1)$, $\mathrm{w_{fa}} = 12 \times (1 \times 0.1 + 1) = 13.2$. Then we deal with the result by rounding to get the integer 13. After calculating the three-dimension weights by aggregate function, we get the final w_f = {13, 22, 3, 9, 21, 11}. Finally we calculate the final result w_f with Dijkstra algorithm [10] to get the final value from every node to other nodes. Table 1 shows the case from node A to other nodes.

Table 1. Result of Dijkstra(A) of graph of probability of traffic jam

B	C	D	E
13	21	32	35

From the aggregate function we can see that the bigger the value is, the higher risk to meet the traffic jam will be.

5 Experiments

5.1 Framework

We show the whole framework in our architecture [11]. First, according to the special problems and different requirements, we establish the suitable models with the related factors, which will affect the result in the real world. Then, consider the relationship among these weights to define the aggregate function $f(w_1, w_2, \ldots)$. After that, put the weights of multi-dimension weighted graph in the aggregate function to get the final result of weight. This process can realize the translation from multi-dimension weighted graph to one-dimension weighted graphs. Finally, we use the Dijkstra algorithm to get the result which we need. Figure 2 shows the framework.

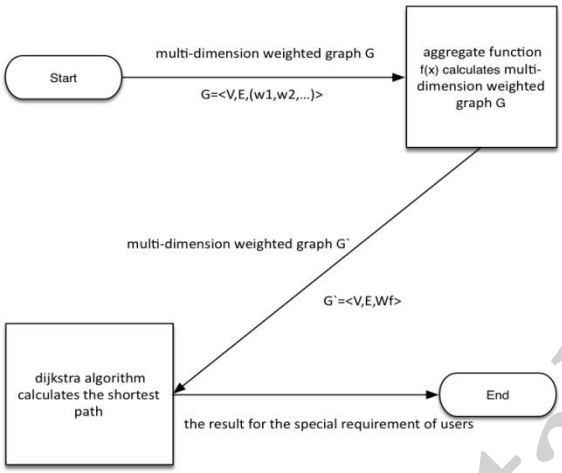


Fig. 2. The framework

We will put some more examples to show that the model can solve a lot of practical problems.

5.2 Efficiency

According to the graph of probability of traffic jam, we test the efficiency with the following number data size 10, 20, 30, 40 and 50 and time the corresponding time of program running. Then figure 3 shows the result of efficiency.

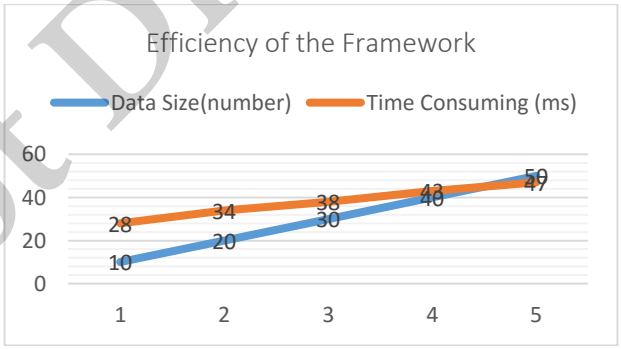


Fig. 3. The efficiency of the framework

From the result we can see that the slope of rising data size is bigger than the slope of rising time consuming. Therefore, the efficiency of the model plays an important role in the age of big data.

5.3 Graph of Traffic Accidents

We want to choose a safer path rather than the shortest path when we go to a dangerous place. According to the traffic accidents, we define the following model:

$$G = < V, E, (w_l, w_q, w_v, w_k) > \quad (6)$$

- V represents the locations in the city. We use A, B, C, …to represent them.
- E represents the roads in the city. We note a, b, c, … to represent them.
- (w_l, w_q, w_v, w_k) represents four-dimension weights, where w_l represents length of roads, and we note $L \in (0,+\infty)$; w_q represents the traffic volume. The more the volume is, the higher risk of accidents will be. We note q = {1, 2, 3} (1 represents a small volume, 2 represents a middle volume, 3 represents a large volume); w_v represents the maximum speed (the faster the speed is, the more dangerous it will be), and we note $V \in (0,+\infty)$; w_k represents the risk factor, and we note $K \in (0,1)$.

According to the research of accidents, we define the following aggregate function to deal with the four weights:

$$f(w_l, w_q, w_v, w_k) = \frac{w_l * w_q}{100} * \frac{w_v}{(1-w_k)} \quad (7)$$

We still use the earlier graph to make the experiment. We establish the following data:

- $V = \{A, B, C, D, E\}$,
- $E = \{a, b, c, d, e\}$,

And the set of four-dimension weights of the 6 edges is W = {(12, 1, 80, 0.1), (11, 2, 70, 0.5), (1, 3, 90, 0.8), (6, 2, 60, 0.3), (15, 2, 75, 0.2), (5, 2, 80, 0.6)}. First, put W into the aggregate function $f(x)$. After calculating the four-dimension weights by aggregate function, we get the final w_f = {10, 30, 13, 10, 28, 20}. Finally we calculate the final result w_f with Dijkstra algorithm to get the final value from every node to other nodes. Table 2 shows the value of risk from node A to other nodes.

Table 2. The result of Dijkstra(A) of graph of traffic accidents

B	C	D	E
10	20	40	53

According to the aggregate function we can see that the bigger the value is, the higher risk to meet the traffic accidents will be.

5.4 Graph of Traffic Cost

According to the framework, we establish a model for the user who care about the traffic cost. we define the following model:

$$G = < V, E, (w_l, w_x, w_e) > \quad (8)$$

(w_l, w_t, w_e) is three-dimension weight, where w_l represents length of roads, and we note $L \in (0,+\infty)$; w_t represents cost of consumed fuel of per kilometer, and we use $X \in (0,+\infty)$; w_v represents the maximum speed (the faster speed, the more dangerous), and we use $E \in (0,+\infty)$;
According to the research of cost, we define the following aggregate function to deal with the three weights:

$$f(w_l, w_t, w_e) = w_l \times w_x + w_e \quad (9)$$

We still use the earlier graph to make the experience. We establish the following data:

- $V = \{A, B, C, D, E\}$,
- $E = \{a, b, c, d, e\}$,
- The set of three-dimension weight of the 6 edges is W= {(15, 4, 20), (11, 2, 25), (8, 3, 15), (33, 2, 28), (15, 2, 22), (40, 2, 30)}.

First, put W into the aggregate function $f(x)$. After calculating the three-dimension weight by aggregate function, we get the final w_f = {80, 47, 39, 94, 52, 110}. Finally we calculate the final result w_f with Dijkstra algorithm to get the final value from every node to other nodes. Table 3 shows the value of cost from node A to other nodes.

Table 3. the result of Dijkstra(A) of graph of traffic cost

B	C	D	E
80	52	127	166

According to the aggregate function we can see that the bigger the value is, the higher cost spending on the path will be.

5.5 Graph of Traffic Time

According to the framework, we establish a model for the user who care about the traffic time. we define the following model:

$$G = < V, E, (w_l, w_v, w_d) > \quad (10)$$

(w_l, w_v, w_d) is three-dimension weight, where w_l represents length of roads, and we note $L \in (0,+\infty)$; w_v represents cost of consumed fuel of per kilometer, and we note $V \in (0,+\infty)$; w_d represents the value of traffic jam. As we know, the traffic time is related to the case of traffic jam. Therefore, we will use the previous result in this model. We note D = {13, 22, 3, 9, 21, 11};
According to the research of cost, we define the following aggregate function to deal with the three weights:

$$f(w_l, w_v, w_d) = \frac{w_l}{w_v} * \frac{w_d}{(w_d - 1)} \quad (11)$$

We still use the previous graph to make the experience. We establish the following data:

- $V = \{A, B, C, D, E\}$,
- $E = \{a, b, c, d, e\}$,
- The set of three-dimension weights of the 6 edges is W = {(120, 40, 13), (110, 80, 22), (100, 20, 3), (60, 15, 9), (150, 70, 21), (50, 10, 11)}.

First, we put W into the aggregate function $f(x)$. After calculating the three-dimension weights by aggregate function, we get the final w_f = {3, 1, 7, 4, 2, 5}. Finally we calculate the final result w_f with Dijkstra algorithm to get the final value from every node to other nodes. Table 4 shows the value of time from node A to other nodes.

Table 4. the result of Dijkstra(A) of graph of

B	C	D	E
3	2	4	11

According to the aggregate function we can see that the bigger the value is, the higher time spending on the path will be.

6 Conclusions

Path planning is a problem which we are always researching and probing. Although the applications of path planning are emerging in an endless stream, there is no suitable application for general users for their personal requirements. This paper establishes a multi-dimension weighted graph to exactly realize the simulation of the practical problems in the real world. We put the factors which will affect each other together to constitute the multi-dimension weighted graph, then according to the relationship among the weights to define aggregate function, which can calculate the factors to meet the different requirements of different users.

This paper establishes a framework by the combination of multi-dimension weighted graphs and aggregate functions. Then we simulate a graph of probability of traffic jam to show the process of this framework. We improve the previous related works based on weighted graph with only one-dimension weight and imperfect aggregate functions. Finally, we make it more suitable to solve the problem of path planning in the real world.

We put some other examples such as the graph of traffic accidents, the graph of traffic cost and the graph of traffic time. Intuitively, the framework can solve a lot of problems, and it can regard the previous result as a factor in this model. We also define the corresponding aggregate function to calculate the three examples above.

We will improve the second processing module, in which we will use another algorithm to deal with the final weight in our future work. We hope the framework can solve a lot of practical problems beyond the path planning.

Acknowledgements

We would like to thank Yaqi Chen for previous survey and useful comments. This work is supported by the program of the National Key Research and Development Program of China (2016YFB1000603) and the National Natural Science Foundation of China (NSFC) (61502336, 61373035). Xiaowang Zhang is supported by Tianjin Thousand Young Talents Program.

References

1. Peter Stiles and Ira Glickstein. Route Planning[C]. IEEE, 1991: 420 – 425.
2. Guanglin Zhang, Xiaomei Hu，Jianfei Chai，Lei Zhao，and Tao Yu. Summary of Path Planning Algorithm and its Application [J]. Modern Machinery, 2011, 5: 85-90.
3. Chun-long Yao, Xu Li, and Lan Shen. Weighted Directed Graph Model for Searching Optimal Travel Routes by Public Transport [J] Application Research of Computers. 2013, 30(4): 1058-1063.
4. Ran Hao. Fire Rescue based on Shortest Route Model and Its Solution Strategies [J]. China Science and Technology Information, 2010(19): 29-30.
5. Mei Feng. The Transportation Problem basasd on General Weighted Graph [J]. Mathematics in Practice and Theory, 2008, 38(9): 131-135.
6. Sergio Flesca, Filippo Furfaro, and Sergio Greco. Weighted Path Queries on Semistructured Databases[J]. Information & Computation, 2006, 204(5): 679-696.
7. Dan Stefanescu and Alex Thomo. Enhanced Regular Path Queries on Semistructured Databases[J]. Current Trends in Database Technology-EDBT-2006, 2006, 4254: 700-711.
8. Anton Dries and Siegfried Nijssen. Analyzing Graph Databases by Aggregate Queries[J]. MLG 2010, Jul-2010, 2012:37-45.
9. Reinhard Diestel. Graph theory[M]. Tsinghua University Press, Fourth Edition, 2013.
10. Thomas H.Cormen and Charles E.Leiserson. Introduction to Algorithm[M]. Machinery Industry Press, Second Edition. 2006.
11. Jelle Hellings, Bart Kuijpers, Jan Van den Bussche, and Xiaowang Zhang. Walk Logic as a Framework for Path Query Languages on Graph Databases[C]. In: Proceedings of ICDT 2013, Genoa, Italy. ACM, 117-128, 2013.
12. Xiaowang Zhang and Jan Van den Bussche. On the Power of SPARQL in Expressing Navigational Queries[J]. The Computer Journal, 58 (11): 2841-2851, 2015.
13. Hong Fang and Xiaowang Zhang. pSPARQL: A Querying Language for Probabilistic RDF (Extended Abstract)[C]. In: Proceedings of ISWC Posters and Demos 2016, Kobe, Japan.

Graph-based Jointly Modeling Entity Detection and Linking in Domain-Specific Area

Jiangtao Zhang[§†] and Juanzi Li[†]

[§]The 305th Hospital of Chinese People's Liberation Army, Beijing 100017, China
[†]Department of Computer Science and Technology,
Tsinghua University, Beijing 100084, China
zhang-jt13@mails.tsinghua.edu.cn, lijuanzi@tsinghua.edu.cn

Abstract. The current state-of-the-art Entity Detection and Linking (EDL) systems are geared towards general corpora and cannot be directly applied to the specific domain effectively due to the fact that texts in domain-specific area are often noisy and contain phrases with ambiguous meanings that easily could be recognized as entity mention by traditional EDL methods but actually should not be linked to real entities (i.e., False Entity mention (FEM)). Moreover, in most current EDL literatures, ED (Entity Detection) and EL (Entity Linking) are frequently treated as equally important but separate problems and typically performed in a pipeline architecture without considering the mutual dependency between these two tasks. Therefore, to rigorously address the domain-specific EDL problem, we propose an iterative graph-based algorithm to jointly model the ED and EL tasks in domain-specific area by capturing the local dependency of mention-to-entity and the global interdependency of entity-to-entity. We extensively evaluated the performance of proposed algorithm over a data set of real world movie comments, and the experimental results show that the proposed approach significantly outperforms the baselines and achieve 82.7% F1 score for ED and 89.0% linking accuracy for EL respectively.

Keywords: Entity Detection and Linking, False Entity Mention, Domain-specific Entity Linking, Joint Model

1 Introduction

The problem of entity linking (EL), which involves linking extracted entity mentions to corresponding Knowledge Base (KB) entries is starting from [1, 7]. However, most of existing approaches [4, 3, 18] aim at the general KBs and cannot be directly used in the domain-specific corpora. With the increasing demand for constructing and populating domain-specific KBs, domain-specific EL techniques have been emerging as an effective way to manage and query information for specific fields. The difficulty of domain-specific EL is that the entity mentions in domain-specific area are often potentially highly ambiguous and various: 1) the same mention may refer to several different entities; 2) some extracted mentions in the text are just normal phrases and should not be link to the entities (i.e., False Entity Mention(FEM)). 3) some common phrases could be real entity mentions in domain-specific corpora. Recently a few works [17, 10] begin to explore domain-specific EL task but these works do not fully consider these

issues mentioned above. Therefore we argue that domain-specific EL techniques deserve much deeper exploration.

Moreover, in most literatures, ED (Entity Detection) and EL (Entity Linking) are frequently treated as equally important but separate problems and typically performed in a pipeline architecture without considering the mutual dependency between these two tasks [9]. Therefore, in this paper, we propose a novel graph-based joint model combining ED and EL on a movie review corpora by overcoming the following challenges:

Poor mention boundaries: Although EL task can go wrong even when provided correct mentions, a large number of EL errors are caused by poor mention boundaries. Although the poor boundary problem is addressed as longest coverage matching in DBpedia spotlight and keyphrase extraction in Wikify, the boundary problem is especially severe in the domain-specific area. For the example shown in Fig.1, both "*wall*" and "*wall street*" could be potentially linked to corresponding entities (movies) and it is difficult to determine which one is correct by traditional pipeline-based approaches [6, 12, 14, 7], which just take extracted named entities as input of EL without considering the uncertainty and imperfection of the named entity extraction process. Intuitively, if we can leverage the feedback information from EL (outside knowledge information) to direct the process of ED, the issue of poor mention boundaries could be addressed. Recently, some works [11, 2, 15] perform ED and EL process jointly but their techniques do not take this issue into consideration and are best-suited for general KB instead of domain-specific KB.

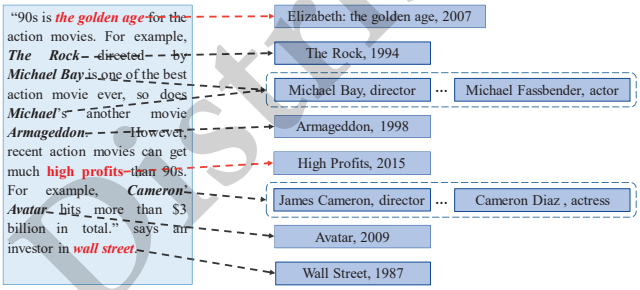


Fig. 1. An example for the task of domain-specific entity detection and linking

False Entity Mention (FEM): Many previous proposed approaches assume that each entity mention extracted from a text *should* be linked to an entity in the KB or NIL to indicate there is no matching entry [13]. However, in domain-specific area, such assumption may not be hold as in general corpora. For example, in Fig.1, the extracted mentions "*wall street*", "*golden age*" and "*high profits*" could be potentially linked to corresponding entities respectively because these mentions are the titles of entities representing movies. However, these mentions in this context are just common phrase and should not be identified as *true entity mention* (TEM). Therefore, we denote the extracted mention that *should not* be linked to any entites as the *False Entity Mention* (FEM). Notice that FEM is different with NIL (unlinkable mention) in some previous approaches [13] which indicates there is no matching entity in KB but should be linked.

Mutual Dependency: The main drawback of traditional pipeline-based approaches stems from the fact that they do not take into consideration the mutual dependency between ED and EL processes. But we argue that these two tasks are tightly coupled and the mutual information between these two tasks could be used to improve the performance of both. For example, in Fig.1, the knowledge information of linking results of "*Avatar*" and "*The Rock*" could be helpful to filter out the FEM "*the golden age*" and "*wall street*" due to the fact that the main thread of this text is talking about the action movies but the movies '*the golden age*" and "*wall street*" are not action movies. Moreover, such information of ED results is also useful for the ranking of "*Michael*" and "*Cameron*", which are the directors of movies "*The Rock*" and "*Avatar*" respectively.

Based on above observation, in this paper, we propose a new graph-based algorithm in specific domain via jointly incorporating ED with EL task. The main idea of our approach is as follows: First, we define and construct a Joint Graph based on work [5], our contribution is that the structure of our constructed graph encodes both the mention detection certainty, mention-to-entity linking confidence and the interdependent information between different entities together. Second, we calculate the initial score for each vertex and the weight of each edge in the graph. At last, we propose an iterative graph-based algorithm to step by step improve the detection accuracy and linking precision via propagating the interdependency between EL decisions.

Contributions The main contributions of this paper are summarized as follows.

- To the best of our knowledge, our research is among the first to point the poor mention boundary problem and define the important concept FEM, both of which are critical for domain-specific EDL task.
- We proposed a novel iterative graph-based algorithm that jointly models ED and EL tasks by iteratively enhancing the confidence of entity detection and certainty of entity linking, which allow us to achieve better performance than the traditional EDL methods.
- To verify the effectiveness and efficiency of our proposed method, we conducted extensive experiments on a manually annotated dataset of real world movie comments and a domain-specific KB. The experimental results show the effectiveness of our proposed approach.

The remainder of this paper is organized as follows: Section 2 describes some preliminaries and give the task definition. Section 3 presents our Joint Graph for modeling ED and EL and section 4 proposes the iterative algorithm based on constructed Joint Graph. Section 5 gives our experimental results, and Section 6 concludes.

2 Task Description

In this section, we begin by introducing some related concepts and notations. Next, we give the definition of our task.

2.1 Notations

Let $E = \{e_1, e_2, ..., e_{|E|}\}$ denotes the set of all entities of a domain-specific KB. Then we define a *mention* as a textual phrase (e.g., the "*the Rock*" in Fig.1)

which can *potentially* be linked to an entity in the domain-specific KB. Given a document d, we consider every possible n-gram (e.g. $n \leq 10$) as a *candidate mention* defined as $M = \{m_1, m_2, ..., m_{|M|}\}$. Further, we let $E(m_i) = e_{i1}, e_{i2}, ..., e_{i|E(m_i)|} \subseteq E$ denote the set of *candidate entities* which a *candidate mention* m_i could be linked to. For example, in Fig.1, the set of entities that mention "*Cameron*" could be linked to is $E($"*Cameron*"$) = \{$"*James Cameron*", "*Cameron Diaz*"$\}$. Specifically, we use $m.e \in E(m)$ to denote the true corresponding mapping entity of a mention m, i.g., the mapping entity of "*Cameron*" is "*James Cameron*".

Notice that not all *candidate mentions* should be linked to entities. For example, "*wall street*" in the Fig.1 is just a common textual phrase instead of a correct entity mention according to its context although there exists a movie named *Wall Street*. Therefore, we denote these mentions which *should not* be linked to any entities as *False Entity Mentions (FEMs)* which can be defined as $M_F = \{m_{f1}, m_{f2}, ..., m_{|M_F|}\} \subseteq M$. We also define those mentions that *should* be linked to entities in E as *True Entity Mentions(TEMs)* , which is denoted as $M_T = \{m_{t1}, m_{t2}, ..., m_{|M_T|}\} \subseteq M$. As the example shown in Fig.1, the set of TEMs is $M_T = \{$"*The Rock*", "*Michael Bay*", "*Michael*", "*Armageddon*", "*Cameron*", "*Avatar*"$\}$ and the FEMs is $M_F = \{$"*the golden age*", "*high profits*", "*wall street*"$\}$. Obviously, $M_T \cup M_F = M$.

2.2 Task Definition

The goal of our task is to map an extracted correct entity mention $m \in M$ to the corresponding entity $m.e \in E(m)$ in a domain-specific KB. In other words, given an input document d, we need to extract each TEM $m \in M_T$ and find its corresponding entity $m.e$ for each $m \in M_T$ while filtering out all FEMs from M. The input of our task is the text of a document d in a specific domain and a domain-specific KB pertaining to the domain while the output is the true entity mentions $M_T \subseteq M$ and their corresponding entities $\{m.e|\forall m \in M_T\}$.

Our task is composed by two joint parts, namely entity detection (ED) and entity linking (EL). ED is the task of identifying the boundaries, predicting the set M of given document d and extracting the true mentions $M_T \subseteq M$. EL is the task of disambiguating and linking each extracted mention $m \in M_T$ to its corresponding entity $m.e$ in the giving domain-specific KB.

3 The Joint Graph

3.1 Overview

In this subsection, we present the overview of our Joint Graph. Given a document d, We define the Joint Graph as $G = (V, A)$ where V is the vertexes set denoting all mention-to-entity pairs and E is the set of edges representing the interdependency between vertexes. Specifically, for each $m_i \in M$, and its candidate entity list $E(m_i) = \{e_{i1}, e_{i2}, ..., e_{i|E(m_i)|}\}$, the vertexes are formulated as a set $V = \{v_k = (m_i, e_{i,j})|\forall e_{i,j} \in E(m_i), \forall m_i \in M, 1 \leq k \leq |V|\}$. Each vertex v_k in the graph is associated with an score $s(v_k)$ indicating the strength of detection certainty of m_i and linking confidence between m_i and $e_{i,j}$. For each pair of vertexes $\langle v_k, v_l \rangle$ in the graph, we add an undirected edge $\langle v_k, v_l \rangle$ to A, with a weight $w(\langle v_k, v_l \rangle)$ indicating the strength of interdependency between their entities. In this way, two types of dependencies are modeled in the Joint Graph:

1. Local dependency between mention and candidate entity
   In Joint Graph, the dependency between an entity mention m_i and a candidate entity $e_{i,j}$ is encoded as the score $s(v_k)$ of the vertex $v_k = (m_i, e_{i,j})$.
2. Global Interdependency between EL decisions
   By connecting candidate entities using the edges, the interdependency between EL decisions is encoded into the structure of the Joint Graph. In this way, the Joint graph allows us to deduce and use indirect and implicit dependency between different EL decisions. For example, the mention "*The Rock*" is related to the entity "*The Rock, 1994*", which in turn is related successively to the entity "*Michael Bay, director*". As a result, the relationship between "*Michael*" and "*Michael Bay, director*" could be strengthened while the relationship between "*Michael*" and "*Michael Fassbender, actor*" will be weakened.

For illustration, Fig. 2 shows the Joint Graph representation of the EDL problem in Example 1. To ease the representation, we do not draw all edges in the Joint Graph. From Fig. 2 , we can see that the score of the TEMs vertex is high and there is a strong semantic relatedness between any two of the true mapping entities of TEMs. On the contrary, between the vertexes of TEM and FEM, the semantic relatedness is weak, which demonstrates that the Joint Graph can effectively model mention-to-entity linking confidence as vertex scores and entity-to-entity interdependency as edge weights.

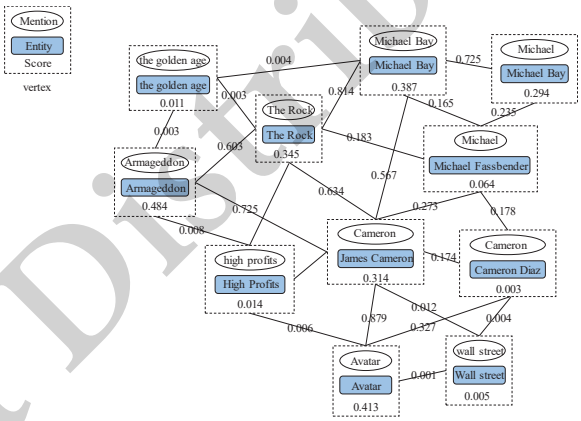


Fig. 2. The Joint Graph of Example 1

3.2 Graph Construction

Before we construct the Joint Graph, we need to generate candidate mentions M and candidate entities $E(m)$ in the given document d first. Here, we consider every possible n-gram (e.g. $n \leq 10$) in d as a candidate mention and adopt the construction method described in [17] to overgenerate candidate mentions and entities.

The construction of Joint Graph takes two steps: vertexes generation and vertexes connection.

Vertexes generation: Each mention $m_i \in M$ is paired with its every candidate entity $e_{i,j} \in E(m_i)$ in d to form a vertex in the Joint Graph. Then, each vertex $v_k = (m_i, e_{i,j})$ will be assigned to a score $s(v_k)$ to indicate the mention detection certainty of m_i and mention-to-entity linking confidence between $(m_i, e_{i,j})$, which will be introduced in Section 4.1.

Vertexes Connection: Next, we add the interdependent edge to the constructed vertexes. For each vertexes pair $\langle v_k, v_l \rangle$, $v_k = (m_i, e_{i,j}), v_l = (m_p, e_{p,q})$ in Joint Graph, if there is semantic relatedness between their entities (i.e. $e_{i,j}$ and $e_{p,q}$), we add an edge with weight $w(\langle v_k, v_l \rangle)$ between them to indicate their interdependent strength. Notice that Edges are not drawn between different nodes for the same mention since only one of candidate entities for the same mention may be the *true* mapping entity.

There has been several research which focused on computing the relatedness between entities [19, 15] . In our approach, we adopt the Wikipedia Link-based Measure (WLM) algorithm [8] to calculate the relatedness of two entities $e_{i,j}$ and $e_{p,q}$. The WLM is based on the Wikipedia's hyperlink structure. The basic idea of this measure is that two Wikipedia articles are considered to be semantically related if there are many Wikipedia articles that link to both. We apply the same algorithm to our KB: Given two entity e_i and e_j, we define the semantic relatedness between them as $WLM(e_i, e_j) = 1 - \frac{\log(\max(|E_i|,|E_j|)) - \log(|E_i \cap E_j|)}{\log(|W|) - \log(\min(|E_i|,|E_j|))}$, where E_i and E_j are the sets of entities that link to e_i and e_j respectively in the KB, and W is the set of all entities in KB. Then we have $w(\langle v_k, v_l \rangle) = WLM(e_{i,j}, e_{p,q})$.

We show the example of semantic relatedness between vertexes in Fig.2. The value shown beside each edge in Fig.2 is the edge weight calculated using WLM. From Fig.2, we can see that there is a strong relatedness relationship between any two of the *true* mapping entities.

4 Graph-based iteration algorithm

4.1 Initial Score

In this section ,we elaborate our iterative graph-based algorithm. First, each vertex $v_k = (m_i, e_{i,j})$ in the Joint Graph will be assigned with an initial score $s(v_k)$ indicating the confidence of a candidate mention being a TEM and the strength of a mention being linked to a candidate entity by leveraging the following four features.

Popularity: Most of current research [13, 17], et al., use the popularity as an important feature in EL task which indicates popularity of a mention being linked to an entity by leveraging the count information from KB. Therefore, we formalize the popularity of a vertex $v_k = (m_i, e_{i,j})$ as follows:

$$pop(v_k) = \frac{count_{m_i}(e_{i,j})}{\sum_{e_{i,j} \in E(m_i)} count_{m_i}(e_{i,j})}, v_k = (m_i, e_{i,j}) \quad (1)$$

where $count_{m_i}(e_{i,j})$ is defined as the number of times that entity $e_{i,j} \in E(m_i)$ is linked by the mention m_i.

Linkable probability: We also leverage the count information in the KB to get the linkable probability of a mention indicating the probability that a mention m_i is a TEM, which can be formalized as follows:

$$lp(v_k) = \frac{\sum_{e_{i,j} \in E(m_i)} count_{m_i}(e_{i,j})}{count(m_i)} \tag{2}$$

where $count_{m_i}(e_{i,j})$ is defined as the number of times that an entity $e_{i,j} \in E(m_i)$ is actually linked by the mention $m_i \in M$. $count(m_i)$ is the total number of appearances of mention m_i.

Coherence: One would expect that entities mentioned in the same context are likely to be topically coherent, i.e. they are likely semantic related [16]. Therefore, we exploit this semantic relatedness between entities in the document d to define the coherence feature $coh(v_k)$ of a vertex $v_k = (m_i, e_{i,j})$ as the average value of the semantic similarity between each context entity e_c and its entity $e_{i,j}$.

$$coh(v_k) = \frac{\sum_{e_c \in C_E(m_i)} SmtRel(e_c, e_{i,j})}{|C_E(m_i)|} \tag{3}$$

where $C_E(m_i)$ means the set of context entities which co-occur with m_i in the same document. In our algorithm, we also adopt WLM to get the semantic similarity $SmtRel(e_c, e_{i,j})$.

Context similarity: It has been an effective way to use the context information to perform entity disambiguation. Therefore, we define the context similarity $c_s(v_k)$ of vertex $v_k = (m_i, e_{i,j})$ as the similarity between the context around m_i and the full text of $e_{i,j}$ via leveraging Jaccord algorithm.

$$cs(v_k) = Jaccard(S_m, S_e) = \frac{|S_m \cap S_e|}{|S_m \cup S_e|} \tag{4}$$

Where S_m denotes the bag of words for context of m_i while S_e means the bag of words for the full text of $e_{i,j}$.

Based on these features illustrated above, we assign the initial score $s(v_k)$ for each vertex $v_k = (m_i, e_{i,j}) \in V$ as the weighted sum of these features as follows:

$$s(v_k) = \overrightarrow{W} \cdot \overrightarrow{F} \tag{5}$$

where $\overrightarrow{F} = \{pop(v_k), lp(v_k), coh(v_k), cs(v_k)\}$ is a feature vector, and $\overrightarrow{W} = \{w_1, w_2, w_3, w_4\}$ is a weight vector, $\sum w_i = 1$. The weight vector $\overrightarrow{W}$ can be easily learned by supervised machine learning technique such as SVM on a training data set. Obviously, the score of a vertex $v_k = (m_i, e_{i,j})$ indicates the certainty of m_i being a TEM and confidence of m_i being linked to $e_{i,j}$.

4.2 Iterative Algorithm

In order to simplify the description of our proposed iterative graph-based algorithm, we first introduce the following three notations for our graph-based algorithm:

S: The initial score vector $S = \{s_1, s_2, ..., s_{|V|}\}$, where $s_k = s(v_k)$.

S_f: The final score vector $S_f = \{s_{f1}, s_{f2}, ..., s_{f|V|}\}$, where $s_{fk} = s_f(v_k)$. To ease the presentation, we denote the final score vector S_f exactly after round r iteration as S_f^r.

B: we define the adjacency matrix of the Joint Graph G as the iteration matrix B. B is a $|V| \times |V|$ matrix, where the value of element $B[k, l]$ is the edge weight between vertex v_k and v_l.

To compute the final score vector S_f, we first set its initial value s_f^0 as the initial score vector S, i.e., $S_f^0 = S$. Then we can update the final score vector S_f in an iteration manner as follows,

$$S_f^{r+1} = \lambda S + (1 - \lambda) B S_f^r \tag{6}$$

where $\lambda \in [0, 1]$ is the relative importance fraction of the two parts, of which appropriate value will be evaluated in section 5. From this equation, we can see that our algorithm combines information from the initial score vector S and the interdependent information between vertexes by updating the final score vector iteratively until the final score stabilizes within a certain iteration steps which is set to 10 in our experiment.

At last, we can choose the mapping entity $m_i.e$ for entity mention m_i as:

$$m_i.e = \underset{e_{i,j} \in E(m_i)}{\arg\max}\ s_f(v_k), v_k = (m_i, e_{i,j}), \forall e_{i,j} \in E(m_i) \tag{7}$$

Since there are FEMs in the given document, we have to deal with this problem by validating whether the returned entity $m_i.e$ with highest score according to Equation 7 is a correct mapping entity for mention m_i. We adopt a simple method: learning a FEM threshold τ to validate the highest score entity. If the final score $s_f(m_i.e)$ is greater than the FEM threshold τ, we return $m_i.e$ as the correct mapping entity for entity mention m_i, otherwise we return it as FEM and treat it as common phrase. The FEM threshold τ is learned by linear search based on the training data set, which is set to 0.25 in our experiment.

5 Experiments and Evaluation

To evaluate the effectiveness and efficiency of our proposed approach, we present an extensive experimental study in this section. All the programs were implemented in Python and all the experiments were conducted on a server (with four 2.7GHz CPU cores, 1024GB memory, Ubuntu 13.10).

Data Set We conduct experiments on a gold standard data set for our task and adopt the Keg-Movie-Ontology (KMO) as the target domain-specific KB which have been used in [17]. The KMO, constructed by knowledge engineering laboratory of Tsinghua University, is a high quality KB, which integrates several English and Chinese movie data sources from LinkedIMDB, Douban and Baidu Baike, and contains 23 oncepts, 91 properties, more than 700,000 entities and 10 million triples. The gold standard data set contains user comments from several well established websites in China, such as 163, sina, sohu and tianya, etc which have been manually annotated.

Table 1 lists some statistical data of the gold standard data set. From the table we can see that there are 842 comments, which include 2529 FEMs and 11848 TEMs. The number of all candidate entities is 42105. Average number of mentions (includes TEMs and FEMs) in one comment and candidate entities per candidate mention is 17.05 and 2.92, respectively.

Documents	$\|FEMs\|$	$\|TEMs\|$	CEs	$\overline{\|M\|}$	$\overline{\|E(m)\|}$
843	2529	11848	42105	17.05	2.92

Table 1. Statistical data of the user data set

Baseline Methods Due to the fact that the traditional approaches could not directly apply on our data set and KMO, we created two classic baselines employed the traditional pipeline architecture that takes extracted entity mentions as the input to the following EL task. Moreover, in order to fairly evaluate the effectiveness of our proposed approach, we also adopt the method used in [17], named IJM(Interactive Joint Model) as another baseline.

Prior Probability-based method (POP). In this baseline, we only use linkable probability and popularity for ED and EL respectively. We set a threshold and only retain the mention whose linkable probability is higher than the threshold which is set to 0.045 in our experiment. Then we choose the the entity with the highest popularity among all the candidate entities as the mapping entity for this entity mention.

Context Similarity-based method: $(CSim)$. We constructed a context vector for each mention and a profile vector for each candidate entity (e.g. using TFIDF). Then we measure the similarity of these two vectors for each pair of a mention and a candidate entity (e.g. cosine distance). Finally, the entity with the highest similarity is considered as the mapping entity for the mention. We also set a threshed and only retain the mention whose highest similarity score is larger than the pre-set threshold which is set to 0.087.

Interactive Joint Model: (IJM). The method IJM, proposed in [17], used an interactive framework between ED and EL tasks to improve the performance of both tasks iteratively via updating the values of features of these two tasks in an interactive manner.

Evaluation Metrics Our task involves jointly modeling ED and EL processes which influence each other, therefore we also adopt the evaluation metrics used in [17], i.e.,

- **ED:** precision, recall and F1-measure;
- **EL:** accuracy over correctly recognized entities;
- **Overall ED+EL:** precision, recall and f-measure; the precision/recall is computed as the product of the NER precision/recall by the EL accuracy.

Influence of Fraction Factor λ the $\lambda \in [0, 1]$ is the relative fraction factor between the initial score and score of last iteration. From equation 6 we can see that if $\lambda = 0$, the iteration only considers interdependency propagation. If $\lambda = 1$, there is no iteration and only the initial score works. Thus, the value of λ indicates the balance between the local dependency of mention-to-entity and global interdependency of entity-to-entity. We evaluate the relationship between the value of λ and the overall F1 score, as indicated in Fig.3. From the figure we can see that when $\lambda = 0.2$, the F1 get the highest score. Therefore, in our experiment, the value of λ is set to the 0.2.

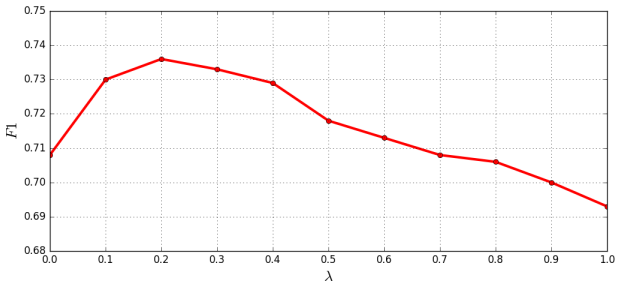


Fig. 3. F1 versus λ

Result and Analysis In order to evaluate the effectiveness of our jointly iterative graph-based algorithm, we configured the proposed approach into four different settings:

- *Fixed Weights+No Iteration (FW+NoI):* We don't use the machine learning method to train the weights. We assume that all features have the same weight, that is, the weight of all features is 0.25. Furthermore, we don't perform the iteration, i.e., $\lambda = 1$.
- *Initial Score+No Iteration (IS+NoI):* We use the initial scores computed by the Equation 5 without performing the iteration, i.e., $\lambda = 1$ in Equation 6.
- *Random Initial Score+Iteration (RIS+I):* We use random initial scores instead of the initial scores computed by Equation 5 and perform the iteration according to Equation 6.
- *Initial Score + Iteration (IS+I):* We use the initial scores computed by the Equation 5 and perform the iteration according to Equation 6.

Approach	Overall ED + EL			EL	ED		
	precision	recall	F1	accuracy	precision	recall	F1
POP	0.615	0.509	0.557	0.792	0.776	0.643	0.703
CSim	0.597	0.590	0.594	0.825	0.724	0.715	0.719
IJM	0.741	0.690	0.714	0.875	0.847	0.788	0.816
FW+NoI	0.665	0.648	0.656	0.851	0.781	0.762	0.771
IS+NoI	0.717	0.670	0.693	0.866	0.828	0.774	0.800
RIS+I	0.727	0.660	0.692	0.859	0.846	0.768	0.805
IS+I	**0.764**	**0.710**	**0.736**	**0.890**	**0.858**	**0.798**	**0.827**

Table 2. Comparison of experiment results

Table 2 gives the comparison of our proposed approach and all other methods mentioned above. The experimental results demonstrate that different configurations of our proposed graph-based algorithm significantly outperforms the two baseline methods (i.e., *POP* and *CSim*) and our final approach *IS+I* also outperforms the *IJM* proposed in [17], which demonstrates the effectiveness of our proposed model.

In general, we can see that our proposed algorithm achieves high accuracy for EL in all configurations, which shows that our algorithm is very effective for EL task. The interdependency between the referent entities in the same document can provide critical evidence to the EL decision.

For the assessment of the *POP* baseline, obviously, the probability of being a TEM is high for the mention with high linkable probability. However, due to *POP* uses the method of simply setting a threshold to exclude the mention with small linkable probability, *POP* gets a high precision but low recall. For the *CSim* baseline, because it considers context rather than prior probability, the recall of *CSim* is higher than *POP*, but the precision of *CSim* is damaged because it also introduces the FEMs.

Additionally, for different configurations of our algorithm, the performance of *FW+NoI* improves both ED and EL performance than baselines because four features are considered not merely prior probability. The performance of *IS+NoI* further improves as it considers the importance of different features by leveraging machine learning techniques. Meanwhile, the key point of the *RIS+I* is to investigate the influence of the iteration without considering the initial scores. The results indicate that overall precision further improves due to the fact that iteration exclude FEMs effectively while recall falls because no feature is considered.

Moreover, although *IJM* consider the interaction of ED and EL and use an interactive framework to jointly model these two tasks, our proposed method *IS+I* outperforms the *IJM* due to the fact that it decodes both the local dependency of mention-to-entity and global interdependency of entity-to-entity into a joint graph and use a similarity-flooding-like algorithm to propagate the dependency.

Finally, as expected, by modeling and exploiting local dependency of mention-to-entity and global interdependency of entity-to-entity, the final configuration of our method *IS+I* gets the highest performance in terms of overall precision and recall which achieved 32% F1 improvement compared with the baseline *POP*, 24% F1 improvement compared with the baseline *CSim* and 3% F1 improvement compared with *IJM*.

6 Conclusion

The traditional EDL systems aim at general domain area. An unfortunate effect of this aim is that such generalist systems are often disappoint when they are applied to domain-specific area. Furthermore, most of existing EDL techniques ignore examining the interdependency of entities extraction and linking. In this paper, we proposed and evaluated an iteratively joint graph-based algorithm to model the ED and EL task by capturing the local dependency of mention-to-entity and global interdependency of entity-to-entity. The experiment results show that our proposed approach offers competitive performance to the three baseline systems, which indicate that it will be very useful for the domain-specific applications.

Acknowledgments The work is supported by 973 Program (No. 2014CB340504), NSFC-ANR (No. 61261130588), Tsinghua University Initiative Scientific Research Program (No.20131089256), Science and Technology Support Program (No. 2014BAK04B00), and THU-NUS NExT Co-Lab.

References

1. Bunescu, R., Pasca, M.: Using encyclopedic knowledge for named entity disambiguation. In: Proceesings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06). pp. 9–16 (2006)
2. Guo, S., Chang, M.W., Kiciman, E.: To link or not to link? a study on end-to-end tweet entity linking. In: HLT-NAACL. pp. 1020–1030 (2013)
3. Han, X., Sun, L.: A generative entity-mention model for linking entities with knowledge base. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1. pp. 945–954 (2011)
4. Han, X., Sun, L.: An entity-topic model for entity linking. In: EMNLP-CoNLL '12. pp. 105–115 (2012)
5. Han, X., Sun, L., Zhao, J.: Collective entity linking in web text: A graph-based method. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 765–774 (2011)
6. Lin, T., Mausam, Etzioni, O.: Entity linking at web scale. In: AKBC-WEKEX '12. pp. 84–88 (2012)
7. Mihalcea, R., Csomai, A.: Wikify!: Linking documents to encyclopedic knowledge. In: Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management. pp. 233–242 (2007)
8. Milne, D., Witten, I.H.: Learning to link with wikipedia. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management. pp. 509–518 (2008)
9. Nguyen, D., Theobald, M., Weikum, G.: J-nerd: Joint named entity recognition and disambiguation with rich linguistic features. Transactions of the Association for Computational Linguistics pp. 215–229 (2016)
10. Olieman, A., Kamps, J., Marx, M., Nusselder, A.: A hybrid approach to domain-specific entity linking. CoRR (2015)
11. Pu, K.Q., Hassanzadeh, O., Drake, R., Miller, R.J.: Online annotation of text streams with structured entities. In: CIKM. pp. 29–38 (2010)
12. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to wikipedia. In: HLT. pp. 1375–1384 (2011)
13. Shen, W., Wang, J., Jiawei, H.: Entity linking with a knowledge base: Issues, techniques, and solutions. In: IEEE Transactions on Knowledge and Data Engineering. pp. 443 – 460 (2014)
14. Sil, A., Cronin, E., Nie, P., Yang, Y., Popescu, A.M., Yates, A.: Linking named entities to any database. In: EMNLP-CoNLL. pp. 116–127 (2012)
15. Sil, A., Yates, A.: Re-ranking for joint named-entity recognition and linking. In: CIKM. pp. 2369–2374 (2013)
16. Sil, A., Yates, A.: Re-ranking for joint named-entity recognition and linking. In: Proceedings of the 22Nd ACM International Conference on Information Knowledge Management. pp. 2369–2374 (2013)
17. Zhang, J., Li, J., Li, X.L., Shi, Y., Li, J., Wang, Z.: Domain-specific entity linking via fake named entity detection. In: DASFAA. pp. 101–116 (2016)
18. Zhang, W., Sim, Y.C., Su, J., Tan, C.L.: Entity linking with effective acronym expansion, instance selection and topic modeling. In: IJCAI'11. pp. 1909–1914 (2011)
19. Zhang, W., Su, J., Tan, C.L., Wang, W.T.: Entity linking leveraging: Automatically generated annotation. In: Proceedings of the 23rd International Conference on Computational Linguistics. pp. 1290–1298 (2010)

Link Prediction via Mining Markov Logic Formulas to Improve Social Recommendation

Zhuoyu Wei, Jun Zhao, Kang Liu, and Shizhu He

National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, 100190, China
{zhuoyu.wei,jzhao,kliu,shizhu.he}@nlpr.ia.ac.cn

Abstract. Social networks have been a main way to obtain information in recent years, but the huge amount of information obstructs people from obtaining something that they are really interested in. Social recommendation system is introduced to solve this problem and brings a new challenge of predicting peoples preferences. In a graph view, social recommendation can be viewed as link prediction task on the social graph. Therefore, some link prediction technique can apply to social recommendation. In this paper, we propose a novel approach to bring logic formulas in social recommendation system and it can improve the accuracy of recommendations. This approach is made up of two parts: (1) It treats the whole social network with kinds of attributes as a semantic network, and finds frequent structures as logic formulas via random graph algorithms. (2) It builds a Markov Logic Network to model logic formulas, attaches weights to each of them to measure formulas contributions, and then learns the weights discriminatively from training data. In addition, the formulas with weights can be viewed as the reason why people should accept a specific recommendation, and supplying it for people may increase the probability of people accepting the recommendation. We carry out several experiments to explore and analyze the effects of various factors of our method on recommendation results, and get the final method to compare with baselines.

1 Introduction

Social networks have been a main way to obtain information in recent years. People get the latest news, knowledge of specific fields, or even just stories and jokes from them. There is a relationship between users called $Follow(follower, followee)$, which means the follower would like to pay attention to the followee or the content published by the followee. What kind of followees people followed determines what kind of messages they can get from social networks. Therefore, social recommend task can be view as predicting missing link on the social network.

An excellent social recommendation system can rescue people from searching and choosing, by bringing what they are interested in or helping them build new interests. At the beginning, the recommendation methods of e-commerce were

ported to social networks but the performance was not satisfactory. To improve the accuracy of recommendation, researchers propose a variety of solutions or techniques, such as taking explicit or implicit information into account, analyzing the user's social behaviors, and so on. They make the social recommendation techniques have a great development and go on improving apace. At the same time, many popular social networks, such as Twitter and Tencent Weibo, provided the reason why a user will accept a specific recommendation. For example, the reason can be that the user has 3 friends following the followee, or there is a high degree of similarity between tweets published by the user and the followee. Usually, labeling reasons to recommendations can increase the probability of users accept them.

However, there are still several problems which current recommended methods cannot solve well. The first and most headachy one is the cold start problem: Too little histories of new users make methods based on collaborative filtering failure. Secondly, heterogeneous attributes and relations cannot be modeled well nor used effectively. Although we have introduced a lot of features, such as users' age, tweets' keywords, social relations, location, accepted time, even the current mood of the users, few methods can effectively use them. Some methods assume all features are independent, which missed relevance between different types of relations and attributes. Others methods unite each two or more features to build new features and most of them are useless, which produce a huge feature space and make model extremely complex. Thirdly, the reasons why users will accept recommendations are generated from templates or rules listed manually, which takes a lot of time and may miss some cases.

In this paper, we propose a novel approach to bring logic formulas in social recommendation system and try to solve the above problems. Our method inherits the graph-based structure of the social works, and adds users' attributes to the graph. Distinguished from the Social-Attribute Network, the graph labels different semantic concepts to different types of nodes and edges. If we only take the concepts of nodes and edges into account, there are a lot of same structures(especially loops) on the graph. These conceptual structures can be viewed as frequent logic formulas from the perspective of first-order logic. We propose the Randomly Finding Loops algorithm to find these frequent logic formulas on the graph. Then we use the Markov Logic Network(MLN) to model directly logic formulas by treating each edges as a random variable and attach weights to formulas, rather than constructing, grounding a MLN and learning its structures and weights in the traditional way[24]. Finally, we construct queries with each user and followees recommended to it in the training data set, and learn the weights discriminatively.

We carried out several experiments on the Tencent Weibo data set and subsets from KDD-Cup'12 track 1[21] to explore and analyze the effects of various factors of our approach on recommendation results, and then compare our approach with several baselines.

The major contributions of the paper are as follow: (1) We are the first to bring logic formulas in social recommendation system, and use them to repre-

sent the relations between social relations and kinds of attributes. (2) Distinguished from conventional methods based on random walk or grounding the MLN, we combine the advantages of both approaches by attaching weights to loops(formulas) to build the MLN directly, and learn the weights discriminatively rather than assigning value to them. (3) Our method generates reasons why users will accept recommendations automatically rather than manually.

The remainder of this paper is organized as follows. Section 2 introduces previous methods that are related to this work. Section 3 details how to use MLN to model social recommendation task. Section 3 details the Randomly Finding Loop Algorithm. Then experimental results are presented in Section 5, followed by the conclusion in Section 6.

2 Related Work

2.1 Social Recommendation Technique

Social recommendation techniques are different from traditional recommendation for e-commerce. It need model both uses' interests and items'(objects recommended) characters, and handle relations in social networks. Some traditional recommendation algorithms simply based on contents[3][4] or collaborative filtering[25] don't work well, because they cannot deal with heterogeneous data. At present there are already a number of models or methods who can handle such heterogeneous data containing attributes, relations and build a unify system to make recommendation. These methods can be divided into two categories, matrix factorization model[6][7] and graph-based model[8][2][27]. The former captures implicit relations between users and items, and merges all kinds of attributes,relations, even feedbacks[20] via factor vectors. The Matrix Factorization Model is the state-of-the-art method for collaborative filtering and collaborative ranking[16][17]. It uses factor vector to represent attributes, links, even users and items themselves, then the inner product of one user's vectors and one item's vectors is treated as final rating score. But it can only capture direct relations between factor vectors and create too many variables, which can lead to over-fitting and a long training time. Factorization machines[22][23] as an expansion of factorization model, it can handle more than two variables' interactions; While Karatzoglou et al.[12] solves the problem by expanding to the tensor decomposition approach. In this way, they must have more useless variables, which is a similar but more serious problem. While the latter transforms attributes to edges and combines into a heterogeneous graph, then applies random walking[5][2], propagation[13], paths finding, or just search techniques on it. Neighborhood-based methods are special cases of graph-based model when we only take 2-length paths into account. Item-based methods[25], user-based methods[26] and similarity calculating methods[9] all belong to neighborhood-based methods. For more general graph-based models: Social Attribute Network Model[8][28][29] creates an augmented network by adding attributes as nodes and undirect link between users and attributes; Methods based on propagations[11][30] defines a type of values and propagates them on the social relations

graph. These methods can easily get useful multivariable interactions by randomly finding longer paths. However, they don't distinguish between types of paths and assign weights to paths directly accounting to degrees of nodes. We build a social semantic graph and use logic formulas to distinguish types and obtain weights from the learning process.

2.2 Markov Logic Network

Markov Logic Network(MLN) was first proposed by Pedro Domingos[24] formally. MLNs conbimes probability and logic by attaching weights to first-order formulas, and viewing these as templates for features of Markov Networks, and they can be applied to link prediction task. Recommendation systems can be views as practical instances of the the link prediction task. MLNs can find the relaiton paths called formulas. They are treated as a template of world together, and allocate weights attached to formulas to maximize the likelihood of the real world. Although MLNs can easily represent entities, attributes, and relationships in a social network, they rarely are applied to social recommendation system currently for its extreme computational complexity of learning or inference. Many techniques have been proposed to speed up this process: The discriminative learning methods[10][19] are used to decrease the number of random variables; Stanley Kok[14] clusters entities and relations before find formulas, and tries to find longer formulas by randomly finding motifs[15]. Though algorithm's efficiency has been improved, these methods still have to ground all relations with all entities. The grounding process spends a huge amount of time, even makes the problem cannot be computed. We abandon the grounding process and approximate the likelihood for learning by heuristic and stochastic sampling mechanism. This idea comes from finding frequent patterns algorithm on graph, such as Musk[1] and simpling DNF patterns[18], and it can be translate to learn structures of MLNs. This approach makes it possible to apply MLNs on large social recommendation data sets.

3 MLN For Social Recommendation

3.1 Building Social Semantic Graph

In the social network, we have a set of users noted as U and a set of attribute values of the users noted as A. In order to construct discriminative task, we create a subset of users noted as $U_i (U_i \subseteq U)$ for each User i in U as its alternative recommendations set, and User i can accept or ignore these recommendations. Then User i with each recommended user from U_i can be combined into a pair, noted as $Accept(user_i, user_r)$ in the form of triplets. They are called queries whose value is true when accepted or false when ignored. Our task is to predict the possibility of each query is true.

We build a direct semantic graph $G(N, E, C, R)$ whose nodes and edges are label types, to model U, A and all kinds of relations between them. N is the

set of nodes and C is the set of nodes' types. All users in the social network and their attribute values are treated as nodes in N, and each nodes has a type in C, called *Concept*. E is the set of direct edges and R is the set of edges' types. All kinds of relations in social networks are treated as direct edges in E, and each edges also has a type in R, called *Relation*. The relation set contains social relations and action relations (e.g. *Retweet* action, *Comment* action and *At* action). From the perspective of Markov Networks, the triplets are treated as random variables and they are the nodes in the Markov Network. Therefore, we can introduce the MLN technology to create templates of the social semantic graph[24], which is the theoretical support of our approach.

In detail, the node set N should contain follow parts: (1) All users from U are added to N as nodes, and their concept is *user*; (2) All attribute values in A are treated as nodes with their attribute names as nodes' concepts. For example, *male* and $female$ are nodes, whose concept is *gender*; Decades, such as $1990s$ and $2000s$, are nodes, whose concept is *birthyear*; Keywords from users' statuses and comments are also nodes, whose concept is *keyword*.

Analogously, the edge set E contains follow parts: (1)The direct edge from *userA* to *userB* should be added , if userA has followed userB, noted as *Follow(userA, userB)*. (2)The own relations or mutex relations from U to A are added to E as edges, such as $Gender(user, male)$, $GenderFalse(user, female)$, $BirthYear(user, 1990s)$, $Keyword(user, keyword)$ and so on.

3.2 Markov Logic Formula

If we want to estimate the possibility of that *Accept*(*user*, *user*) is true, we need treat it as a query, and then generate logic formulas containing the query. Meanwhile we need count the times of each logic formulas appearing. Here we show some examples for the logic formulas.

- $Follow(u_A, u_D) \wedge Follow(u_B, u_D) \wedge Follow(u_B, u_C) \Rightarrow Accept(u_A, u_C)$, $u_A \rightarrow u_D \leftarrow u_B \rightarrow u_C \leftarrow u_A$
- $Keyword(u_A, k_1) \wedge Keyword(u_C, k_1) \Rightarrow Accept(u_A, u_C)$, $u_A \rightarrow k_1 \leftarrow u_C \leftarrow u_A$
- $Keyword(u_A, k_2) \wedge Keyword(u_C, k_2) \Rightarrow Accept(u_A, u_C)$, $u_A \rightarrow k_2 \leftarrow u_C \leftarrow u_A$

All triplets on the left side of '$\Rightarrow$' are evidences of the query on the right side. We treat triplets as random variables, where the probability of evidences are true is 1 while the probability of query need to be estimated. Then we build a clique with all triplets in the same formula, and use the MLN to model it. The triplets are atomic, and we assum the atoms in evidence set are independent of the query. For MLNs, this means that the Markov Blanket of a query only contains evidence atoms[10].

From the perspective of the social semantic graph, these above are all entitative loops with u_A as start point and end point, examples of which are displayed behind the above logic formulas. Such entitative loops, or called entitative formulas, can be generated by running the finding loops algorithm on the social

semantic graph, which will be detailed in Section 4. What's more, the the processes of finding loops for different queries are independent of each other, so we parallelize these processes to make full use of computing resource.

Algorithm 1: Process Framework

PROCESS FRAMEWORK

1 **Build** static Social Attribute Graph
2 Start to **Maintain** global formulas set Υ
3 **for** each y in QuerySet **Y**
4 **FindLoop** for query y
5 **Count Locally** logic formulas
6 **BuildDataPoint** for query y
7 **Learn** weights for global formulas with DataPoints

Replace nodes in these entitative loops with their concepts, and we get conceptual loops. The formula set Υ are made up with these conceptual loops in MLNs[24], and a weight is attached to each of them. Finally, the weight vector $\mathbf{w}$ can be obtained from discriminative learning with train recommendations and it plays a decisive role in discriminating for test recommendations[10].

3.3 Discriminative Weight Learning

In this sub-section, we learn the weights of all conceptual formulas. We maximize the conditional log-likelihood(CLL) of the MLN with regularization, which is classic model of discriminative learning MLNs.

We create a query for each pair of a User u and a recommended User i from the alternative recommendations set U_i, noted as $Accept(u, i)$. We put all such queries into the query set Y, and run the finding loops algorithm for them. The conceptual formulas and counts obtained from the process are treated as features of the data point for the query y. In this way, we get the $|Y|$ data points as training data. Therefore, the CLL of Y is expressed as following under the evidence set X:

$$CLL = \sum_{k=1}^{n} \log P(Y_k = y_k | X = x) \tag{1}$$

Where k means the kth data point and Y_k is the kth query's label, whose value is 1 or 0 and noted as y_k, representing whether the recommendation is accepted. And,

$$P(Y_k = y_k | X = x) = \frac{e^{\sum_{j \in \Upsilon_{Y_k}} w_j n_j(x, y[Y_k = y_k])}}{e^{\sum_{j \in \Upsilon_{Y_k}} w_j n_j(x, y[Y_k = 0])} + e^{\sum_{j \in \Upsilon_{Y_k}} w_j n_j(x, y[Y_k = 1])}} \tag{2}$$

Where Υ_{Y_k} is the set of conceptual formulas with at least one entitative loop be found in finding loops for data point k. w_j is the weight of the jth formula, whose index j is global. $n_j(x, y[Y_k = y_k]))$ is the number of the jth conceptual formula's true entitative loops, and similarly for $n_j(x, y[Y_k = 0]))$ and $n_j(x, y[Y_k = 1]))$.

Then reviewing the second and third logic formulas in sub-section 3.2, we find the only difference between them is linked by different keywords. We need

take the difference into account, because different entities brings different contributions. We assign a value for each entitative relation, and different types are calculated in different ways: (1)For *Follow* and *Accept*, their values are still 1; (2)For three action relations, *At*, *Retweet*, *Comment*, we normalize the counts of action relations(i.e. $At(user_A, user_B)$'s count) by the total action counts of the user, and take them as values for these entitative relations; (3)$Keyword$ relations' values are set to their $tf - idf$ or other token-document values. After defining the values of edges, we use the following equation to calculate the value for a loop.

$$v(L) = \sqrt[n]{\prod_{i=1}^{n} v(E_i)} \tag{3}$$

Where n is the length of the loop. The equation eliminated the effect caused by different lengths of loops. And the $P(Y_k = y_k|X = x)$ equation (2) changes into:

$$P(Y_k = y_k|X = x) = \frac{e^{\sum_{j \in \Upsilon_{Y_k}} w_j V_j(x, y[Y_k = y_k])}}{e^{\sum_{j \in \Upsilon_{Y_k}} w_j V_j(x, y[Y_k = 0])} + e^{\sum_{j \in \Upsilon_{Y_k}} w_j V_j(x, y[Y_k = 1])}} \tag{4}$$

Where we use V_j to take the place of n_j, and V_j is the sum of the jth conceptual formula's entitative loops' values $v(L)$.

We take the negative CLL as the loss function and minimize it. Add the L2-regularization as an additional term, C as the regularization coefficient, and the loss function changes to $L(\mathbf{w}) = CLL + C\|\mathbf{w}\|_1$. The main process is sketched in Algorithm 1.

4 Randomly Finding Loops

4.1 Find loops For A User

For a query, *Accept*(*user*, *recommend*), we want to find formulas like this: $Relation_{\pm}$ $(user, node_1) \wedge Relation_{\pm}(\ node_1, node_2) \wedge ..... \wedge Relation_{\pm}$ $(node_{n-1},$ $recommend) \Rightarrow Accept(\ user, recommend)$. Where $Relation_{\pm}(node_1, node_2)$ represents one of the two direct edges, $Relation(node_1, node_2)$ and $Relation(node_2,$ $node_1)$. Remove edges' relations and represent the loop as a sequence of nodes, $user \rightarrow node_1 \rightarrow node2 \rightarrow ..... \rightarrow node_{n-1} \rightarrow recommend \rightarrow user$. Sequences of nodes like these can be got by searching and traversing on a simplified undirect graph, which was built by ignoring concepts of nodes ,relations and directions of edges. Then for each two adjacent nodes in one sequence, we can get directly a entitative relation set of all $Relation_{\pm}(node_1,\ node_2)$ from the primary social semantic graph. Finally, a cartesian product of these relation sets of adjacent nodes can be treated as a set of all entitative formulas of the node sequence for the query.

The $Accept(user, recommend)$ as a query is always on the right side of $' \Rightarrow'$, and the evidences on the left side belong to the social semantic graph's edges set E which are all true. According to a first-order logic, the truth value of one formula is as same as the query's.

For the queries of one same User u and different recommended user in U_i, we make the following merger: (1)Change finding loops to finding paths; (2)All recommended users in User u's recommended user subset U_i are put into end node set EN; (3)Find paths staring with User u and ending with any recommended user in U_i, then make sure the lengths of these paths is limited between the maximum and the minimum. (4)Remove users never appearing as end nodes from U_i, because they are useless for model training.

We have to put forward a few rules, which can be view as pruning strategies: 1) prohibit backtrack to avoid getting palindromic node sequences, but the loop whose is 2 and the two edges are not identical are kept; 2) a loop containing a query edge as evidence is pruned, because queries have no exact truth values.

The time complexity of the algorithm for all users in U is $O(|U|H^L)$. Where H is the average size of each node's adjacent node set in N, and L is the maximum length we set for loops. Even when H and L are not very large, the H^L can be a enormous value and the time complexity is unbearable. Therefore, we have to give up complete search on the the whole social semantic graph.

4.2 Random Sampling

Retrospecting the equation (4), we want to estimate $P(Y_k = y_k|X = x)$ for Query y_k and it is a ratio. It relates to the ratio of $V_j(x, y[Y_k = y_k])$ and the sum of $V_j(x, y[Y_k = 1, 0])$, which increasing results in $P(Y_k = y_k|X = x)$ increasing. So we do not care about exact values of $V_j(x, y[Y_k = y_k, 1, 0])$ but the ratio.

The ratio of positive queries and negative queries is determinate when data set is determinate either for training or testing, or even alternative recommendation set in real world. Therefore, we can get all $P(Y_k = y_k|X = x)$ approximatively by ensuring fair treatments of positive and negative examples, and fair treatments of all concept formulas. The simplest way is to random walk on the Graph G and to allocate the same probability to adjacent nodes of one node in the transition probability matrix, but less formulas can be found or many conceptual formulas' counts is 0 when the percentage of sampling is very low. If we raise the percentage, the sampling mechanism will become insignificant. We need a heuristic strategy to find as many entitative formulas as possible when finding loops and keep all $P(Y_k = y_k|X = x)$ changeless at the same time. While the greedy strategy is a good choice. It means the algorithm will choose next nodes close to the target items. When we random walk and arrive at $node_u$ and want to transfer to its adjacent nodes, traverse each node noted as $node_v$ in $adj(node_u)$ and turn up the transition probability from $node_u$ to $node_v$ when there is a target user in $adj(node_v)$. Where $adj(node_x)$ is the set of nodes adjacent to $node_x$. The specific allocation method for the transition probability matrix P is as follows:

$$P(u,v) = \begin{cases} \dfrac{d(u,v)}{\sum_{x \in adj(u)} d(u,x)} & v \in adj(u), \\ 0 & v \notin adj(u). \end{cases} \tag{5}$$

$$d(u,v) = \begin{cases} 1 & v \in adj(u) \text{ and no target}, \\ \dfrac{c}{l_v} & v \in adj(u) \text{ and existing targets}, \\ 0 & v \notin adj(u). \end{cases} \tag{6}$$

Where $d(u,v)$ denotes the weight on the $Relation(u,v) \in E$, l_v is the number of nodes adjacent to $node_v$, and c is a constant which is larger and the algorithm tends to be more greedy. Such a transition matrix can ensure the proportionality of sampling while realizing the greedy strategy, which was proved in [1].

In this way, the time complexity of the randomly finding loops algorithm changes to $O(|U|M^L)$, where M is the maximum number of nodes to be visited from a visted node. And it does't require a large M, which is a tradeoff between computing time and the number of conceptual logic formulas found.

5 Experiments

5.1 Dataset and Evaluation Metrics

We choose an open data set, the Tencent Weibo Data Set(TWDS) from KDD-Cup'12 track 1[21], for our experiments. There are kinds of attributes, relations, even circumstances in the TWDS, and then we select a few representative ones, including *Follow* relations, *At*, *Retweet*, *Comment* actions with times, keywords with weights, gender, and birth year. The alternative recommendations sets of the TWDS are also also a bit special, all users in which are specific ones distinguished from ordinary friends in social networks, which can be celebrities, famous organizations, some well-known groups, or anything is public and famous[21]. Table 1 shows the statistics of these data sets. The recommended task is a classic ranking task, so apply the evaluation metrics of ranking to the recommended task is convictive. The Mean Average Precision(MAP) is a popular rank evaluation method to evaluate the proposed approach[31]. The KDD-Cup'12 track 1 use $AP@3$ as the final evaluation metric, and we expand it to $AP@n$ as our evaluation measures, where n is set to 1, 3, 5, 10.

Table 1. The Statistics of Datasets

Train Size	Test Size	Repetitive Rate	Train Accept Rate	Test Accept Rate
1392872	1196410	22.9%	13.0%	11.0%

5.2 Method Comparison

In this sub-section, we compare our method with several baselines. The detailed implementations are listed below:

– *RandomGuess*: It exchanges positions of the recommended users randomly as the final result. Concretely, we exchange 1000 recommended user pair randomly for a specific user after reading the test data, which ensures that the output is completely random. If there are results of other methods worse than this, these methods are useless.

– $ItemBased-CF$: Item-based collaborative filtering. It calculates similarity for each pair of two items and recommends items to one user, which are similar to items followed by the user. It is a representative neighborhood-based method and it is easier to realize because the number of items is much smaller than the users'. The similar between two items $Sim(i,j)$ is calculated by Equation (7), where $Follow_i$ means the follower set of Item i.

$$Sim(i,j) = \frac{Follower_i \cap Follower_j}{\sqrt{|Follower_i|} \cdot \sqrt{|Follower_j|}} \tag{7}$$

– $MatrixFactorization$: Matrix Factorization Model. It is an excellent approach for recommendation systems, which captures implicit relations between users and items. It construct factor vectors for each user and item by decomposing the rating matrix. The factor vectors got can be use to predict the missing rating, which is used to recommend. For a pair of a user and a item, the rating r_{ui} is calculated by Equation (8)[17].

$$r_{ui} = \mu + b_i + b_u + q_i^T p_u \tag{8}$$

Where p_u and q_i are respectively the factor vector of User u and Item i. We learn it by minimizing the squared error function (9), where r_t is the true value from the rating matrix.

$$L(u,i) = \sum_{(u,i)\in K} (r_{ui} - r_t)^2 + \lambda(\|p_u\|^2 + \|q_i\|^2 + b_u^2 + b_i^2) \tag{9}$$

5.3 Results

Table 2 shows the results of all methods, and we can obtain the following observations:

1) Our method performs best on $AP@1$, which indicate the formula-based method is inclined to predict the top result.

2) The performance of our method decreases with N (in $AP@N$) increases, which indicates our method is not good at predicting missing links without strong evidence.

3) Matrix Factorization outperforms ours on $AP@3$, and it is the state-of-the-art for TWDS dataset. The winner of the KDD Cup 2012 developed its method based on Matrix Factorization. However, it does not outperform on the top place of the recommendation, which is the most import for almost all link prediction tasks. Therefore, it is necessary to merge formula-based method and matrix factorization method to achieve higher quality social recommendation.

6 Conclusion and Future Work

This paper treats social recommendation as a link prediction task on the social graph, and proposes a formula-based method to construct probabilistic formulas to predict potential links. Our method employs MLN to merge the force of

Table 2. Results for different ML

$Methods$	AP@1	AP@3	AP@5	AP@10
$RandomGuess$	0.097	0.180	0.202	0.186
$ItemBased-CF$	0.207	0.326	0.311	0.284
$MatrixFactorization$	0.220	0.366	0.327	0.298
$OurMethod$	0.277	0.353	0.315	0.268

various logic formulas and we conduct an experiment on a public social recommendation dataset in KDD Cup 2012. Our method achieve a good performance and perform best on precision at the top place of recommendation list.

In the future, we will explore the different effect of formula-based methods and matrix factorization, and try to merge them. To achieve higher quality social recommendation, we will also try to employ distributional representation methods, which are proved effective on the knowledge base and may be also good at social recommendation.

Acknowledgments. This work was supported by the Natural Science Foundation of China (No. 61533018), the National Basic Research Program of China (No. 2014CB340503) and the National Natural Science Foundation of China (No. 61272332 and 61602479). And this work was also supported by Google through focused research awards program.

References

1. M. Al Hasan and M. J. Zaki. Musk: Uniform sampling of k maximal patterns. In *SDM*, pages 650–661, 2009.
2. L. Backstrom and J. Leskovec. Supervised random walks: predicting and recommending links in social networks. In *WSDM*, pages 635–644. ACM, 2011.
3. M. Balabanović and Y. Shoham. Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40(3):66–72, 1997.
4. C. Basu, H. Hirsh, W. Cohen, et al. Recommendation as classification: Using social and content-based information in recommendation. In *AAAI/IAAI*, pages 714–720, 1998.
5. Z. Burda, J. Duda, J. Luck, and B. Waclaw. Localization of the maximal entropy random walk. *Physical review letters*, 102(16):160602, 2009.
6. K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, and Y. Yu. Collaborative personalized tweet recommendation. In *SIGIR*, pages 661–670. ACM, 2012.
7. T. Chen, L. Tang, Q. Liu, D. Yang, S. Xie, X. Cao, C. Wu, E. Yao, Z. Liu, Z. Jiang, et al. Combining factorization model and additive forest for collaborative followee recommendation. In *Proceedings of the KDD-Cup'12 Workshop*, 2008.
8. N. Z. Gong, A. Talwalkar, L. Mackey, L. Huang, E. C. R. Shin, E. Stefanov, D. Song, et al. Jointly predicting links and inferring attributes using a social-attribute network (san). In *ACM Workshop on Social Network Mining and Analysis (SNA-KDD)*, 2012.
9. J. Hannon, M. Bennett, and B. Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 199–206. ACM, 2010.
10. T. N. Huynh and R. J. Mooney. Discriminative structure and parameter learning for markov logic networks. In *ICML*, pages 416–423. ACM, 2008.

11. M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 135–142. ACM, 2010.
12. A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 79–86. ACM, 2010.
13. H. Kashima, T. Kato, Y. Yamanishi, M. Sugiyama, and K. Tsuda. Link propagation: A fast semi-supervised learning algorithm for link prediction. In *SDM*, volume 9, pages 1099–1110. SIAM, 2009.
14. S. Kok and P. Domingos. Learning markov logic network structure via hypergraph lifting. In *ICML*, pages 505–512. ACM, 2009.
15. S. Kok and P. Domingos. Learning markov logic networks using structural motifs. In *ICML*, pages 551–558, 2010.
16. Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *SIGKDD*, pages 426–434. ACM, 2008.
17. Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
18. G. Li and M. J. Zaki. Sampling minimal frequent boolean (dnf) patterns. In *SIGKDD*, pages 87–95. ACM, 2012.
19. D. Lowd and P. Domingos. Efficient weight learning for markov logic networks. In *PKDD*, pages 200–211. Springer, 2007.
20. H. Ma. An experimental study on implicit social recommendation. In *SIGIR*, pages 73–82. ACM, 2013.
21. Y. Niu, Y. Wang, G. Sun, A. Yue, B. Dalessandro, C. Perlich, and B. Hamner. The tencent dataset and kdd-cup'12. In *KDD-Cup Workshop*, volume 2012, 2012.
22. S. Rendle. Factorization machines. In *ICDM*, pages 995–1000. IEEE, 2010.
23. S. Rendle, Z. Gantner, C. Freudenthaler, and L. Schmidt-Thieme. Fast context-aware recommendations with factorization machines. In *SIGIR*, pages 635–644. ACM, 2011.
24. M. Richardson and P. Domingos. Markov logic networks. *Machine learning*, 62(1-2):107–136, 2006.
25. B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *WWW*, pages 285–295. ACM, 2001.
26. Y. Shi, M. Larson, and A. Hanjalic. Exploiting user similarity based on rated-item pools for improved user-based collaborative filtering. In *Proceedings of the third ACM conference on Recommender systems*, pages 125–132. ACM, 2009.
27. S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha. Like like alike: joint friendship and interest propagation in social networks. In *WWW*, pages 537–546. ACM, 2011.
28. Z. Yin, M. Gupta, T. Weninger, and J. Han. Linkrec: a unified framework for link recommendation with user attributes and graph structure. In *WWW*, pages 1211–1212. ACM, 2010.
29. Z. Yin, M. Gupta, T. Weninger, and J. Han. A unified framework for link recommendation using random walks. In *Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on*, pages 152–159. IEEE, 2010.
30. J. Zhang, C. Wang, P. S. Yu, and J. Wang. Learning latent friendship propagation networks with interest awareness for link prediction. In *SIGIR*, pages 63–72. ACM, 2013.
31. M. Zhu. Recall, precision and average precision. *Department of Statistics and Actuarial Science, University of Waterloo, Waterloo*, 2, 2004.

GRU-RNN based Question Answering over Knowledge Base

Shini Chen, Jianfeng Wen, and Richong Zhang

State Key Laboratory of Software Development Environment
School of Computer Science & Engineering
Beihang University

Abstract. Building system that could answer questions in natural language is one of the most important natural language processing applications. Recently, the raise of large-scale open-domain knowledge base provides a new possible approach. Some existing systems conduct question-answering relaying on handcraft features and rules, other work try to extract features by popular neural networks. In this paper, we adopt recurrent neural network to understand questions and find out the corresponding answer entities from knowledge bases based on word embedding and knowledge bases embedding. Question-answer pairs are used to train our multi-step system. We evaluate our system on FREEBASE and WEBQUESTIONS. The experimental results show that our system achieves comparable performance compared with baseline method with a more straightforward structure.

1 Introduction

Recently, some structured knowledge bases have been published, such as Freebase [6], DBpedia [1], YAGO [17]. The vertices and edges in these graphs with different labels represent entities and relations in real world. The availability of knowledge bases makes it possible to discover relational knowledge from clean and structured data storage. Especially when we using human language to query the knowledge base, mapping the question text with the stored knowledge is a great challenge.

To map the search desire to the triples in the knowledge base, most of the existing studies [11] [7] focus on understanding the question and finding the matching entities and relations in the knowledge base. One of the characteristic features of the knowledge base or knowledge graph is that there exists a fix number of relations. However, the user provided questions may vary significantly. The key issue for successfully locating the correct answers for a question is to discover the hidden links between the questions' syntactical structures and the relations. In practice, the questions' syntactical structures usually follow some specific patterns. In this study, we advocate that the latent semantic matching between the question and the knowledge triple provides an opportunity to model

the hidden relation between question patterns and the relations in the knowledge base. Specifically, we propose three steps for solving the question answering problem over knowledge base. For the relation identification from the question text and question mapping on the knowledge base, further than existing deep learning method which simply put corpus into deep neural networks, we design a two-column GRU-based RNN for characterizing the latent semantics between question text and the knowledge triples. Empirical studies on the commonly-used WEBQUESTIONS for question answering task evaluation confirms the effectiveness of our proposed model for both relation identification and question and answer mapping.

The remainder of this paper is organized as follows. Section 2 introduces the related studies for solving the problem of question answering over the knowledge base. Section 3 delivers the procedures and the two-column GRU-based RNN model. Section 4 describes the training and inference of the proposed model. Section 5 presents experimental evaluation of our framework. Finally, Section 5 will discuss and conclude the paper.

2 Related Work

The state-of-art method in knowledge based QA can be divided into two mainstreams, namely, semantic parsing based and information retrieval based.

Semantic parsing based method focus on learning semantic parsers which parse natural language question into logical form and query knowledge base to lookup answers. In [4], authors propose an approach that generates query candidates by recursively generating logical form with a mapping of phrases to knowledge base predicates and a small set of composition rules, and rank query candidates by log-linear model. In [5], a set of candidate logical form is generated and then a paraphrase model is introduced to choose the realization that best paraphrases the input question, and the corresponding logical form is produced. While early approaches heavily relied on manually annotated logical form and high-quality lexicons to train semantic parser, recent work has focused on training semantic parser only using question answer pair. In [3], the proposed approach translates a given natural language question to the matched SPARQL query and use learning-to-rank techniques to learn pair-wise comparison of query candidate. [19] formulates semantic parsing as a staged search problem, mapping natural language question into a query graph which resembles subgraph of knowledge graph.

Information retrieval based method first retrieves a large set of candidate answers from knowledge base, and then rank them by fine-grained extracted features from question and answer. In [18], authors propose a model for di-

rectly learning the pattern of question answer pair. Firstly, question dependency parse is converted to candidates topic graphs by rules, then the relations and properties in topic graph are fed into a logistic regression model as features to classify correctness of questions candidate answer. Recently, many studies embed questions and knowledge graph entries in a low-dimensional vector spaces and retrieve the answers by computing similarities in learned embedding space. For example, [7] combines the embeddings of words in question as its representation, and encodes answer by summing embeddings of entities and relations that appear in question-answer path and surrounding subgraph. Then, the score of a question-answer pair is given by dot production of question embedding and answer embedding. In addition, [11] uses multi-column convolutional neural network to generate three question aspects, and ranks candidate by considering answer type, answer path and answer context. In [8], authors conduct QA process under the embedding based Memory Networks framework. In [12], the sequence translation framework are exploited to feed question characters into encoding LSTM and to obtain the knowledge base triples from an attention-based decoding LSTM.

In general, these existing studies focus on transformation from the question words to a knowledge base query. However, the sequential patterns of question word is ignored for building the QA model. In practice, this sequential information is important for the question understanding. In this study, we exploit the GRU-RNN model, which can characterize the sequential patterns of input question text, identify the question pattern, and match the question and the knowledge base triples.

3 Model

3.1 Problem Definition

In general, three aspects are considered for the question answering problem over a knowledge graph. The first aspect is to identify the entities that have appeared in the question. It focuses on identifying the words in the question that may be translated to the entities in the knowledge graph; the second aspect is to discover the relation mentioned in the question sentence; and the third aspect is to understand the semantics of the entities and relations and match them with the existing entities and relations or paths in the knowledge graph and then to rank the matched triples or paths. In this study, we propose three main modules for solving the above mentioned aspects.

3.2 Entity Matching Module

The question of how exactly the topic entity is identified has been discussed by many research, e.g. [14] [15]. In this study, we exploit the solutions provided by [3] to identify the entity in the question. The match between the question words and the knowledge entities could be literal or via an alias of the entity name.

We first POS-tag a question by the Stanford tagger [16], and apply some simple rules to filter subsequences(n-grams) of question to get candidate entity word set. The rules are: (1) a subsequence containing only single word must tagged NN(noun) (2) consecutive words tagged NNP(proper noun) cannot be split into two subsequences. The filtered subsequence set $\mathcal{S}$ is used to retrieve a list of entities from knowledge base, whose name or alias is literally similar to a candidate in $\mathcal{S}$. We use dictionary provided by [3], which contains mappings from name or alias to Freebase entities with matching scores. We set a threshold to limit the number of candidate entities.

3.3 Relation Identification Module

The most important step for retrieving the answer from a knowledge base is to identity the question semantics and to locate the corresponding knowledge graph relations. In practice, the questions' syntactical structure usually follow some patterns. If we remove the entity word from the question sentence, the remaining sequence of words can somehow represent the `question pattern`, or the semantic pattern of a question sentence. The relation in `FREEBASE` is organized as the format of $relation\_field.excepted\_subject\_type.relation\_name$, which can be considered as a sequence of sub-relation labels. For instance, *people.deceased_person.cause_of_death* can be considered as a sequences of people, deceased_person, cause_of_death.

The relation identification problem now is translated into modeling the semantic similarity between the `question pattern` and the knowledge base relation. For example, on one hand, question *what did george orwell died of* and question *what was jesse james killed with* should be mapped into the same knowledge base relation *people.deceased_person.cause_of_death*. On the other hand, the same knowledge base relation *people.deceased_person.cause_of_deat* may correspond to question patterns *what did _ died of* and *what was _ killed with*.

To discover the semantic relation between `question pattern` and the knowledge base relation (pattern-relation pair), we build a two-column GRU-based RNN, which is displayed in **Figure1**. We will introduce this model in the following subsections.

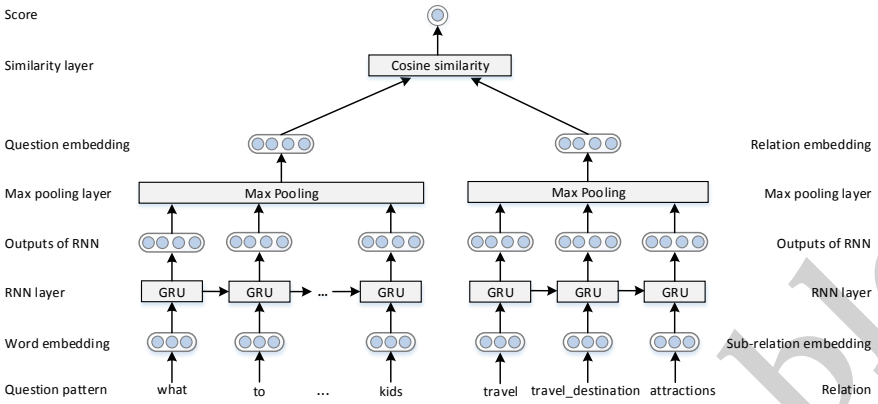


Fig. 1. The Two-columns GRU-based RNN Model

3.4 Question-Triple Matching Module

As the final goal of the QA task is to find the answers from the knowledge base, the relation between the question and the triple in the knowledge graph is to be determined. To model this relation, we translate the triple into a sequence of words by concatenating the subject, the relation and the object of a triple together. For example, the triple m.02khkd *people.deceased_person.cause_of_death* m.012hw is translated as *jesse james people deceased person cause of death assassination*. Here, *jesse james* and *assassination* are standard name of entity m.02khkd and m.012hw respectly. We use the same two-column GRU-based RNN, as shown in **Figure1** to characterize the question-triple relations.

3.5 GRU-based RNN for QA

Figure 1 shows the learning architecture for the relation identification and answer matching system. This architecture consists of two-column independent recurrent neural network(RNN) with Gated Recurrent Unit(GRU) cell [10]. This two-column GRU-RNN is used to character the similarity between two input sequences. Each GRU RNN layer takes a sequence of vectors as input, and produces a output vector for each input vector. In our system, the input are sequence of words or sub-relation labels, so we apply a lookup layer to transform them into vectors. For better understanding the latent semantics underlining the word sequences, we use the embedding as input to the GRU cell.

The lookup layer transforms every input word w_i of q or sub-relation label rl_i of the sub-relation label sequence into a input embedding vector $\mathbf{x_i} = \mathbf{W}\phi_i$, where ϕ_i is the one-hot vector representing w_i or rl_i, $\mathbf{W} \in \mathbb{R}^{k \times |Elements|}$ is the matrix of word embedding, k is the input vector dimension, $|Elements|$ is the

total number of words and sub-relation labels. Output vector h_i for input vector x_i is calculated by:

$$\mathbf{z}_i = \sigma(\mathbf{W_z} \cdot [\mathbf{h}_{i-1}, \mathbf{x}_i]) \tag{1}$$

$$\mathbf{r}_i = \sigma(\mathbf{W_r} \cdot [\mathbf{h}_{i-1}, \mathbf{x}_i]) \tag{2}$$

$$\tilde{\mathbf{h}}_i = tanh(\mathbf{W_h} \cdot [\mathbf{r}_i * \mathbf{h}_{i-1}, \mathbf{x}_i]) \tag{3}$$

$$\mathbf{h}_i = (1 - \mathbf{z}_i) * \mathbf{h}_{i-1} + \mathbf{z}_i * \tilde{\mathbf{h}}_i \tag{4}$$

where $[.]$ is the concat operator; $\sigma(\cdot)$ is the sigmoid function; $\cdot$ means the matrix production and $*$ is a element-wise production. Specially, we assign the zero-vector for h_0.

Next, the following max-pooling layer will output a fix-length vector $\mathbf{v}$, where

$$\mathbf{v} = \max_{i=1,...,n} (h_i) \tag{5}$$

the max$(\cdot)$ is the element-wise operator over $\{h_i\}$ and n is the length of input sequence. The top layer of architecture evaluates similarity between two final output vector by the two column network. Here, we use cosine similarity as metric.

During the Relation Identification process, the input of left column network is word sequence of question pattern and the input of right column network is sub-relation sequence of certain relation. When conducting question triple matching, we feed complete question word sequence and word sequence of triple into each column network respectively. Note, except for word embedding matrix used in question triple matching is shared between two column networks, other parameters in architecture are independent.

4 Train and Inference

4.1 Training

We adopt margin-based ranking loss function to estimate parameters.

Relation Identification: The training data is denoted by $\mathcal{D} = \{(p_i, r_i) : i = 1, ..., |D|\}$, where p_i is the `question pattern` of the i^{th} training question, and the r_i is the corresponding sub-relation label sequences in the knowledge base. The function $S(p, r)$ represents the cosine similarity between the embedded vector of question pattern p and the embedded vector of the relation r, which are the output vectors of the two GRU-based RNN modules (the left and

the right columns of our proposed model shown in Figure 1). This objective function is formulated as:

$$\sum_{i=1}^{|\mathcal{D}|} \sum_{\tilde{r} \in \tilde{R}(r_i)} max\{0, m - S(p_i, r_i) + S(p_i, \tilde{r})\} \tag{6}$$

where the $\tilde{r}$ is a negative relation for question, which is different with r_i. We will introduce the details of choosing negative examples in the Experiments section. We exploit Adam[13] algorithm to minimize the objective function and to learn the GRU parameters and input embedding vectors.

Question-Triple Matching: We use the same two-column RNN to train the Question-Triple matching model. For this task, the training data is $\mathcal{D} = \{(q_i, t_i) : i = 1, ..., |D|\}$, where q_i is the i^{th} training question, and the t_i is one of its correct answer triples in the knowledge base. The word sequences q_i and t_i are the inputs of the left and right columns of our proposed model.

4.2 Inference

Once our model is trained, we can use this model to answer new questions. Given a new question q, by using the entity linking technique proposed in [3], we select the entities whose score is higher than a pre-given threshold as the topic entity set $\mathcal{S}$. Then, we find subgraph of entity $s \in \mathcal{S}$, and extract all one-hop paths or two-hop paths passing CVT node, the relations in the paths are chosen as possible candidate relations. Here, we ignore the first relation in two-hop path. Next, we make use of our learned **Relation Identification** module to discover the top-k (p, r) pair, where k is hyper-parameter and we set it to be 3 in our work. Finally, we find all triples in knowledge base that satisfy the form of $(s, r, ?)$ as candidate triples. We denote this candidate triples as C_q and adopt our **Question-Triple Matching** module to rank them. Because there exists some multi-answer questions, we generate predicated triples set $\widehat{C}_q$ as:

$$\widehat{C}_q = \{\widehat{t} | \widehat{t} \in C_q \; and \; S(q, \widehat{t}) > S(q, t^\star) - m\} \tag{7}$$

where $S(q, t^\star)$ is the highest score and we use the same threshold m as in **Equation** 6.

5 Experiments

We conduct experiments on the `WEBQUESTIONS` testing set to evaluate our system.

DATASET: `WEBQUESTIONS` [4] is a popular dataset to evaluate efficiency

of QA system, which consists of 5810 question-answer pairs. Because WEBQUESTIONS provides only question-answer pair, we simulate question answering process to collect relation information for training Relation Identification model. Firstly, we use Entity Matching Module described in **Section 3.2** to get candidate topic entities for questions. Then the 1-hop or 2-hop passing CVT paths on the FREEBASE that connect a candidate topic entity to at least one answer entity are identified as candidate relations. Finally, the relations connecting the most answer entities are voted as correct relations. Other relations founded in QA process are regarded as negative relations.

FREEBASE: As the WEBQUESTIONS dataset uses entities in FREEBASE, we adopt this knowledge base to develop our model. FREEBASE is a large collaborative knowledge base consisting of data composed mainly by its community members. To make FREEBASE fit in memory, we apply the similar preprocess method presented in [7] to extract a subset of FREEBASE.

5.1 Setting

In our experiments, all hyper-parameters are chosen on the WEBQUESTIONS validation set. The size of word vectors d_w, sub-relation vectors d_r and hidden state of GRUs d_g are selected among $\{64, 128, 192, 256\}$. We used mini-batch Adam algorithm [13], where batch size is 40, initial learning rate α is selected among $\{0.1, 0.01, 0.001, 0.0001\}$. Initial weighs of GRUs are drawn from a 0-mean truncated normal distribution with 0.1 standard deviations. Embedding of word and sub-relation are initialized in same way. The bias inside GRUs are started as 1.0 to make cell not reset and not update. The margin m in Equation6 is set to 0.1. Optimal configurations are: $d_w = 192$, $d_r = 192$, $d_g = 192$, $\alpha = 0.001$.

5.2 Experimental Result

We compare our system in terms of average F1 score as computed by the official evaluation script provided by (Berant et al., 2013). For each testing question, we compare the predicted answer set to gold answer set, and compute its F1 score. After going through the whole testing set, we get the popular macro F1 metric that is the average value of the F1 score of all testing samples. As shown in **Table 1**, our system achieves comparable or better result than baseline system on WEBQUESTIONS.

We also conduct experiments to examine the effect of the core Relation Identification module. Given a question, We applied Relation Identification module to rank its candidate topic entity and relation pair (s, p), As shown in **Table 2**,

the correct (s, p) of 60% questions are ranked at first place. Note that, when only using Relation Identification Module to achieve QA task, all retrieved entities have the same score. Therefore, the listed p@k results of Relation Identification are evaluated on correct relations. After further analyzing, we discover that 170 questions have no corresponding paths between topic entity and answer entities in FREEBASE. Ignoring these 170 questions, the Relation Identification model achieves P@1=66.5%, with the average ranking of the correct (s, p) being 4.28, and the average number of candidate (s, p) pairs is 70.7. For a given question, if we take directly the entities connecting to s by p as the predicted answer set, where the (s, q) pair is the first ranked result of the Relation Identification module, we get F1 = 40.9%, which is an acceptable result. This proves that the Relation Identification module achieve a good efficiency.

Table 1. Evaluation result on the testing set of WEBQUESTIONS, compared to baselines. The results of baselines are from their original papers.

Method	F1
Berant et al., 2013 [4]	31.4%
Berant and Liang, 2014 [5]	39.9%
Bao et al., 2014 [2]	37.5%
Yao and Van Durme, 2014 [18]	33.0%
Bordes et al., 2014a [7]	39.2%
Bordes et al., 2014b [9]	29.7%
Dong et al., 2015 [11]	40.8%
Our Method	**42.0%**

Table 2. Evaluation results of different settings. The listed p@k results of Relation Identification are evaluated on correct relations.

	All	Relation Identification
F1	42.0%	40.9%
p@1	43%	61%
p@3	52%	69%
p@5	57%	73.2%

5.3 Relation Word Detection

In this section, we show how the trained GRU-RNN extracts relation key word from input question pattern. As we know, the GRU layer generates vector h_t for each input word, the following max-pooling layer takes the maximum value of each vector dimension to form final question pattern representation p. Intuitively, for each dimension of vector, the word with the maximum value contributes the most. We feed some question patterns into GRU-RNN and inspect output vector for each word. A interesting phenomenon is observed that different dimensions of vector are sensitive to different relation words. For instance, the 57th dimension will turn on when meeting words indicating relation people.person.children. And the 43rd dimension will be activated by words in-

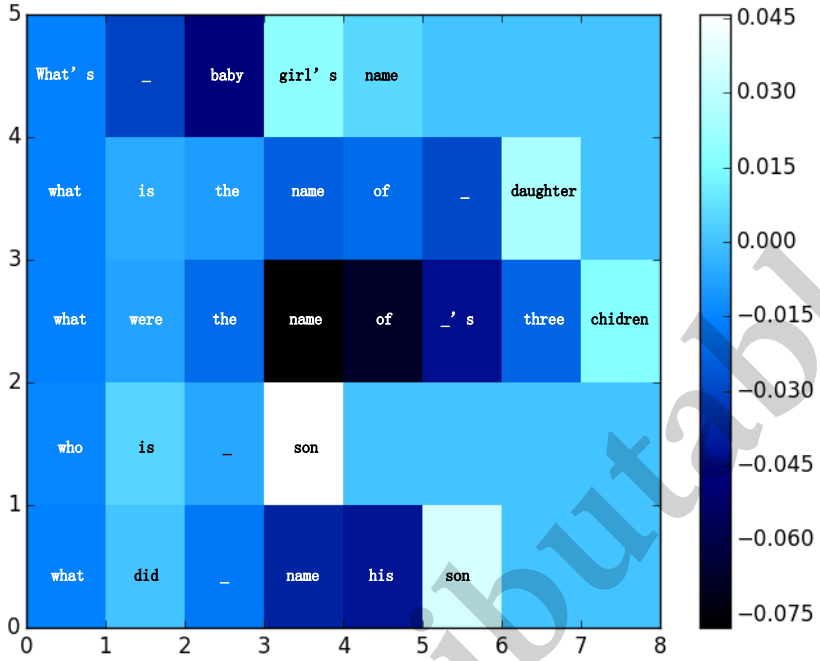


Fig. 2. The 57th dimension of word vectors output by GRU-RNN layer when given 5 question patterns expressing semantic of people.person.children. The vertical axis is the question pattern index from 1 to 5, and the horizontal axis is the word index from 1 to 8 numbered from left to right in a sequence of words, and color codes show activation values. Relation key words like girls, daughter, children, son have relatively high value in each question pattern.

dicating relation *people.marriage.spouse*. **Figure2** shows the 57th dimension of word vectors output by GRU-RNN when given 5 following question patterns:

* What's _ baby girl's name ?
* What is the name of _ daughter ?
* What were the name of _s three children ?
* Who is _ son ?
* What did _ name his son ?

which all express the same semantic of *people.person.children*. Obviously, words with the maximum value in each question pattern respectively are girls, daughter, children, son, son, which are exactly relation key words.

5.4 Error Analysis

We randomly select some questions from the wrongly answered questions to find out possible causes.

Entity linking: In entity linking stage, some entity mentions failed to be linked due to POS error. Meanwhile some other entity mentions are correctly located but its corresponding topic entity are dropped due to the low matching score.
Relation Predication: For a part of questions, there do not exist any 1-hop or 2-hop passing CVT node path from its topic entity to answers. As a result, our method can not answer this type of questions for now. What's more, some questions are roughly answered because there is no single relation exactly expressing the semantic of question. For example, we answer *who is Keyshia cole dad* with Keyshia cole's dad and mom based on relation $people.person.parent$, because there is no relation like $people.person.dad$. Overall, most of errors come from incorrect rank of relations.
Constraints and Aggregations: Some questions contain constraint words. For instance, the question *who did jackie robinson first play for*, asking the role that Jackie Robinson played as his first time. Only identifying the relation *sports.sports_team_roster.team* is not sufficient to correctly answer it. Such that, we need further aggregation operation or develop more advanced mechanisms.
Label error: Some errors in fact are caused by label issues and are not real mistakes. For instance, standard answer set to What are the songs that Justin Bieber wrote only contains 10 songs, which is not completely labeled. And sometimes, the answers that we provide is accepted. For example, we answer *Where did francisco coronado come from* with the entity $Salamance$ which is a city northwestern Spain, while the gold answer is $Spain$. What's more, the standard answers of some questions are wrong.

6 Conclusion

In this paper, we propose our knowledge graph based question answering system. We divide the question answering problem into three different parts, and provide three corresponding sub-systems. The GRU-based RNN is the core tool to `Relation Identification` and `Candidates Ranking`, because we take the natural language and KB triples as sequences data. Our system achieve comparable result than other baselines, including both semantic parsing and features extraction methods, with a intuitive and simple system structure rather than the complex human handcraft feature or delicate neural network they used.

References

1. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data. In: The semantic web, pp. 722–735. Springer (2007)

2. Bao, J., Duan, N., Zhou, M., Zhao, T.: Knowledge-based question answering as machine translation. Cell 2(6) (2014)
3. Bast, H., Haussmann, E.: More accurate question answering on freebase. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. pp. 1431–1440. ACM (2015)
4. Berant, J., Chou, A., Frostig, R., Liang, P.: Semantic parsing on freebase from question-answer pairs. In: EMNLP. p. 6 (2013)
5. Berant, J., Liang, P.: Semantic parsing via paraphrasing. In: ACL (1). pp. 1415–1425 (2014)
6. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: Proceedings of the 2008 ACM SIGMOD international conference on Management of data. pp. 1247–1250. ACM (2008)
7. Bordes, A., Chopra, S., Weston, J.: Question answering with subgraph embeddings. arXiv preprint arXiv:1406.3676 (2014)
8. Bordes, A., Usunier, N., Chopra, S., Weston, J.: Large-scale simple question answering with memory networks. arXiv preprint arXiv:1506.02075 (2015)
9. Bordes, A., Weston, J., Usunier, N.: Open question answering with weakly supervised embedding models. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 165–180. Springer (2014)
10. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
11. Dong, L., Wei, F., Zhou, M., Xu, K.: Question answering over freebase with multi-column convolutional neural networks. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. vol. 1, pp. 260–269 (2015)
12. Golub, D., He, X.: Character-level question answering with attention. arXiv preprint arXiv:1604.00727 (2016)
13. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
14. Liang, P., Jordan, M.I., Klein, D.: Learning dependency-based compositional semantics. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. pp. 590–599. Association for Computational Linguistics (2011)
15. Ling, X., Singh, S., Weld, D.S.: Design challenges for entity linking. Transactions of the Association for Computational Linguistics 3, 315–328 (2015)
16. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J.R., Bethard, S., McClosky, D.: The stanford corenlp natural language processing toolkit. In: ACL (System Demonstrations). pp. 55–60 (2014)
17. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: a core of semantic knowledge. In: Proceedings of the 16th international conference on World Wide Web. pp. 697–706. ACM (2007)
18. Yao, X., Van Durme, B.: Information extraction over structured data: Question answering with freebase. In: ACL (1). pp. 956–966. Citeseer (2014)
19. Yih, W.t., Chang, M.W., He, X., Gao, J.: Semantic parsing via staged query graph generation: Question answering with knowledge base. In: Association for Computational Linguistics (ACL) (2015)

Research on judging character relation triples based on sentence pattern

Zhao Jiapeng, Yan Yang, Liu Tingwen*, Shi Jinqiao

Institute of Information Engineering, Chinese Academy of Science, Beijing, China

liutingwen@iie.ac.cn

Abstract. Extracting character relation triple (S, P, O) from large number of unstructured text is crucial to the construction of knowledge graph, knowledge representation and reasoning of character relations. Aiming at low accuracy in extracting triples from unstructured text, we put forward a supervised approach to judge whether extracted triples are correct. The approach need to build a knowledge base which contain character's attributes first, and learn a sentence pattern tree according the character attribute knowledge base and the training data. When training, extracting triples from the text and manual labeling whether the triple is correct. Then constructing patterns according the position of "triple", "pronoun" and "word" in the sentence by level. At the same time, the correct and error number of are recorded on each pattern. When testing, judging triple by the number recorded in matching pattern. According the test result, our approach does better in the training time, the testing time and the F1-value (76.6%) than the ordinary approach based on feature engineering (75.7%). At last, we make sentence pattern tree as a feature to improve the feature engineering approach (77.5%). In addition, this approach has a better expansibility than traditional approach, and has guiding significance to the construction of the training set.

Keywords: knowledge graph; personal relation extraction; pattern match; feature extraction

1 Introduction

The foundation work to study people's network behavior is the characters knowledge graph construction. It is crucial to the analysis of related text on the web. The triple(S, P, O) is an important part of the knowledge graph construction. However not only the number of triples extracted by information extraction is huge, but also the precision of extracted triples is difficult to satisfactory. To solve this problem, this paper presents an approach to judge whether extracted triples are correct.

Entity relation is the semantic relation between entities. Automatic Content Extraction (ACE) conference defines the relation extraction as: according the

adfa, p. 1, 2011.

predefine relation type, judging whether the specific semantic relations exist or the given relation type is correct. Relation Extraction is one of the most important approaches to get character relation triples. Currently, the mainstream approach in entity relation extraction mainly contains: Pattern Match, Semantic Analysis, Feature Classification and so on.

The Pattern Match approach [1,2,3] first formulates the corresponding patterns and relation types according the observation and analysis of instances in training set. Then, match instances in the testing set with patterns preformulated. If any match, we can judge the relation type by the pattern.

The main problem of Pattern Match approach is most of patterns are formulated artificially, which make it consume a large amount of human resources. In big data era, the huge scale data makes it impossible to formulate comprehensive and accurate patterns. In addition, when the specific area changes, the original pattern may be won't work well any more. Usually we need to re-formulate new patterns to make it adapt to the new area. For example, paper 4 formulated rules like the relation indicating words must contain verbs to realize the character relation judging; Paper 5 aiming at the problem of the irrelevant items extraction and missing key information existing in previous work. Through the statistical analysis of the error data, they put forward the approach of making use of part of speech tagging to develop syntactic and lexical constraint patterns to solve the problem; Paper 6 use a semi-supervised approach to extraction. The approach requires manual participation 10-15 minutes every day. However the target of manual intervention is blind. Not targeted! This paper proposes a judging approach based on Pattern Match, it enumerates various possible situations according the distribution of the training data.

The Semantic Analysis approach deduced some formalized representation which could reflect the meaning of sentence, according to the syntactic structure and the meaning of each notional word in the sentence [7]. By the formalized representation, characters relation could be judged. Using the dependency relation extraction approach, only the part of speech of words is considered, such as paper 8 uses part of speech to formulate patterns. These approaches have no consideration of the semantic gap and usage gap between verbs. For example, both sentence A and sentence B can match some part of speech pattern, but the specific words may lead different meaning; Paper 9 constructs a feature set by the Semantic Role Labeling. Then, a statistical feature combination approach is proposed, and the SVM (Support Vector Machine) classifier is used to realize the semantic analysis; Paper 10 proposes a semantic analysis of noun verbs semantic role labeling based on the traditional verb semantic role labeling. The approach could be used to realize the information extraction; Paper 11 mainly uses the Semantic Role Labeling in the Open Information Extraction. The pattern match approach proposed in this paper doesn't analysis the dependence relation of the sentence, this makes the approach avoid those problem exist in semantic analy-

sis.

The feature engineering approach judges whether the given character relation is correct by N-Gram features, word-frequency features[12], TF-IDF features[13], sometimes may also contains some pattern features, semantic analysis features[14,15] in sentences. Classifier such as SVM [16], maximum entropy, decision tree is taken to transform the judging problem to a binary classification problem. Some approaches [17] also utilize the external resources to improve the accuracy of the relation judging. The problem of the feature engineering approach is: Firstly, the feature space for representing text is in very high dimension. It results in low efficiency of training and testing. Secondly, when the classifying quality is not so good, it's hard to discover the concrete instance which is wrong, the only thing we can do is to adjust parameters of the classifier or select new features. Thirdly, when the difference of feature distribution in the training set and testing set is great, the classifying quality is bad. It's hard to build a comparatively complete training data set.

According the shortcomings of previous work, we put forward a supervised approach to judge whether the triple is right. The approach need to build a knowledge base which contain people attributes first, then learned a sentence pattern tree according the knowledge base and the training data. When training, fetch triples from the text, manual label whether the triple is correct. Then construct patterns according the position of "triple", "pronoun" and "word" in the sentence by level. At the same time, record correct and error number of triples match to patterns. When testing, according the correct and error number of patterns which the sentence matches to judge triples. There is no need for our approach to the complex analysis of semantic analysis such as dependency relation and syntax. It could lean a set of patterns automatically according the given training set. When the field changes, it could self-study only by given the training set of the corresponding field. Thanks to the tree structure of our patterns, the efficiency of training and testing are relatively high. When the judging result is wrong, our approach could find the error instances timely. It's convenient to analyze the causes of errors. Aiming at the shortcomings of our patterns in not considering character attributes the distance between characters and relation indicator words and the distinguishing ability of relation indicator words, we extract features of character and improve our pattern approach.

2 Judging approach based on Sentence Pattern

In this paper, we predefine 19 kinds of Chinese relations (同为校花(campus beauty), 昔日情敌(rivals in love), 老师(teachers and students), 撞衫 (clothing clashing), 前女友(ex-girlfriend), 偶像(idol), 暧昧(ambiguous), 绯闻女友 (gossip girl), 传闻不和 (Hearsay discord), 前妻 (ex-wife), 闺蜜 (confidante), 同学

(classmate), 妻子 (wife), 分手 (separate), 翻版 (carbon copy), 朋友(friends), 经纪人 (agent), 老乡 (fellow-villager), 同居(cohabitation)), these relations belong to entertainment domain. The reason we choose this domain is that the domain has a rich type of relations.

2.1 Selection of relation indicating words

For each kind of relation, we need to find relation indicating words to distinguish them. The number of relation indicating words need to as small as possible and they can represent the 19 kinds of relations effectively.

For data of a given type, the training data is represented as $P= \{p_1, p_2,\ldots,p_n\}$, p_i is text i in the corpus. After segmenting each p_i in P, we can get a dictionary $W = \{w_1, w_2, \ldots , w_m\}$, w_i is ith word in the dictionary. Then the selection of relation indicating word could translate into finding the subset S ($S \subseteq W$) in the dictionary. S should cover P (For each word in pi, at least one appear in S); S is the minimal set which meet the above conditions, represented as $|S|=\min \{|S_i|\}$, Si is the subset of all satisfying dictionary. |*| indicates the number of set *. Finally, the solved minimal cover of the training set is the relation indicating words.

In a variety of real corpus, there are some high-frequency but meaningless words. It makes some meaningful words are left in the basket. This leads the weight of some keywords reduces. It has a bad influence in the post-processing of the character relation judgment. For that, we made some manual adjustments.

2.2 N layers Sentence Pattern Tree(N-SPT)

Construction of N-SPT.

For judging the specific relation between characters by certain sentence, the sentence need contain SPO triple that represent characters relation. Our approach takes the SPO triple consist of characters and relation indicating words as the core, and increases the number of characters layer-by-layer to extend patterns, which can obtain patterns with hierarchical structure to describe sentences in corpus.

- Definition of N-SPT

The paper present a kind of N layers Sentence Pattern Tree (N-SPT) based on relation indicating words and characters position relations in sentence and syntactic features, shown in Fig .1.

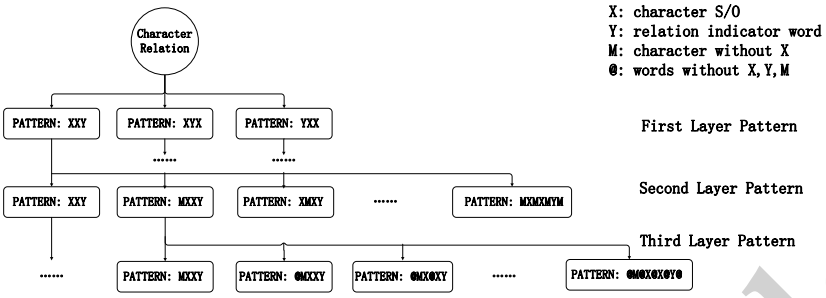


Fig. 1. Structure diagram of N-SPT

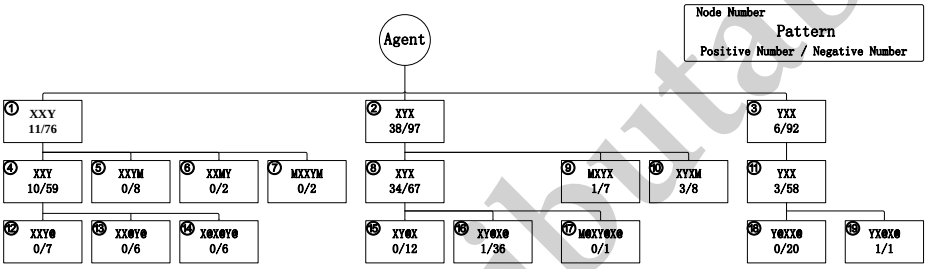


Fig. 2. Agent relation part of the N-SPT learned by the training set

The first layer of N-SPT only consider the location relation between characters X and relation indicating words Y, which consists of three classes: YXX, XYX , XXY. The location of relation indicating words is crucial to relation judgment. For example: suppose the relation type is “传闻不和” (Hearsay discord), the relation indicating word is “不和”. “赵薇周迅不和” meets the pattern XXY, which represent the relation is accurate. Nevertheless, “胡可不和程前演情侣” meets the pattern XYX, which represent the relation is error.

The second layer of N-SPT considers the influence of third person or personal pronoun M for relation judgment. Such as: “冯绍峰否认赵薇周迅不和”, “杨澜谈王菲李亚鹏分手”. For each pattern in the first layer, 24 patterns can be generated. For example: for “YXX”, can generate YXX (not contain the third character)、MYXX、YXMX、YXXM、MYMXXM and so on.

The third layer of N-SPT considers @ have an effect on the second layer (Word string @ only considers if there is any word exist, but not consider the specific content and number of words). For example: for MYXX, MYXY (not contain redundant string), M@YXX, MY@XX@, @MY@XX.

- Statistics of N-SPT

For given 19 relation, the paper build a Pattern Tree for each relation. Using the sentences processed in the training set, statically learning Pattern Tree. Parts of

the agent relation N-SPT learned by the training set is shown in Fig.2.

Characters relation judgment based on N-SPT.
According the strategy formulated by N-SPT, each sentence will match 3 patterns at most and 1 pattern at least. Using given sentence matches in N-SPT, the positive $PosNumT_i$ and negative number $NegNumT_i$ recorded in the node can be used to Judge the character relation, as the Equation 1 and 2 defined.

$$Tp_i = \frac{\min(PosNumT_i, NegNumT_i)}{\max(PosNumT_i, NegNumT_i)} \quad (1)$$

$$TemplateId = \min_i \ Tp_i \ \ (1 <= i <= 3) \quad (2)$$

3 Characters relation Judgment based on feature engineering

In this paper, we set the judging result of N-SPT as one-dimensional feature. Through the analysis of the corpus, we extract some features from text, and use a classifier to judge whether the triple extracted from sentence is correct.
This classifier only take the position of character word and relation word into consideration, instead of the property of character, the division of relation word, and the distance of character and relation word. We improved it with classifier of hybrid features. To such sentences which are filtered with the rule of heuristic approach, we extract features from the character attribute knowledge base, relation indicating words feature, word-spacing feature as the candidate of the feature classifier approach.

3.1 Feature extraction based on the character attribute knowledge base

- The character attribute feature

Aiming at each person in the character attribute knowledge base, including the name, gender, race, height, weight, occupation, the place of birth, registered residence, the date of birth and death, alias and so on, we select all of the above attributes except the name as features. At the same time, we select the number of attributes (not all attributes of a person we can get), the occurrence time of the character in the training data, the occurrence time of the first and second word of character's name as the candidate feature. In total, we have fifteen features.

- The combination features of character's attribute

According to character's attribute, the combination of two character properties

which need to be determined facilitates the determination of part relation. For instance, if the place of birth or the registered residence is same, the "fellow-townsman" relation is right; whether the gender of two characters is same, the "spouse" relation is wrong. Therefore, we defined four feature combinations as follows:

- Whether the place of birth or registered residence of two characters is same;
- The difference of two characters' age;
- Whether the gender of two characters is same;
- The length of the same prefix of two characters' name.

- The feature of relation indicating words

The relation indicating words is got through the approach introduced in chapter 2.1, and the kind of relation indicating words not only has a low dimension, but also can distinguish 19 types of relations effectively, 72 features in total.

- The distance feature between words

For some relations such as "暧昧", "闺蜜" and so on, after the analysis of training data, the distance of character and relation word determined the relation is right or not to a certain extent. At the same time, the N-SPT approach hasn't considered the distance feature of character words and relation indicating words. So, we calculate the distance as the candidate feature, SP distance and PO distance, 2 features in total.

3.2 Pattern tree features

- N-SPT feature

According to the given sentence, target character and the relation need to be judged. Firstly, we preprocess the sentences and identity whether the target character and relation indicating words are in the sentences or not. If they do not exist, we can judge the relation is error. If exist, we match the sentence with hierarchy. If the sentence matches the pattern, we record the right and wrong numbers and go deep in the next hierarchy; On the contrary, if the sentence could not match, we record the right and wrong number as -1. We set the right and wrong number of patterns as candidate features. In total, we get 6 features in all of the three hierarchies.

- N-SPT result feature

The effect of N-SPT is very good in the training data. The purpose of using feature classifier approach is to improve the judging effect of N-SPT. In hence, we set the judging result of N-SPT as one of the candidate features.

3.3 Feature selection

For the selected candidate features, we use entropy formula(formula (4)) to select the best feature for 19 relations, the Entropy(S) is the entropy of collection S, Gain(S,A)is the information gain of sentence collection S, Sv is the collection of correct or error relations

$$Entropy(S) = -p_+ \log_2 p_+ - p_- \log_2 p_- \quad (3)$$

$$Gain(S, A) = Entropy(S) - \sum_{v \in V(A)} \frac{|S_v|}{|S|} Entropy(S_v) \quad (4)$$

We first choose features with the information gain for each type of relations. Finally, we use the decision tree classifier to judge the character relations.

4 Experiment result

4.1 Experiment Data

- Definition of symbols

  S/O-----double entities
  P-------relation type

- Description of training data
  - 19 types of features.
  - The character attribute knowledge base (attribute, 12150 items)
    - Data type: random ID, entity ID, attribute name 1, attribute value 1 ..... attribute name n, attribute value n;
  - labeled training data(7813 items)
    - Data type: relation name P, entity S, entity O, sentence, the positive/negative examples (0/1), ID of entity S and ID of entity O in the character attribute knowledge base.
  - The testing data(2610 items)
    - Data type: relation name P, entity S, entity O, sentence, the positive/negative examples (0/1), ID of entity S and ID of entity O in knowledge database.
- Evaluation method
  Judging whether the SPO triple extracted from sentence is correct.

4.2 Evaluation of experiment

We use precision, recall, F1 value as the evaluation index. The formulas are

shown as follows:

$$precision_i = \frac{|\cap(predictionspo_i, referencespo_i)|}{|predictionspo_i|} \quad (5)$$

$$recall_i = \frac{|\cap(predictionspo_i, referencespo_i)|}{|referencespo_i|} \quad (6)$$

$$F1_i = \frac{2 * precision_i * recall_i}{precision_i + recall_i} \quad (7)$$

$$F1 = \frac{\sum_{i=1}^{n} F1_i}{n} \quad (8)$$

Meanwhile, predictionspo$_i$ is the number of SPO sets belong to ith relations judged by our approach, referencespo$_i$ is the number of SPO sets really belong to ith relations, n represents the 19 kinds of relations. We use F1 as the evaluation standard.

4.3 Experiments and results

Firstly, we preprocessed the sentences and removed the stop words and signals, and keep some important signals such as 《,》, “,” and so on. We use character ID to search the name, because the name is not unique. According to the analysis of sentences, we made some heuristic rules to assist the judgments of relations.

- If the sentence does not contain the relation indicating words, the relation is error;
- If the given name with adjacent word is another name, the relation is error. For instance, for“媒曝金妍儿与金元中疑似分手”,the“分手”relation of “金妍儿”and “金元” is “金元中”;
- If the given name or relation is contained by signals, the relation is error. For example: “苏醒谭维维等天娱艺人深情献唱《同学》”, “杨幂邓超着情侣睡衣拍《分手大师》” and so on;
- If the given name exist and its’ friends and relatives exist, for “唐一菲与凌潇肃母亲不和”, “安妮斯顿与男友贾斯汀前女友交心”, the relation is error.

For some filtered sentences, we firstly get the relation indicating word. Then judging the characters relations by N-SPT and use the 7813 sentences as training data, 2610 sentences as test data. We can achieve a F1 value 76.63%.

In chapter 2.3, we get the candidate features, and in 7813 items of the training data, we use cross-validation approach to select the best feature (the information entropy more than 0.01). We use the decision tree provided by WEKA[18] to judge the character relation (WDec classifier), F1 value is about 77.506%. Table 1 compared the experiment result of N-SPT and WDec classifier in detail.

Table 1. Result of the 19 relations Judge by N-SPT and WDec Classifier

Relation Type	N-SPT/WDec Precision (%)	N-SPT/WDec Recall (%)	N-SPT/WDec F1-Value (%)
同为校花	77.1/77.1	96.4/96.4	85.7/85.7
昔日情敌	85.7/85.7	85.7/85.7	85.7/85.7
老师	67.4/71.0	54.7/51.9	60.4/60.0
撞衫	81.1/81.1	88.2/88.2	84.5/84.5
前女友	65.2/65.2	100/100	78.9/78.9
偶像	76.3/76.3	85.5/86.6	80.6/81.1
暧昧	78.4/78.4	80.0/81.6	79.2/80.0
绯闻女友	88.0/88.0	84.6/84.6	86.3/86.3
传闻不和	73.5/59.4	67.6/67.6	70.4/68.5
前妻	87.5/87.5	77.8/77.8	82.4/82.4
闺蜜	69.2/69.2	36.0/37.5	47.4/48.6
妻子	73.2/73.2	87.2/91.1	79.6/81.2
朋友	57.4/71.1	79.5/69.2	66.7/70.1
分手	74.1/76.9	66.7/66.7	70.2/71.4
翻版	71.4/71.4	71.4/71.4	71.4/71.4
同学	87.5/87.5	63.6/63.6	73.7/73.7
经纪人	80.0/80.0	85.7/92.3	82.8/85.7
老乡	69.7/79.3	88.5/92.0	78.0/85.2
同居	90.6/90.6	93.5/93.5	92.1/92.1
Total			76.6/77.5

Table 2. List of the comparison for each classifier in Training Time, Testing Time and F1 value

Approach	Training Time	Testing Time	F1-Value
WDec Classifier	6.1min	1.4min	77.51%
N-SPT	1.4min	0.45min	76.63%
BestResult	about 30min	about 30min	75.68%

We use N-SPT to judge the character relation directly. The F1 value is about 76.63%. WDec is more than 77.506%. We also compared the time of training, testing and F1-value (shown in table 2). The result demonstrates that our approach is better than the BestResult [19] in the data set.

4.4 Experiment analysis

The experiment use N-SPT to judge whether the character relation is correct. The training time, testing time and F1-value are all better than BestResult in the dataset. It demonstrates N-SPT can judge the character relation efficiently and accurately. The reason that our approach achieves desired results and disadvantages of our method are shown as follows:

- BestResult uses N-Gram features, dependency tree features to judge characters relation. The dimension of features is very high, it leads the training time and testing time consume too much. But in our approach, the N-SPT proposed by us has a good summary of the corpus features and the feature dimension is very low. It improves the efficiency of training and testing.
- According to the experiment result, WDec has advantage in 17 kinds of relations. It is proved that the character attribute knowledge base feature, relation indicating words feature, the distance of words make up the disadvantage of N-SPT. The improve approach is effective.
- For some sentences such as “范冰冰骂哭赵薇那英田震反目”, “郭碧婷郭采洁反目”, “顾里郭采洁众叛亲离”, “关之琳上正妻黑名单刘嘉玲抢闺蜜男友” and so on, our method cannot judge whether two characters is contained in one sentence. We can divide the sentence into parts to solve this problem in the further work.
- N-SPT has guiding significance in building training dataset, as shown in Figure 2. When the N-SPT is not complete, for example, in node ⑤⑥. The layer is less than 3, and we can add some sentences which match the sub-node of ⑤⑥ to the training data. The quality of training data will be improved by completing the N-SPT.
- Compared with the word-bag model, N-SPT can locate the wrong position very fast, and adjust accordingly. For the relation “同居”, “容祖儿想跟胡歌同居”, “容祖儿跟胡歌同居”, these sentences can all match the template of “X@XY” because the word "想" has a effect on the judging result. But the pattern “X@XY” is correct, so that we can make judging rule “X 想@XY” is error.

5 Conclusion and further work

5.1 conclusion

Our approach is a supervised approach using the training data to construct the tree patterns. Compared with traditional works, our approach can construct patterns of the entire training data with little manual participation. When the domain is changed, we just need to adjust some coefficient to construct new patterns. In the retrieval and restoration, the effect of training and testing are both very high. Because the character and relation indicating words are given, we do not need to process the other parts of these sentences.

We just considered the tree structure in generating the pattern, ignored the attribute of character. Pointed at this fault, we added the character property knowledge database, combination features of character attributes, relation indicating word and so on. The experiment result shows these improvements have good effects.

5.2 further work

The N-SPT presented in this paper works well while processing sentences with concise and simple structures, but it still needs improvement when handling more complex sentences, It still leads to relatively big error when matched to the third level of N-SPT, but the current N-SPT only has three levels, so it can be treated with clustering approaches, such as K-MEANS, hierarchical clustering and LDA, etc, clustering the words in the rest character strings @ on the third level template, and clustering the words which affect relation determination into particular category, thus expending it into the fourth or even deeper levels. N-SPT is highly extensible, so the next focus of this paper will be how to extend N-SPT to even deeper levels in order to process complex sentences. The current way of constructing N-SPT with training is to build a N-SPT for each relation, but there might be several reference words for each relation, and the different usage of each reference word might result in the error of building N-SPT for relations, While building a N-SPT for each reference word might also cause data sparsity problem, so further research is required in order to balance the difference usage of the reference words for relations of N-SPT and the data sparsity problem. This paper only tried decision tree to categorize different combined features at present, different classifiers will be used to test their effect on relation determination in follow-up researches.

Reference

1. Kluegl P, Toepfer M, Beck P D, et al. UIMA Ruta: Rapid development of rule-based infor-

mation extraction applications [J]. Natural Language Engineering, 2016, 22(01): 1-40.
2. Kozareva and E. Hovy. 2010. Learning arguments and supertypes of semantic relations using recursive patterns. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 1482–1491. Association for Computational Linguistics.
3. Fang Y, Chang K C C. Searching patterns for relation extraction over the web: rediscovering the pattern-relation duality [C]//Proceedings of the fourth ACM international conference on Web search and data mining. ACM, 2011: 825-834.
4. Qin, Bing, Liu, An'an, Liu, Ting. Unsupervised Chinese Open Information Extraction [J]. Journal of Computer Research and Development, 2015, (5): 1029-1035
5. Etzioni O, Fader A, Christensen J, et al. Open Information Extraction: The Second Generation[C]//IJCAI. 2011, 11: 3-10
6. Carlson A, Betteridge J, Kisiel B, et al. Toward an Architecture for Never-Ending Language Learning[C]//AAAI. 2010, 5: 3
7. Lim S, Lee C, Ra D. Dependency-based semantic role labeling using sequence labeling with a structural SVM[J]. Pattern Recognition Letters, 2013, 34(6): 696-702.
8. Gamallo P, Garcia M, Fernández-Lanza S. Dependency-based open information extraction[C]//Proceedings of the joint workshop on unsupervised and semi-supervised learning in NLP. Association for Computational Linguistics, 2012: 10-18.
9. Li SQ, Zhao TJ, Li HJ, Liu PY, Liu S. Chinese semantic role labeling based on feature combination[J]//Journal of Software. 2011, 22(2):222−232.
10. Li JH, Zhou GD, Zhu QM, Qian PD. Semantic role labeling in Chinese language for nominal predicates. Journal of Software[J]// 2011,22(8):1725−1737.
11. Christensen J, Soderland S, Etzioni O. An analysis of open information extraction based on semantic role labeling[C]//Proceedings of the sixth international conference on Knowledge capture. ACM, 2011: 113-120.
12. Sun A, Grishman R, Sekine S. Semi-supervised relation extraction with large-scale word clustering[C]//Proceedings of the 49th Annual Meeting of the Association for Computati0onal Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, 2011: 521-529.
13. Weston J, Bordes A, Yakhnenko O, et al. Connecting language and knowledge bases with embedding models for relation extraction [J]. arXiv preprint arXiv:1307.7973, 2013.
14. Zahedi M, Kahani M. SREC: Discourse-level semantic relation extraction from text [J]. Neural Computing and Applications, 2013, 23(6): 1573-1582.
15. Nie T, Shen D, Kou Y, et al. An Entity Relation Extraction Model based on Semantic Pattern Matching[C]//Web Information Systems and Applications Conference (WISA), 2011 Eighth. IEEE, 2011: 7-12.
16. Glass M, Barker K. Bootstrapping relation extraction using parallel news articles[C]//Proceedings of the IJCAI workshop on learning by reading and its applications in intelligent question-answering, Barcelona. 2011.
17. Apostolova E, Tomuro N. Combining Visual and Textual Features for Information Extraction from Online Flyers[C]//EMNLP. 2014: 1924-1929.
18. Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten (2009); The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1.
19. Zhang, Zhihua, Wang, Jianxiang, Tian, Junfeng, Wu, Guoshun, LAN, Man. Blocked person relation recognition system based on multiple features [J]. Journal of Computer Applications, 2016, (3): 751-757

Biomedical Event Trigger Detection Based on Hybrid Methods Integrating Word Embeddings

Lishuang Li, Meiyue Qin, Degen Huang

School of Computer Science and Technology, Dalian University of Technology, Dalian China
lilishuang314@163.com, qinmessiy@163.com, huangdg@dlut.edu.cn

Abstract. Trigger detection as the preceding task is of great importance in biomedical event extraction. By now, most of the state-of-the-art systems have been based on single classifiers, and the words encoded by one-hot are unable to represent the se-mantic information. In this paper, we utilize hybrid methods integrating word embeddings to get higher performance. In hybrid methods, first, multiple single classifiers are constructed based on rich manual features including dependency and syntactic parsed results. Then multiple predicting results are integrated by set operation, voting and stacking method. Hybrid methods can take advantage of the difference among classifiers and make up for their deficiencies and thus improve performance. Word embeddings are learnt from large scale unlabeled texts and integrated as unsupervised features into other rich features based on dependency parse graphs, and thus a lot of semantic information can be represented. Experimental results show our method outperforms the state-of-the-art systems.

Keywords: Trigger detection ·Word embeddings ·Hybrid Methods ·Rich features

Introduction

With the development of the Internet, a vast and ever-expanding body of natural language text is becoming increasingly difficult to leverage. This is particularly true in the domain of life science, where biomedical articles are increasing exponentially. We need to automatically extract interested and structured information from biomedical text, which is known as biomedical text mining. In the past, the focus in the field of biomedical text mining was named entity recognition (NER). In recent years, the focus has shifted to relation extraction, especially complex relation extraction which is more difficult than simple binary relation extraction. Biomedical event is one type of complex relation. Trigger, argument and the event type need to be detected when extracting an event. Event extraction systems consist of at least two parts: trigger detection and argument detection, while trigger detection is the preceding task. Thus, trigger detection is of great importance in biomedical event extraction.

Trigger detection aims to detect a span of text that triggers an event. The methods for trigger detection fall into four categories: dictionary-based, rule-based, statistical machine learning and combined methods in which the statistical machine learning method is dominant. Trigger detection is regarded as multiclass classification task in most of the state-of-art event extraction systems. Björne et al. (2009) extracted rich manual features including token features, frequency features and dependency chains and so on. They adopted these features and multiclass classification tool SVM^{multiclass} to detect triggers. Their event extraction system achieved the best performance using this trigger detection. Martinez and Baldwin (2011) regarded trigger detection as word sense disambiguation (WSD) problem and found that WSD outperformed sequential tagging and could improve the performance of sequential tagging methods. They achieved 60.1% F-score on the set of BioNLP'09. Zhang et al. (2013) efficiently mapped the dependency graph of a candidate sentence into semantic/syntactic features, and used these semantic/syntactic features to detect bio-event triggers from the biomedical literature. Their method achieved an F-score of 65.84% on the set of BioNLP'09. Trigger detection was viewed as sequence labeling task by Majumder et al. (2012). They designed elaborate features, such as the frequency of named entity in sliding window, dependency path and adopt Conditional Random Field (CRF) to extract triggers with feature template. The F-score achieved 67.0% on the set of BioNLP'09. Wang et al. (2013) proposed a method based on the deep syntactic analysis. They adopted deep syntactic information to detect triggers and arguments with LibSVM. The results from arguments detection were integrated into trigger detection. They achieved 68.8% and 67.3% F-scores on BioNLP'09 and BioNLP'11 respectively.

The previous works were mostly based on single models, Domingos (2012) pointed out one model was not sufficient. On one task, many models can be constructed and their results can be combined based on different techniques. Ensemble techniques include set operation, voting and stacking, etc. Li et al. (2012) discussed the three techniques on NER task which was regarded as a sequence labeling problem. Due to the relearning process in stacking, the stacking technique

outperformed the other two which directly operated the predict results. Similar to NER, the statistical machine learning methods for trigger detection can be integrated under the construction of single models. In this work, we construct four different models based on two SVM models trained separately using one vs. one and one vs. rest multiclass extension methods, Passive aggressive online algorithm (PA) (Crammer, 2006) and Random Forest (RF) (Breiman, 2001). And then the results from four models are integrated with different ensemble techniques.

On the other hand, the way to digitalize features in previous works was one-hot encoding. The main problem of this method is that it is unable to represent the semantic information. Recently, word embeddings, a vector related with a word, are used in several NLP problems, such as named entity recognition (NER), chunking, and make a contribution to the improvement. Tang et al. (2014) explored the effect of word embeddings on biomedical NER. Turian et al. (2010) discussed its effect on several tasks, including NER and chunking. In this work, we utilize hybrid methods integrating word embeddings to predict trigger in biomedical event. Experimental results show our method outperforms the state-of-the-art systems.

The remaining part of this paper is organized as follows: preliminary algorithms are described in Section 2. Our proposed method is described in Section 3. Experimental results and analysis are illustrated in Section 4. Comparisons are given in Section 5. Finally, discussion and conclusions are shown in Section 6 and Section 7 respectively.

Preliminary Algorithms

Online Passive-aggressive Algorithm

Passive-aggressive (PA) online algorithm is an online algorithm based on perception. The main idea of the algorithm is the maximum classification margin adopted in SVM. It updates the classifier using the current instance greedily and predicts the current instance correctly with the maximum margin and remains the new classifier as close as possible to the current one.

In order to improve the robustness of a classifier and reduce the number of possible combinations, several outstanding classifiers after optimized on the parameter *C* are selected and the mean of selected classifiers is adopted. In our work, the trigger class with the highest scores is regarded as the predicted results when using online algorithms. The interested readers can refer to (Crammer, 2006) for more details.

Support Vector Machines

Support vector machines (SVM) first introduced by Vapnik are learning systems that use a hypothesis space of linear functions in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias derived from statistical theory (Vapnik, 1995; Cristianini and John Shawe-Taylor, 2000).

Given training examples:

$$S = \{(x_1, y_1), (x_2, y_2), \cdots, (x_l, y_l)\}, x_i \in R^n, y_i \in \{-1, +1\},$$

x_i is a feature vector (n dimension) of the *i-th* sample. y_i is the class (positive (+1) or negative(-1) class) label of the *i-th* sample. is the number of the given training samples. SVMs find an "optimal" hyper-plane: $(w*x+b)=0$ to separate the training data into two classes. The optimal hyper-plane can be found by solving the following quadratic programming problem (Vapnik, 1998):

$$\max \sum_{i=l}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=l}^{l} \alpha_i y_i \alpha_j y_j K(x_i, x_j) \quad (1)$$

subject to $\sum_{i=l}^{l} \alpha_i = 0$, $0 \leq \alpha_i \leq c$, $i = 1, 2, \ldots, l$

The function $K(x_i, x_j)$ is called kernel function:

$$K(x_i, x_j) = \Phi(x_i) \bullet \Phi(x_j) \quad (2)$$

Given a test example, its label y is decided by the following function:

$$f(x) = \mathrm{sgn}[\sum_{x_i \in SV}^{l} \alpha_i y_i K(x_i, x) + b] \quad (3)$$

Random Forests

Random forests (RF) are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest.

Significant improvements in classification accuracy have resulted from growing an ensemble of trees and letting them vote for the most popular class. In order to grow these ensembles, often random vectors are generated that govern the growth of each tree in the ensemble. An early example is bagging (Breiman, 1996), where to grow each tree a random selection (without replacement) is made from the examples in the training set. Another example is random split selection (Dietterich, 2000) where at each node the split is selected at random from among the *K* best splits. For the *k*th tree, a random vector θ_k is generated, independent of the past random vectors $\theta_1, \ldots, \theta_{k-1}$ but with the same distribution; and a tree is grown using the training set and θ_k, resulting in a classifier $h(x, \theta_k)$ where x is an input vector. For instance, in bagging the random vector θ is generated as the counts in N boxes resulting from N darts thrown at random at the boxes, where N is number of examples in the training set. In random split selection θ consists of a number of independent random integers between 1 and *K*. The nature and dimensionality of θ depend on its use in tree construction.

After a large number of trees are generated, they vote for the most popular class. We call these procedures random forests.

Word Embeddings

A distributed representation, also known as word embeddings, is dense, low dimensional, and real-valued. Word embeddings are typically induced using neural language models, which uses neural networks as the underlying predictive model. There are several word embeddings, such as Collobert and Weston embeddings(C&W) (Collobert et al., 2011), HLBL embeddings (Mnih and Hinton, 2008) and Word2Vec (Mikolov et al., 2013a; Mikolov et al., 2013b).

Considered the time and hardware requirements in different distributed representation methods, Word2Vec was adopted in our work. Word2Vec supplies two models: CBOW and Skip-gram. The Skip-gram model extended on n-gram model is used and shown in Figure 1. It aims to optimize the classification of a word based on other words in the same sentence within a certain range before and after the current word. This tool can generate a dense, low-dimensional, and real-valued vector, which may capture the syntactic and semantic information in each dimension. This information cannot be obtained from words encoded by one-hot.

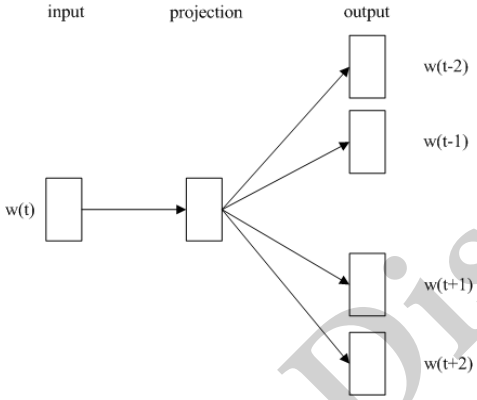


Fig. 1. The Skip-gram architecture.

Our Methods

Features Extraction

In this work, five kinds of features are mainly used, token, frequency, dependency chains, shortest path and word embeddings. The dependency paths parsed by McClosky-Charniak parser (McClosky and Charniak, 2008) and Enju parser (Miyao et al., 2009) are added into the features. Compare to the previous researches in BioNLP, our system extracts more features, which have greatly improved the performance. The features we employ are:

Token features include current token text, POS, stem, binary tests for presence of uppercase, digital or special characters, bigrams and trigrams of the token. Dependency context is of great importance for trigger detection, so we extract token features of candidate triggers in dependency context and linear context besides candidate triggers themselves.

- Token text includes current token and the tokens within a window of three tokens before and after the target tokens.
- POS includes the POS of the current token and the tokens within a window of three tokens. The POS is tagged with McClosky-Charniak parser.
- Stem consists of the stem of the current token, obtained by Porter stemmer (Porter, 1980). This feature can alleviate the effect of morphological changes, such as "involvement" and "involves".
- Binary features include binary tests for presence of uppercase, digital or special characters. Some words with a negative class may contain digitals or capital letters. Some triggers contain special characters, such as "up-regulation", "co-transfected".
- Bigrams and trigrams consist of two or three continuous characters in current token. For example, for the token "binding", its trigrams are "bin", "ind", "ndi", "din", and "ing".

Frequency features are defined as the number of named entities in the current sentence and the context of a candidate trigger, and the frequency of words in bag-of-words. It is obvious that the more entities in a sentence there are, the more likely triggers exist in the current sentence. For the frequency of words in bag-of-words, we take this sentence for an example, "The p53 paradox in the pathogenesis of tumor progression.", the frequency of words in its bag-of-words are "*the:2*", "*p53:1*", "*paradox:1*", "*in:1*", "*pathogenesis:1*", "*of:1*", "*tumor:1*", "*progression:1*", "*.:1*" and "*PROTEIN:1*". Here, the protein names are all replaced with "PROTEIN".

Dependency chains up to depth of three are constructed. When the window size is not large enough, the important information related with candidate triggers may not be considered. Therefore dependency information is added.

Token features of nodes in dependency chains include POS of the token, the token and whether the node is protein or not. These features are added with position information (the distance from proteins) in dependency chains.

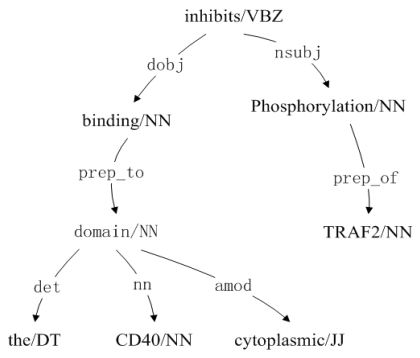


Fig. 2. An example of dependency parsing

Dependency types in dependency chains are also added with position information, sequence of dependency type and direction. An example of dependency parsing is shown as Fig. 2. For the token "inhibits", its dependency chains features are: "*1_binding*", "*1_NN*", "*1_dobj*", "*1_dobj_NN*", "*1_dobj_binding*", "*1_Phosphorylation*", "*1_NN*", "*1_nsubj*", "*1_nsubj_NN*", "*1_nsubj_Phosphorylation*", etc.

Shortest path includes *n*-grams (*n*=2, 3, 4) of the edges in the shortest dependency path between candidate triggers and the nearest protein, and the combinations of the entity types in the shortest path. For more details, please refer to (Miwa et al., 2010).

Word embeddings involve the vectors of the current token. The dimension of the vectors is decided by experiments.

Divergent Classifiers

In our experiment, we utilize three different toolkits and adopt different training algorithms to construct four classifiers.

- PA: follows the maximum edge theory and has good generalization ability like SVM.
- SVM1vs1 and SVM1vsrest: two SVM models trained by one vs. one multi-class extension method and one vs. rest multi-class extension method.
- RF: a combination of tree predictors, and after a large number of trees are generated, they vote for the most popular class.

Hybrid Methods

Our system uses three different ensemble methods which are set operations, voting methods and stacking method to combine the four single classifiers' results. Firstly, the set operations and voting methods which do not need retraining process are adopted to combine the classification results from the four models. They both cost less time than the stacking method because the latter needs retraining. For example, the stacking method with *n*-fold cross evaluation on the training corpus costs much more training time than the combining methods with no retraining. The three hybrid methods are presented in detail in following sections.

Union and Intersection Operation .

According to the union operation, both classification results from two classifiers are classified as the correct results. Obviously this method will make the recall improved but make the precision decreased compared to each single classifier. On the contrary, the intersection of two classifiers will only take the common results as the correct results, which will make the precision improved but the recall decreased. In order to make a trade-off between recall and precision, we perform union or intersection operations on the results from different models depending on precision and recall of different models.

Voting.

The majority voting method assumes that triggers are correctly predicted by most individual systems while different systems cannot get consistent results. The pseudo code of the voting method used in this paper is described in below:

```
Input: predicting result of single
   classifiers for one trigger in-
   stance.
Output: predicting type of trigger

Voting:
   Initial: set result_voting to 1
   by default and elements of array
   count to 0,
   count[1],count[2],…,count[10]
   represent the number of vote for
   each class, respectively.

   Calculate
   count[1],count[2],…,count[10]

   max_value, index = the max value
   in array count and the index of
   max value in array respectively.

   if max_value == 1:
     result_voting = the highest pre-
     diction result of single classi-
     fiers
   else:
     result_voting = index

return result_vote
```

Stacking Method.

Most stacking methods adopt the two-layer framework. The training process is separated into two steps and is described as follows:

- Step 1: *n*-fold cross validation is adopted on the single classifier of the layer-0. Given a data set $D = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ and *k* different learning algorithms, we split the data set *D* into *n* almost equal parts; At each training and testing process, choosing one part as testing corpus and the other *n*-1 parts as training corpus; For this part of testing corpus, we get *k* different classification results from the *k* classifiers. After *n* times training and testing like this, we get *k* different results on entire data set *D*; then we combine the *k* results and the manually

annotated results of *D*, and then get a new training set D_1 for the layer-1.

- Step 2: At this step D_1 is utilized as the training corpus to construct a classifier model based on a learning algorithm and its testing results on the testing corpus are the final results.

The four classifiers described in Section 3.2 are used as the base classifiers at layer-0, and RF is chosen as the classifier at layer-1 because the framework in terms of strength of the individual predictors and their correlations gives insight into the ability of the random forest to predict.

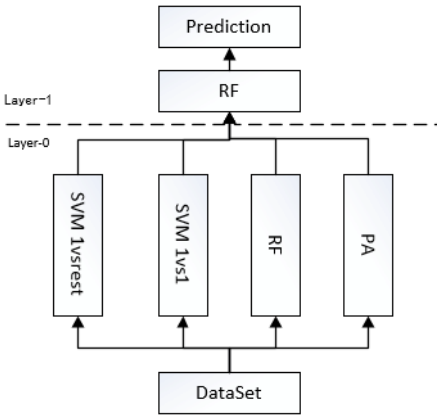


Fig. 3. Two-layer stacking architecture of hybrid method

In the training process, we use 5-fold cross validation to get the predicting results of the four kinds of classifiers on BioNLP'09 and BioNLP'11 training sets respectively. Then we regard the four results as feature vectors to construct a new training set for the classifier at layer-1. Another work we have to do at layer-0 is constructing four classifiers based on the whole training corpus and predicting the classification results on BioNLP'09 and BioNLP'11 development sets based on them respectively. In the same way we combine the four results of single classifiers and get the new testing corpus for the classifier at layer-1. The two-layer stacking frame is shown in Fig. 3.

Experiments and Results

Corpus and Evaluation

All experiments are conducted on the corpora supplied by BioNLP'09 (Kim et al., 2009) and BioNLP'11 (Kim et al., 2011). And the parameters are optimized by using 5-fold cross evaluation on training set. The evaluation criterion P(recision)/R(ecall)/F(-score) is adopted, which is defined as formula (4), where *TP*, *FP* and *FN* are short for True Positives, False Positives and False Negatives respectively.

$$P = \frac{TP}{TP+FP},\ R = \frac{TP}{TP+FN},\ F\text{-}score = \frac{2*P*R}{P+R} \quad (4)$$

Results of Trigger Detection Integrating Word Embeddings Based on PA

To illustrate the impact of word embeddings on trigger detection, we choose PA without word embeddings as baseline. Five groups of experiments are conducted on the development set of BioNLP'09 with different dimensions of the word vectors. The dimension of the vectors is set to 50, 100, 200 and 400 respectively to compare the influence of word embeddings on trigger prediction. The results are shown in Table 1, and our baseline is using all features except word embeddings. BaselineWE50, BaselineWE100, BaselineWE200 and BaselineWE400 mean the dimensions of word embeddings are 50, 100, 200 and 400 respectively when word embeddings are integrated. The type with the highest score is the final result.

Table 1. The results with different dimensions of the word vectors on trigger prediction

Features	Precision	Recall	F-score
Baseline	72.19%	71.33%	71.76%
BaselineWE50	73.93%	70.45%	72.15%
BaselineWE100	74.44%	71.41%	72.89%
BaselineWE200	74.17%	71.81%	72.97%
BaselineWE400	74.58%	71.49%	**73.00%**

From Table 1, we can see all the F-scores using word embeddings are improved compared with Baseline. The F-score improves with the increase of dimension on trigger prediction. The F-scores are improved by 0.39~1.24% with the variance of the dimension of the vectors, which illustrates that the syntactic and semantic information carried by word embeddings has significantly increased the performance.

Table 2. Results based on four single classifiers on BioNLP'09 and BioNLP'11 respectively

Task	Model	Precision	Recall	F-score
BioNLP'09	PA	74.58%	**71.49%**	**73.00%**
	SVM 1vs1	74.30%	69.73%	71.94%
	SVM 1vsrest	**80.02%**	64.30%	71.30%
	RF	79.57%	53.51%	63.99%
BioNLP'11	PA	74.57%	**72.74%**	**73.64%**
	SVM 1vsrest	**81.35%**	64.61%	72.02%
	SVM 1vs1	73.39%	67.77%	70.47%
	RF	78.06%	56.85%	65.79%

Results Based on Four Single Classifiers

Table 2 shows the results from the four single classifiers in shared task BioNLP'09 and BioNLP'11 development sets respectively. We can see that the PA model constantly outperforms the other models. Although RF gets a higher Precision of 79.57%, the lowest Recall leads to the lowest F-score (63.99%).

Results Based on Union and Intersection Operation Method

The performance combining the results of the different models using the simple set operations is shown in Table 3. We conduct the union operation (denoted by the symbol ‘‘∪’’) on the results of SVM 1vsrest，SVM1vs1 and PA as they are all based on maximum-margin theory. It can be seen that Recall increases by 3.19% (74.68% vs 71.49%) compared with PA which achieves the highest Recall among the single classifiers. We also intersect (denoted by the symbol ‘‘∩’’) them to improve Precision 3.33% (83.35% vs 80.02) higher than SVM1vsrest which gets the best Precision. On the basis of the above set operation, we try to use RF (not based on maximum-margin theory) to improve the performance. However, due to the poor performance of RF, it decreases the F-score (shown as the third and fourth row in Table 3). From Table3, we can see that SVM1vsrest ∪ SVM1vs1∩PA can get the best F-score of 73.38% which is 0.38% higher than PA (73%).

Table 3. Results using simple set operation methods on BioNLP’09

Method	Precision	Recall	F-score
PA∪SVM 1vs1∪ SVM 1vsrest	70.25%	**74.68%**	72.40%
PA∩SVM 1vs1∩ SVM 1vsrest	83.35%	60.38%	70.03%
PA∪SVM 1vs1∪ SVM 1vsrest∪RF	68.90%	75.00%	71.82%
PA∩SVM 1vs1∩ SVM 1vsrest∩RF	**87.39%**	49.28%	63.02%
SVM_U:SVM 1vsrest∪ SVM 1vs1	73.03%	72.44%	72.73%
SVM_U∩PA	78.45%	68.93%	**73.38%**
SVM_I:SVM 1vsrest∩ SVM 1vs1	82.45%	61.18%	70.24%
SVM_I∪PA	74.22%	72.20%	73.20%

Results Based on Voting Method

Table 4. Results based on voting method on BioNLP’09

Method	Precision	Recall	F-score
PA+SVM1vs1+SVM1vsrest	77.41%	69.80%	73.41%
RF+SVM1vs1+SVM1vsrest	80.65%	63.90%	71.30%
PA+RF+SVM1vs1	77.38%	68.85%	72.87%
PA+RF+SVM1vsrest	80.44%	64.70%	71.71%
PA+SVM1vs1+SVM1vsrest+ RF	**81.71%**	63.18%	71.26%

Some experiments are conducted to investigate the effectiveness of the voting algorithm. The results are shown in Table 4. The voting method (PA+ SVM1vs1+SVM1vsrest) gets a better F-score (73.41%) than the simple set operations method. The reason may be that it’s easy to reach agreement on the same instance with similar classifications, thus the voting results are more reliable. From the Table 4, we can also find that RF as one member of voting groups may decrease the final result because of its poor performance.

Results Based on Stacking Method

The following three groups of experiments are conducted in the stacking method: (1) Choose two classifiers which get the best Recall and Precision as base classifiers at layer-0 and the stacking results are regarded as our baselines, denoted by baseline1 and baseline2 respectively. (2) Add different classifiers to the baselines. From Table 5 and Table 6 it can be seen that after adding a different classifier, all of F-scores are improved than both baselines respectively. We can also find that adding RF can get the better performance though its performance is poor. Therefore, the diversity among different classifiers plays an important role in stacking method. (3) Use all four classification results as base classifiers at layer-0. We can find that the F-score is under baseline1, which means that more classifiers may not achieve better performance.

Table 5. Results based on two-layer stacking method on BioNLP’09.

Layer-0 Method	Precision	Recall	F-score
PA+SVM1vs1 (baseline 1)	76.12%	**70.29%**	73.09%
RF+SVM1vsrest (baseline 2)	**80.20%**	64.38%	71.42%
PA+SVM1vs1+SVM 1vsrest	77.96%	70.05%	**73.79%**
PA+SVM1vs1 + RF	77.46%	69.73%	73.39%
RF+SVM1vsrest + PA	78.88%	67.73%	72.88%
RF+SVM1vsrest +SVM 1vs1	79.18%	66.53%	72.31%
PA+SVM1vsrest+SVM1vs1 +RF	78.56%	68.21%	73.02%

Table 6. Results based on the two-layer stacking method on BioNLP’11.

Layer-0 Method	Precision	Recall	F-score
PA+SVM1vs1 (baseline 1)	75.58%	72.60%	74.06%
RF+SVM1vsrest (baseline 2)	**81.02%**	65.03%	72.15%
PA+SVM1vs1 + SVM 1vsrest	75.81%	72.64%	74.19%
PA+SVM1vs1 + RF	76.13%	**72.46%**	**74.25%**
RF+SVM1vsrest + PA	78.31%	69.25%	73.50%
RF+SVM1vsrest + SVM 1vs1	80.86%	65.35%	72.28%
PA+SVM1vsrest+SVM1vs1+RF	77.64%	69.35%	73.26%

From Table 5, we can find that the group of PA + SVM1vs1 + SVM1vsrest can get the best performance (73.79% F-score) on BioNLP’09 which is 0.79% higher than PA which achieves the best F-score (73%) among single classifiers. The same stacking experiments are executed in task BioNLP’11 (shown as Table

6), and we can get a similar conclusion. Compared to the single classifier's best F-score (73.64%), the stacking method improve the F-score by 0.61% on BioNLP'11.

Comparisons

Comparisons of Performance of Different Methods

The comparison among the results of the union and intersection operation methods, voting algorithm, two-layer stacking method and the single classifier PA is shown in Table 7. Here we regard the result of PA as a baseline because of its best performance among four single classifiers. From Table 7 we can see that all hybrid methods yield better results than each single classifier. Compared with PA, the three different ensemble methods all improve the precision but decrease the recall in task BioNLP'09. Furthermore, the two-layer stacking method achieves better performance than the other two hybrid methods.

Table 7. Comparisons of performance on different methods on BioNLP'09.

Method	Precision	Recall	F-score
PA (baseline)	74.58%	71.49%	73%
Union and intersection method	78.45%	68.93%	73.38%
Voting (three classifiers)	77.41%	69.80%	73.41%
Two-layer stacking algorithm	77.96%	70.05%	**73.79%**

Comparisons with Other Work

Finally, we make comparisons between our systems and some related work in Table 8. We achieve the best performance on BioNLP'09 and BioNLP'11 development sets. The F-scores are higher than the current best system Wang et al. (2013) by 4.99% and 6.95% respectively.

Wang et al. (2013) proposed a trigger extraction method based on the deep syntactic analysis. Deep syntactic information was used for argument detection, and then the result was merged into the trigger extraction phase. This method achieved 68.8% and 67.3% F-scores on BioNLP'09 and BioNLP'11 respectively. Martinez and Baldwin (2011) regarded trigger classification as a word sense disambiguation (WSD) problem. In the task of BioNLP'09, the F-score reached 60.1%. Majumder (2012) took trigger classification as a sequential tagging task and extracted rich features such as frequency of named-entities in sliding window, POS of word, whether protein or others and name of nearest protein etc. They used CRF tool to tag sequences and achieved an F-score of 67.0% on BioNLP'09. Zhang et al. (2013) used the hash operation to iteratively compute the dependency graph and mapped the dependency graph into neighborhood hash features. Then they combined other basic features, bag-of-words features, frequency features and token features based on SVM. Finally, their approach achieved an F-score of 65.84% on BioNLP'09.

The main difference between our method and the other four methods exists in three aspects: (1) The rich features are the solid foundation, such as token features, syntactic and dependency features, the shortest path. (2) Word embeddings, which can learn much deeper syntactic and semantic information from the large set of out-of-domain data obtained through unsupervised learning, lead to the vectors of words with common semantics are close to each other, and thus improve trigger detection. (3) Hybrid methods: multiple classification results are combined to further improve the performance.

Table 8. Comparisons between our system and some related work.

System	Task	Precision	Recall	F-score
Ours	BioNLP'09	77.96%	70.05%	**73.79%**
	BioNLP'11	76.13%	72.46%	**74.25%**
Wang et al.'s	BioNLP'09	75.30%	64.00%	68.80%
	BioNLP'11	69.50%	56.90%	67.30%
Martinez et al.'s	BioNLP'09	70.20%	52.60%	60.10%
Majumder's	BioNLP'09	69.96%	64.28%	67.00%
Zhang et al.'s	BioNLP'09	79.83%	56.02%	65.84%

Discussion

The three ensemble methods give better performance than every single model. The main reason is that the hybrid methods can exploit the diversity or consistency among different classifiers to make a final decision on the basis of single models. For instance, the trigger "transfection" is classified as Regulation by PA, on the contrary, all of the other classifiers categorized it as Positive regulation". After voting it is marked as "Positive regulation" which is consistent with the correct result.

Among all the three hybrid methods in our paper (set operation, voting and stacking), the stacking method performs best owing to its capability of relearning from the original learning at layer-0. In the relearning process for RF, after a large number of trees are generated, they vote for the most popular class. For example, "Overexpression", which is categorized as Regulation by voting according to most classifiers' results, can be marked correctly as "Gene expression" by the stacking method through the relearning process.

Word embeddings play an important role which implies a lot of useful information, including syntactic and semantic. For example, for the two words, "diminished" and "reduced", they have little common features directly in morphology, but the similarity between their word embeddings measured by cosine similarity is up to 0.897. By using word embedding, the performance on trigger prediction is improved.

Conclusion

The proposed method improves the performance of trigger detection, outperforming most of published works. First, rich features are the solid foundation. Second, word embeddings play an important role. Finally, the hybrid methods make full use of the advantages of different classifiers by combining their results to get a higher performance. By integrating the rich features and word embeddings into hybrid method, our system outperforms the state-of-the-art systems.

Acknowledgment.
The authors gratefully acknowledge the financial support provided by the National Natural Science Foundation of China under No. 61672126, 61173101, 61173100.

Reference

Majumder A.: Multiple Features Based Approach to Extract Bio-molecular Event Triggers Using Conditional Random Field. International Journal of Intelligent Systems and Applications, 4(12):41-47. (2012).

Mnih A, Hinton G.: A Scalable Hierarchical Distributed Language Model. NIPS, pages:1081–1088. (2008).

Tang B, Cao H, Wang X, Chen Q and Xu H. Evaluating word representation features in biomedical named entity recognition tasks. Hindawi Publishing Corporation, BioMed Research International, volume 2014. (2014).

Martinez D, Baldwin T.: Word sense disambiguation for event trigger word detection in biomedicine. BMC Bioinformatics, 12(Suppl 2):S4. (2011).

Thomas G. Dietterich.: An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization. Machine Learning, 40:139-157. (2000).

Fonseca E.R, Rosa J.L.G, Alu ́sio S.M.: Evaluating word embeddings and a revised corpus for part-of-speech tagging in Portuguese. Journal of the Brazilian Computer Society, pages:1-14. (2015).

Witten I.H, Frank E.: Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, San Francisco, USA. (2005).

Björne J, Heimonen J, Ginter F, Airola A, Pahikkala T, Salakoski T.: Extracting Complex Biological Events with Rich Graph-based Feature Sets. Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task, 10-18. (2009).

Wang J, Wu Y, Lin H and Yang Z.: Biological Event Trigger Extraction Based on Deep Parsing. Computer Engineering, volume 39. (2013).

Kim JD, Ohta T, Pyysalo S, Kano Y, Tsujii J.: Overview of BioNLP'09 shared task on event extraction[C]. Proceedings of the Workshop on BioNLP: Shared Task, Boulder, Colorado, June 2009:1-9 (2009).

Kim JD, Pyysalo S, Ohta T, Bossy R, Nguyen N, Tsujii J.: Overview of bionlp shared task 2011. Proc BioNLP Shared Task 2011 Workshop, Association for Computational Linguistics.pp. 1–6 (2011).

Turian J, Ratinov L, Bengio Y.: Word representations: a simple and general method for semi-supervised learning. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages. 384–394. (2010).

Crammer K, Dekel O, Keshet J, Shalev-Shwartz S, Singer Y.: Online passive-aggressive algorithms. Journal of Machine Learning Research, pages: 551–585. (2006).

Breiman L.: Bagging predictors. Machine Learning, 26(2):123–140. (1996).

Breiman L.: Out-of-bag estimation. ftp.stat.berkeley.edu/pub/users/breiman/OOBestimation.ps. (1996).

Li L, Fan W, Huang D, Dang Y, Sun J.: Boosting performance of gene mention tagging system by hybrid methods. Journal of biomedical informatics, 45(1):156-164. (2012).

Porter M.F.: An algorithm for suffix stripping. Program electronic library and information systems, 14(3):130–137. (1980).

Miyao Y, Sagae K, Saetre R, Matsuzaki T, Tsujii J.: Evaluating contributions of natural language parsers to protein--protein interaction extraction. Bioinformatics, 25(3):394–400. (2009).

Miwa M, Saetre R, Kim JD, Tsujii J. EVENT EXTRACTION WITH COMPLEX EVENT CLASSIFICATION USING RICH FEATURES. Journal of Bioinformatics and Computational Biology.Vol. 8, No. 1 (2010) 131–146. DOI: 10.1142/S0219720010004586. (2010).

Cristianini N, Shawe-Taylor J.: An instruction to support vector machines: and other kernel-based learing methods. Cambride University Press. (2000).

Domingos P.: A Few Useful Things to Know About Machine Learning. Communications of the ACM, 55(10):78-87. (2012).

Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P, Collins M.: Natural Language Processing (Almost) from Scratch. Journal of Machine Learning Research, 12:2493–2537. (2011).

Mikolov T, Sutskever I, Chen K, Corrado G, Dean J.: Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems, pages:3111–3119. (2013).

Mikolov T, Yih WT, Zweig G.: Linguistic regularities in continuous space word representations. Proceedings of NAACL-HLT, pages:746–751. (2013).

Vapnik VN.: The Nature of Statistical Learning Theory. Springer-Verlag Press, Berlin. (1995).

Vapnik VN.: Statistical Learning Theory. John Wiley & Sons Press, New York. (1998).

Zhang Y, Lin H, Yang Z, Wang J, Li Y.: Biomolecular event trigger detection using neighborhood hash features. Journal of Theoretical Biology, 318:22–28. (2013).

Construction of Domain Ontology

for Engineering Equipment Maintenance Support

Zeng YongHua, Zhuang JianDong, Su ZhengLian

(College of Field Engineering, PLA University Science and Technology，Nan Jing，210000, China)

Abstract：According to the problem in the domain of engineering equipment maintenance, such as more knowledge points, broad scope, complex relationships, difficult in sharing and reuse, this paper put forward the category and professional field of engineering equipment maintain ontology, and analyzed knowledge source, extracted eight core concepts such as case, product, function, damage, environment, phenomena, disposal and resource, and formed concept hierarchy model further, and then analyzed data properties and object properties of core concepts, and tried to construct the engineering equipment maintain ontology with protege4.3, which put a solid foundation for the knowledge base and engineering equipment maintenance application ontology.

Keywords：Domain ontology; Maintain Support; Engineering Equipment

Document Code: A CLC: E919

1 Introduction

With the rapid development of engineering equipment and its maintenance support information construction, the degree of informatization improved continually, the maintenance support knowledge source on engineering equipment increased rapidly. In order to share and reuse the knowledge from different kind and different structure information system, and in order to meet the requirement of the integration of joint security, it is urgent to strengthen management of engineering equipment maintenance knowledge.

Engineering equipment maintenance knowledge involves many disciplines such as mechanical engineering, electrical engineering, cybernetics, behavioral science and diagnosis technology. And there are many store kinds such as audio, video, model, animation, document, table and application software or system, but it has not unified description way， which will lead maintenance personnel to feel it is hard to find the related resources rapidly and precisely, and which also will lead Engineering equipment maintenance knowledge will not be able to applied effectively[1].

Ontology provides the clear, formal and specification explain of shared concept model, which can explain the semantic in an explicit and formal way. Ontology can improve the interoperability of high different structure system, which will lead to knowledge be shared and reused efficiently. So, the construction of engineering equipment maintenance ontology will be benefit of sharing and reusing engineering equipment maintenance knowledge.

Ontology can be divided into top ontology, domain ontology, mission ontology and application ontology. Domain ontology is a professional ontology for special science, which definite the concepts and relationships of concepts, and describe the basic principles, main entities and activity relationships. Domain ontology provides the public understanding foundation, which is thought to be the most promising method to solve the information and knowledge island. Ontology is the concept basis and meta-model of knowledge base. In order to build an engineering equipment maintenance knowledge system successfully, this paper try to build an engineering equipment maintenance ontology preliminary closely in combination with the demand of engineering equipment repair.

2 Overview of Domain Ontology Construction

2.1 Principle

In the long practices of ontology construction, people have advanced many principles. The most influential principle was put forward by Tom Gruber in 1995, which concludes: clarity and objectivity, consistency, extensibility, minimum coding preferences and minimum ontology commitments[2]. Engineering equipment maintenance ontology construction will obtain this principle.

2.2 Tools and Methods

Domain ontology construction is an very onerous and complex system engineering. There are more than 60 building tools, but there is not a standard method. And domain ontology can’t be automatically built, which can only be built by special peoples. We select protege4.3 as building tool and select seven steps as built method.

3 Construction of Equipment Maintenance Domain Ontology

Fully use for reference the soul of ontology construct methodology, we try to construct engineering equipment maintenance support domain ontology, combined with the circular iterative idea of circular obtain methodology, with the step of "seven steps", and with the method of engineering item management on ‘spire prototype method’. Material steps as follows.

3.1 Nail down professional category and domain

First step of domain ontology construction is to nail down the professional category and domain. As we know, engineering equipment maintenance pays most attention on engineering equipment's damage of using phase and related products, situation and repair. Engineering equipment maintenance ontology's user main concludes equipment maintenance support personnel, designer, developer, users and teaching staff in colleges and universities training institutions. The aim of engineering equipment maintenance ontology is to organize the maintenance knowledge with ontology idea and description language, which provides the realization of the knowledge representation[3].

3.2 Comb the resources of domain knowledge

Ontology consists of five elements: concepts, relations, functions, axiom and examples. Concepts can form a classification level, can express the relationship, and can constraint through the relations, functions and axiom. According to the elements of ontology, we get the basic knowledge resources of engineering equipment repair through analysis[4-9].

(1) First resource: authoritative dictionary and encyclopedia. For example, we can get the definition of engineering equipment maintenance, engineering equipment repair technology from 《military encyclopedia》, and we can get the definition of equipment damage, equipment maintain from 《military language》.

(2) Second resource: related domain thesaurus. For example, we can get the concept classification system and knowledge hierarchy relationship such as parts.

(3) Third resource: domain experts. In the view of some unclear concepts and relations, we can ask engineering equipment repair domain experts for confirm.

(4) Fourth resource: standard guidelines. We can get some concepts from the standard guidelines such as repair technical conditions, procedures, and we can analysis the relationship of concepts.

(5) Fifth resource: periodical literature. There are often some repair knowledge in the magazines such as engineering machine and repair, in part due to the strong flexibility of engineering equipment and its repair. So we can get some concept for conference.

(6) Sixth resource: related management information system. We can get some repair case from maintenance query system and maintenance management information system, and then we can construct case model.

3.3 Abstract Core Concepts and Built hierarchy

On the basis of analysis and full collection of domain information, we list all the potential core concepts, and finally we confirmed eight core concepts by the way of identify, analysis and statistics, which include case, product, function, damage, phenomenon, environment, resource and disposal.

Case includes repair case and upkeep case, which mainly record history engineering equipment maintenance knowledge. Repair case record the total process of damage happened and dispose, which include damage description, diagnosis and analysis, fault judge and exclude, repair schedule. It is an important source of equipment engineering maintenance knowledge.

Product is the aim of damage and maintain, and it is the basic object of damage mechanism and repair support countermeasure analysis, and it is also the important object of maintenance support knowledge association and comparability analysis. According to its complexity, we divided product into equipment, system and part.

Function is an abstract description of the specifically ability of product or technology system, which depict the transport and conversion procedure of power stream, matter stream, and information stream, depict the efficacy and ability[10]. We divide function into basic function and assistant function.

Damage is the main cause of repair, different damage need different repair method, different materials, different tools, different repair personnel and different using disposal. We divide damage into battle damage, occasional failure, wear failure, unavailable supply, mis-operation, maladjusted and so on.

Phenomenon is an important factor of fault diagnosis, and it is very important for maintenance decision-making. We divide phenomena into physical phenomenon and biochemistry phenomenon. Physical phenomena include vision, smell, touch and hearing. We also can divide phenomenon into abnormal phenomenon and normal phenomenon.

Environment is an important condition factor of maintenance decision-making. We divided it into geography and threaten, and further we divided element into highland, sea bells, Jack Frost, swamp, desert, woodland and plain, we divided hostility threaten into foreland battlefield and rear battlefield[11].

Resource is also an important factor of maintenance decision-making in battlefield or emergency. We divide resource into technique resource and entity resource, and further divide technique resource into repair tool, repair facility, repair equipment and repair personnel, and further divide technique resource into maintenance guide, upkeep regulation, maintain condition and so on.

Disposal is a settle scheme of damage, which include using disposal and maintain disposal. Using disposal main

include debase use, injured use, change operation mode and hazardous use. Maintain disposal main include upkeep and repair[12].

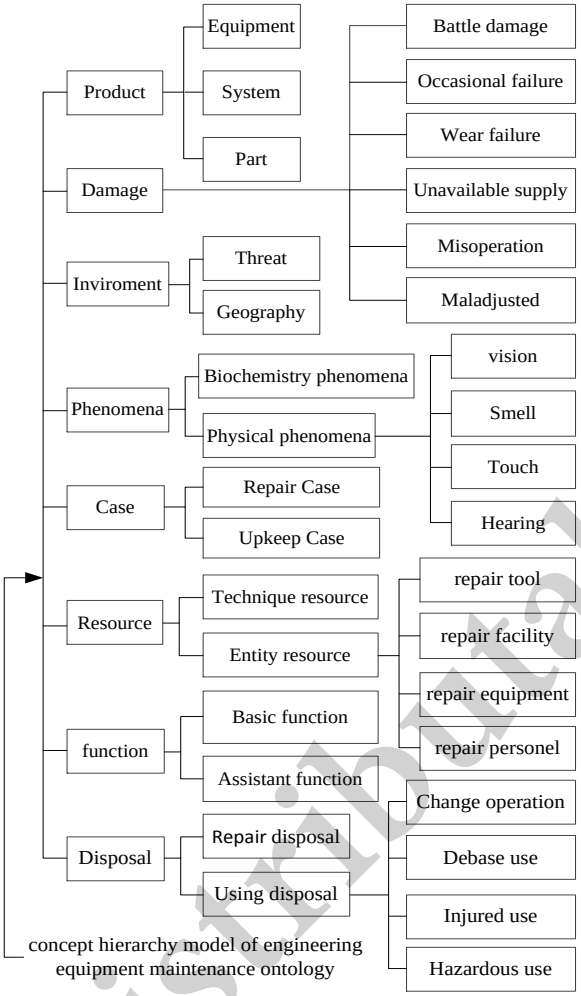


Fig1 concept hierarchy model of engineering equipment maintenance ontology

We expand core concepts, construct the concept model of whole ontology, and then we get the hierarchical model of concept, which is shown as figure 1.

3.4 Concept's Data Property Analysis

The hierarchical model of concept formed the main skeleton body of engineering equipment maintain ontology, but we should expand the concept according to the demand of description to complete the maintain domain ontology construction.

For example, we should use length, width, height, weight and material to describe product, and we will use damage type, damage characteristic, damage mechanism, damage cause, damage degree, damage disposal to describe the concept of damage, and we will use phenomena description and phenomena characteristic to describe phenomena, and we will use case type, case time, case site, case associated product, case associated damage, case description and case evaluate to describe an case. We can get main data properties of engineering equipment maintain ontology by the way of analysis of the description demand of core concepts.

3.5 Concept's Object Property Analysis

According to the application of engineering equipment maintenance knowledge, we mainly analysis the knowledge from the perspective of the products' mechanism, join and dismounting, damage mode and maintain, which is specified as follows.

(1) Structure and Mechanism Relationship Analysis

It is very important to master the knowledge of engineering equipment structure and mechanism, which is basis to carry out maintenance support of engineering equipment. Reference to the FBS model, we think there are main seven

object relationships, which include function hierarchical, function correlation, behavior contain, behavior cause and effect, structure hierarchical, structure generic, structure and function mapping[10].

(2) Join and Discounting Analysis

Replacement repair have been changed into the main repair means of basic-level troops in war. Join and discounting relationship of parts will influence the content and steps of replace, so we should analysis join and discounting relationship of parts on the basis of structure and mechanism. We divide assembling relationship into hierarchical, assort, connect, movement and constraint of assembly, and further we divide connect relationship into clearance fit, excessive cooperate, interference fit, and further we divide constraint into qualitative constraint and quantitative constraint. Fit, alignment, directional and insert relationships consist of qualitative constraint. Angle constraint and distance constraint consist of quantitative constraint.

(3) Analysis of damage and maintain relationship

Damage location is very difficult in maintain of engineering equipment, because damage location associate with many knowledge, such as damage mode, damage mechanism, damage phenomena, damage characteristic, damage type, damage effect, damage dispose and damage case. The mainly relationship include cause and reason, correlation. We divide cause into direct cause and indirect cause, and we also divide cause into initial cause and final cause, we also divide reason into direct reason, indirect reason, final reason. Correlation can also be divided into structure correlation and damage correlation[13].

On the basis of above analysis, we can summarize the relationships, then we get the main relationship include hierarchical relationship, assembling relationship and correlation, which is shown as figure 2.

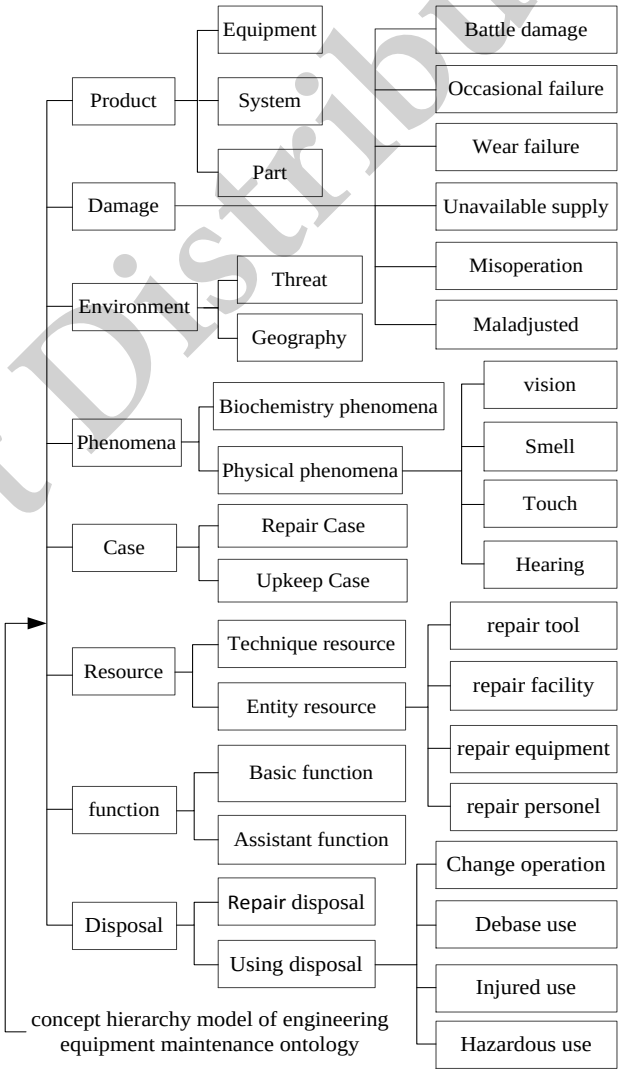


Fig2 Object Property model of engineering equipment maintenance ontology

3.6 Ontology constructed with protege4.3

On the basis of concept hierarchy model, object property model, we tried to built the domain ontology of engineering equipment maintenance support, with protege4.3. First, we built the class according to the concept hierarchy model. Fig 3 showed part of the class hierarchy of engineering equipment maintenance ontology.

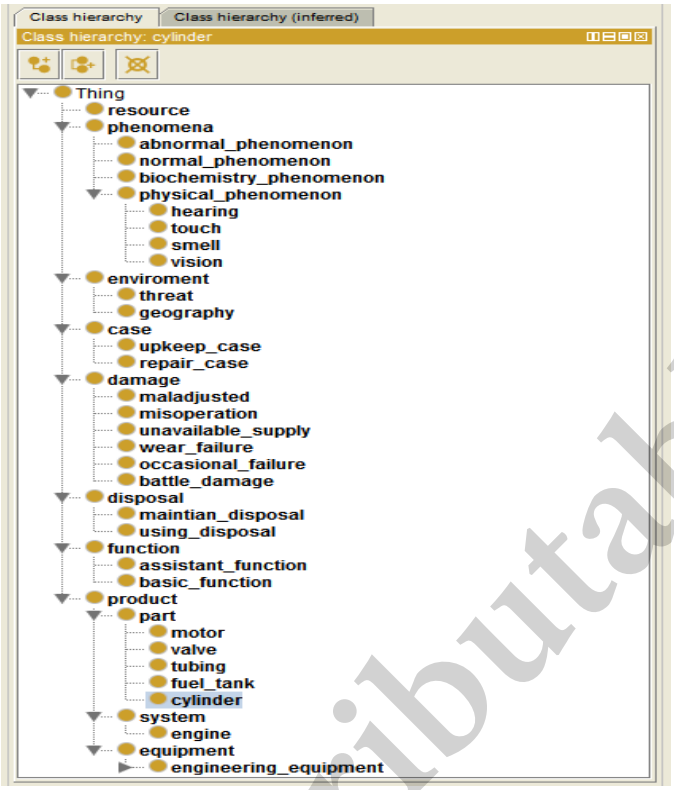


Fig3 Class hierarchy of engineering equipment maintenance ontology

Second, we built object properties according to the object property model. Fig 4 showed part of the class hierarchy of engineering equipment maintenance ontology.

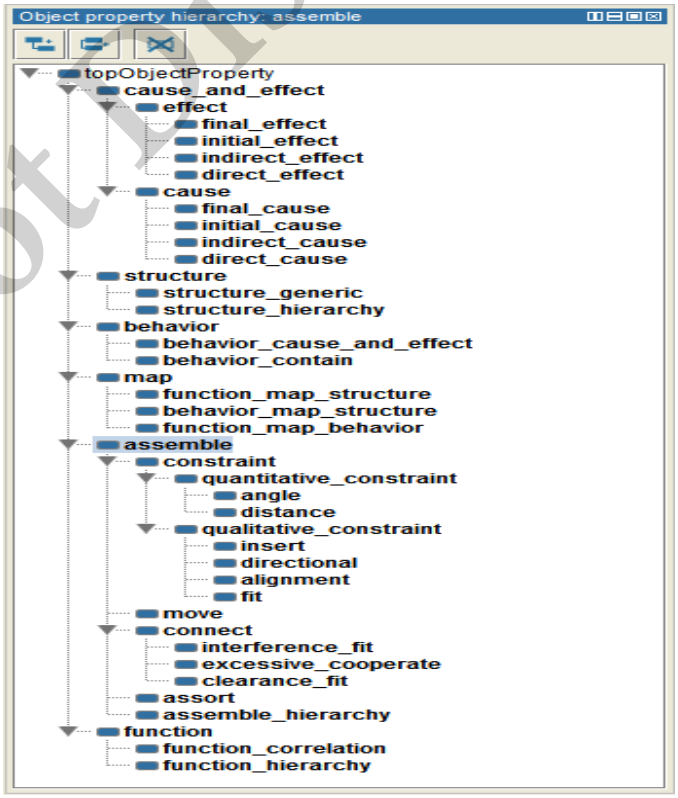


Fig4 Part Object Property of engineering equipment maintenance ontology

Thirdly, we built the data properties on the basis of class. Fig 5 showed part of the data property of engineering equipment maintenance ontology.

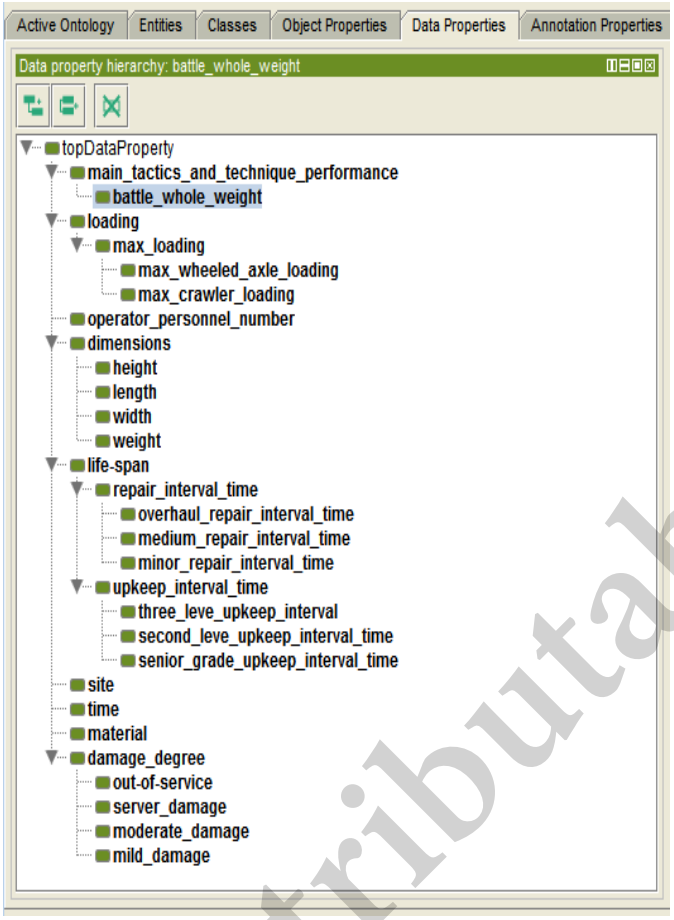


Fig5 Part Data Property of engineering equipment maintenance ontology

At last, we built some individuals of the class, and set data property and object property to the individuals. Then we finished the domain ontology construction preliminary.

3.6 Ontology application

As an engineering equipment maintenance knowledge user, we need to search the interested knowledges. We can input the need into the search textbox of the ontograph plane in protege4.3, and put down the search button, then the below plane will show the related knowledge graph.

For example, if we have built the PLA university of science and technology, the ZL50 loader maintenance teaching book as an individual of resource respectively, and built the ZL50, GJT112 as an individual of engineering equipment respectively, built the M11-C225 as an individual of cylinder, and built the object properties such as write-by, include, and service, and set corresponding object property to such individuals. We can get the knowledge graph from the ontograp plane, which is shown as fig6.

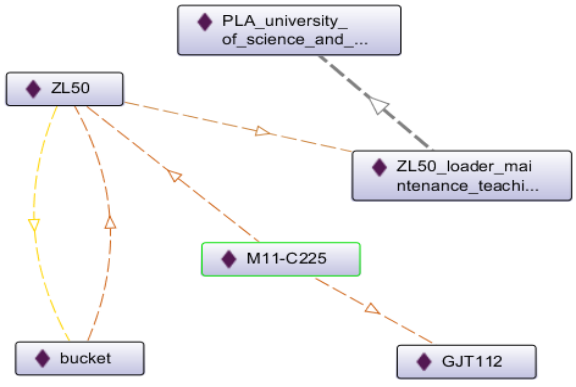


Fig6 Part application of ontology

3.6 Logic detection and evaluate

Engineering equipment ontology was built most by human on the base of tools dictionary such as defence science technology thesaurus. Errors such as logical is easy to happen, so we must use inference engine such as hermit to check, and then we should commit the ontology to domain experts for professionally check.

3.7 Ontology evolution

Ontology construction is an creative design processor. There is not unique method for an professional domain ontology construction, and there are not unique ontology. And domain ontology is developing with the study of domain. So the engineering equipment maintenance ontology will always evolve with all kinds of application ontology development[14-15].

4 Conclusion

Engineering equipment maintenance ontology is an system organization and expression for maintain knowledge of engineering equipment. On the basis of it, we can share and reuse maintenance knowledge, which also can provide found basis for semantic search.

This paper preliminary built engineering equipment maintenance ontology with owl language and protege4.3, which can provide some reference for application ontology built on special version engineering equipment maintenance support and related domain ontology built. And we will further enrich and perfect the ontology with the analysis of specific engineering equipment maintenance knowledge.

Reference

[1] Hu JingQiang, Ji Ya Lin, Meng Yan, Yang Bin, Equipment Support Knowledge Ontology Construction Based on Protege[J]. Modern Electronic Technology, 6(317), 207～210(2010)

[2] Li Kun, Han ZhiQiang, Liu Peng, Yang XiaoBo, The Research and design of military domain ontology Library[J]. Computer knowledge and technology, 6(36), 10196～10198(2010)

[3] Zhou Yang, Li Qing. Ontology modeling and semantic retrieval for aircraft fault knowledge[J]. Computer Engineering and Applications, 47(16), 12～15(2011)

[4] GJB 451A-2005 reliability, maintainability and supportability term[S]. Beijing: the department of General Equipment military standard publisher, (2005)

[5] PLA Military Language[S]. Beijing: military scientific publisher(2011)

[6] Military Engineering Wikipedia dictionary[S]. Beijing: PLA publisher(2003)

[7] Military engineering encyclopedia[M]. Beijing: Weapon industrial publisher(2012)

[8] military keywords dictionary,Beijing: Military Science publisher(1990)

[9] defense science and technology syria word table,Beijing: Military Science publisher(1992)

[10] Ying Hang, Li Shan-ping, Guo Ming, He Sai-long,Research on ontology-based produce knowledge S-B-F representation model[J].Computer integrated manufacturing systems,10, 30～38(2004)

[11] Jiang Wei, Hao WenNing, Yang XiaoJia. Foundation of ontology in military Training Field[J]. Computer Engineering, 34(5), 191～192(2008)

[12] Fan Xiaohui, Shi ChenGuang, Min JianHua, Application and construction of ship gun maintenance[J]. Armory Transaction in Sichuan, 32(8), 120～122(2011)

[13] Liu LingNa, research on the IETM of arms and equipments based on ontology[D], northwestern polytechnical university,53-59(2007)

[14] Su zhenLian, Yan Jun, Chen HaiSong, Zeng YongHua, Construction of ontology-based equipment fault knowledge base[J].system engineering and electronic, 37(9), 2067～2072(2015)

[15] Su zhenLian, Yan jun, Zeng YongHua, Zhang YongQiang, ontology-based equipment support knowledge management mode[J]. Journal of equipment academy,26(4),62～66(2015)

A Mixed Method for Building the Uyghur and Chinese Domain Ontology

Hankiz Yilahun[1], Seyyare Imam[2*] and Askar Hamdulla[1]

1*Institute of Information Science and Engineering, Xinjiang University, China*
2*College of politics and public administration, Xinjiang University, China*
hansumuruh@xju.edu.cn,sayyarim@163.com, askarhamdulla@sina.com

Abstract

As the increasing demands of multilingual semantic query on the World Wide Web, the research on multilingual ontology has gradually become a hot spot. But the study of multilingual ontology on professional field is relatively rare, and a few of the many existing are about the public domain. This paper describes and designs the mixed method for building a new multilingual ontology. By using the above mixed method, construct Uyghur and Chinese bilingual ontology about University management field, through alignment and mapping the concepts and the relations between the different language ontology then merging into one body - multilingual ontology. Finally, preliminary realized semantic query about multilingual ontology using SPARQL, so that will provide basic support for minority languages cross-lingual information retrieval from the perspective of the professional field.

1 Introduction

When the World Wide Web has become the main source of knowledge for people, there are still some problems about low accuracy and low recall rate of information retrieval, even cannot searching any results information. Therefore how to obtain useful knowledge from massive information becomes an urgent problem to be solved. At the same time, the language using by the network is also more and more diverse. For retrieval problem, multi lingual feedback results are more comprehensive than monolingual feedback. Hence, people are no longer satisfied with the retrieval in the one language, instead they require to use a language to retrieve, and the results expressed by a variety of languages. Ontology as a model that can describe the relationship between concepts and concepts at the semantic level, it separates the structure and content of the information, and provides a clear representation of the semantic knowledge. So multi lingual ontology is the key to solve these

* Corresponding Author, sayyarim@163.com

problems (Dai 2008). Multilingual domain ontology is an important resource that to solve some needs about internet information semanticalization and multilingualization, wich has an important role in Multi language technology information service. Its key feature is corresponding concept's consistency in different language ontology. At present, most of the world's cross-lingual ontologies are based on the WordNet or using the same framework of WordNet's structure. For example, EuroWordNet (European word network), RussianWordNet (Russian & English bilingual Ontology), CCD and HowNet (of China Mainland), and The Academia Sinica Bilingual Ontological WordNet (of China Taiwan), etc (Liu 2014).The establishment of these multilingual ontology is a bridge for cross-language information processing. In digital library, the demand of multilingual information retrieval and mining is particularly significant (Zhang 2012). However, In China, multilingual ontologies construction for Uyghur, Mongolian and Tibetan , are still in its initial stage ,in addition to Chinese ,and there are lack of or almost no other language's related research.

China is a unified multi-ethnic country, 53 of the 55 ethnic groups have their own language, which is closely related to the survival and development of the nation. Uyghur language is a mother language of the main ethnic minority (Uyghur) in Xinjiang and the surrounding areas. It is an adhesive language in morphological structure, and belongs to the Altai Turkic languages.

There are vast and numerous classical literature, historical writings and translations in Uyghur language. Whether Uyghur language are as the main carrier of national culture heritage or as the main tool of spreading the knowledge of science and technology culture now, it is inestimable that the unique human culture value and the tremendous role in Xinjiang and its surrounding areas .

2 Related Work

The State Council issued "China's ethnic policy and the national common prosperity and development" in 2009. The white paper pointed out: "in order to make the minority people share in the fruits of the information age, the state has adopted various measures to promote the healthy development of the national minority language and writing standardization and information processing" (Zhao 2011). It has been more than 20 years to study the information processing technology of Uyghur language. Although there are had been made great progress and achieved a lot of results all aspects, but still cannot keep up with the development speed of the information age. If Uyghur language cannot enter the information age, it will lose the basic functions of the language and culture of the carrier, and also will be mercilessly abandoned by this era. Therefore, Uyghur information processing is directly related to the fate of the character, and its significance is self-evident. Because ontology construction is based on the common knowledge that between man and man, man and machine, machine and machine. So, it is increasingly urgent that the construction of the Uyghur ontology in Knowledge Engineering, NLP(Natural Language Processing) and other Artificial Intelligence.

As a preliminary work, In (Hankiz 2015) artificial constructed Uyghur Ontology with protege4.3 about Mathematic and Information Science using domain ontology construction method. This result more comprehensively collected special domain concepts and more accurately described them from a professional point of view. And can say it basically filed the gap about Uyghur Ontology research, provide the basis about cross-language retrieval of Mathematics and Information Science as well. However, the number of concepts and individuals is very small, and the hierarchical relations between concepts are relatively simple, need to further extend and improve. In (Mirsalijan 2015), proposed query expansion technology based on WordNet, that constructed Uyghur semantic dictionary automatically based on WordNet, and did a further query expansion using this dictionary. The method is relatively simple, universal property is good as well, but the noise ratio is relatively large so that cause not very high accuracy.

Sum up the rules of common and unique expressions about different things of each national language, find the similarities and differences between them. Therefore, multilingual Ontology which Unified standard and unified interface will provide an important foundation for the application of multi-national language intelligent information processing, and speed up its implementation.

3 UC Domain Ontology

As a research focus and application purpose of cross-lingual information processing, this paper proposed the necessity of UC(short for Uyghur & Chinese) bilingual domain ontology construction. In order to achieve the above objectives, we starts from construction of UC ontology, and preliminary implement this construction and its semantic query, and lay a solid foundation for the future construction work of Uyghur-Chinese-English-Kazakh-Kirgiz knowledge base.

3.1 Method for Bilingual Ontology Construction

Multilingual ontology construction is divided into three methods. One is building a new ontology from scratch, second is multilingual ontology mapping, and the last is ontology translation or localization. Generally, first method need great workload, the last one has used by many organization already.

1. Construct New One From Scratch

In the absence of source and target language ontology, should learn two language ontology then mapping or translate. Eric Nichols and Francis Bond et al. (2006) had had got multilingual ontology using a variety of Machine Readable Dictionary (MRD). This method extracts single language ontology using English & Japanese dictionary with definition sentences, then alignment the different ontology under lexical layer (Eric 2006).

2. Multi Lingual Ontology Mapping

Mapping between multilingual ontology when there are source and target language ontology here. E.g. as a open resources, WordNet used by many researcher created multilingual ontology through mapping. At present, the research about multilingual ontology focused on language upper ontology. Chinese and English multilingual upper mapping research Including : create bilingual ontology by bilingual alignment using WordNet and HowNet (He 2007).

3. Ontology Translation Or Localization

Translate source to target ontology thus obtain multilingual ontology when there are existing source language ontology and no target here (Zhang 2012)。Because the method is relatively mature, it has been adopted by many organizations in the construction of multilingual ontology, relatively project are EuroWordNet，GlobalWordNet，NeOn etc.

In this paper select the domain of University management and construct the new UC domain ontology by mixed using the 1st and 3rd method. That is , construct source(Chinese) ontology first, then translate and mapping it into target ontology(Uyghur), finally merge them into one . The concrete 3 steps shown in Figure1.

Step1：Because there are so limited text of professional domain in Uygur and rich of the Chinese relatively. So first determine the concepts, hierarchical relations , individuals and properties of concepts from Chinese domain text, then build C-Ontology (short for Chinese Ontology) using Protege4.3, finally store it with OWL(Ontology Web Language) file.

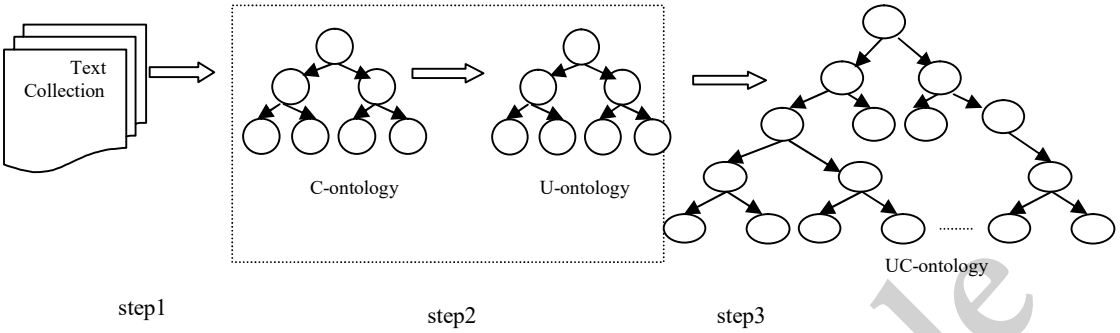


Figure1 Process of UC Domain Ontology Construction

Step2：Then mapping the concepts and relations in C-Ontology into Uyghur concepts and relations. In the process, the lexical ambiguity is determined by the word attribute. ①To the nouns in the C-Ontology, first judge it whether single attribute or no. If it is not a single attribute and with the part of verb attribute from Chinese Uyghur Dictionary, then consider its noun only, and process the verb part as verb attribute. ② To the worlds with other attributes, process them use the same rule.③ In order to improve the accuracy of word matching, need to use the positive and negative matching strategy is needed. If there is one expression of the vocabulary that getting from Chinese Uyghur Dictionary, then use the only one directly. ④ If there are many expressions, then observe its word attribute. If it is single attribute then check up its reverse mapping. That is, if the mapping from Uyghur to Chinese by Uyghur Chinese Dictionary contain the source Chinese word then remain it or cancel it. ⑤ If the Chinese word attribute is not single then go ①. Figure2 explains that, if want to get mapping of CH2 then remain UY1 only, because UY2's reverse mapping not contain CH2, so cancel UY2. Finally can build U-Ontology(short for Uyghur Ontology) matched with C-Ontology using step1 after getting the concepts and properties through mapping.

Step3：Get the UC-Ontology through merging the C-Ontology and U-Ontology with the same domain. The UC-Ontology contain all concepts and relations of the two ontologies and they are matched perfectly each other.

Ontology merging is an effective way of ontology integration, and a kind of method to solve the ontology heterogeneity to realize the reuse and sharing of ontology resources. Ontology merging with same language is divided into two types. That are the merging of ontology with different domain and the merging of with same domain (Liu 2010). This paper considered the same domain ontologies but with different languages. The relatively concepts and relations are can match each other, so realized it using the Import function of Protege4.3. That is importing U-Ontology into C-Ontology then get the new bilingual ontology, and also call them ontology localization. Figure3 is the interface of UC-Ontology with Protege4.3. The relatively concepts matched each other with "same as" relation in the UC-Ontology.

E.g. the instances “武汉大学” and "ۋۇخەن ئونۋېرستى " are with red restriction shown in Figure3.

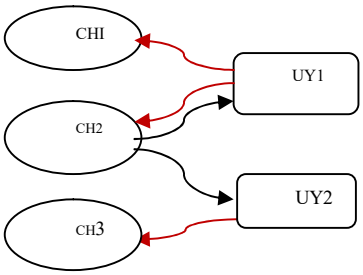


Figure2 positive and negative matching strategy

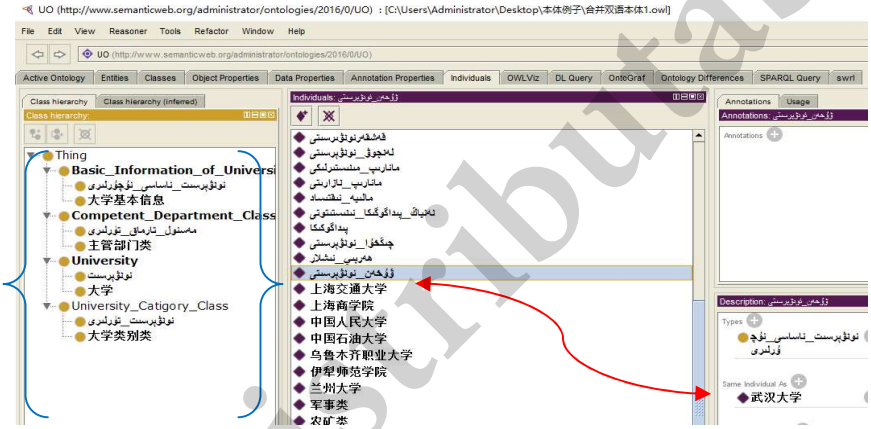


Figure3 UC-Ontology of University Management Domain

3.2 SPARQL Query on UC-Ontology

Jena is the Java framework of construction semantic web application program. It provide the best development environment for the ontology description language that OWL、RDF、RDFS etc. And it has the completely interface for function transfer and processing about ontology parsing, storing, reasoning and searching. Its main framework include the following (Tian 2011)：

（1）RDF API（com. hp. hpl. jena. rdf. model package）.

（2）Ontology Parser： for RDF、RDFS、OWL etc.

（3）RDF Model for persistent storage scheme.

（4）Reasoning Subsystem（com. hp. hpl. jena. rdf. reasoner package）.

（5）Ontology Subsystem：for processing and operating of Ontology.

（6）SPARQL Query Language：for information retrieval.

Now, as a kind of RDF Query Language , SPARQL（Simple Protocol and RDF Query Language standardized by the World Wide Web Consortium. Its importance is similar to SQL's for Relation Database. So it is the first choice of the query language for RDF, OWL etc. (Ji 2011).

Therefore, in this paper select Jena as a the development environment of ontology and SPARQL as a ontology query language.

1. SPARQL Query on Three Tuple

SPARQL is the query language for RDF based on graph merging。Graph Model is three tuple structure model, and forms a basic graph model through these three tuple structures. Basic graph model can be combined into complex graph model. It provides a variety of operators for the connection and merging of graph model. SPARQL also allows quering the three tuple in the ontology model with OWL file. Three tuple is similar to the "subject", "predicate" and "object" in natural language. Three tuple in ontology are <individual, property, value> and <class, property, value> etc. In this paper constructed the ontology with <individual, property, value>. When doing search on ontology, if know the one of the three tuple, then can search out all relatively three tuples. E.g. If we know the individual " **شنجيياڭ ئونۋېرسىتى ： ئۇرۇمچى** "（Xinjiang University）in the ontology, then use the following SPARQL code can easily search out all of three tuple. That are classification of Xinjiang University, competent department, University place, University URL, ranking, grade, school motto, brief introduction, key lab account, and value of "same as", results shown in Figure4.

String search_text = "**شنجيياڭ ئونۋېرسىتى ： ئۇرۇمچى**"

```
Query_string=  "PREFIX
UO:<http://www.semanticweb.org/administrator/ontologies/2016/0/UO#>"
                          +"PREFIX owl:<http://www.w3.org/2002/07/owl#>"
                          +"PREFIX xsd:<http://www.w3.org/2001/XMLSchema#>"
                          +"PREFIX untitled-ontology-
22:<http://www.semanticweb.org/administrator/ontologies/2016/2/untitled-
ontology-22#>"
                           + " Select ?property ?info  "
                          +"Where{"
                          +"{UO:"+search_text+" ?property  ?info .}"
+"}";
```

property	info
UO:مائارىپ_نازارىتى	UO:مەسئۇل_تارماق
UO:"شنجيياڭ ئۇيغۇر ئاپتونوم رايونلۇق مائارىپ نازارىتى"	"مەسئۇل_تارماق_ئىسمى"^^xsd:string
UO:"111"	"ئىسىم_تەرتىبى"^^xsd:int
UO:تۈرى	UO:ئونۋېرسال
UO:"شنجيياڭ ئۇرۇمچى شەھەرلىك تېبانشان رايۇن غالىبىيەت يولى229- نۇمۇر"	"مەسئۇل_تارماق_ئورنى"^^xsd:string
UO:<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	UO:ئونۋېرسىت_ئاساسى_ئۇچۇرلىرى
UO:مەكتەپ_تور_ئادرېسى	"www.xju.edu.cn"^^xsd:string
UO:"ئىستېداتلىق،تەھرىمانە ئىلگىرلەش،ئەمىلىيەتچان،پىكىلىق يارىتىش"	"مەكتەپ_تەلىمى"^^xsd:string
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	owl:NamedIndividual
UO:"1978-يىلى گۈزۈيۈەن تەرىپىدىن شنجياڭ ئۇيغۇر ئاپتونوم رايونلۇق «1997-يىلى 211قۇرۇلۇش» دەپكى نوقتىلىق ئالىي مەكتەپ دەپ بېكىتىلگەن.قىلىپ بېكىتىلگەن	
UO:64"	"نوقتىلىق_تەجرىبىخانا_سانى"^^xsd:int
UO:"ئۆچ يۇلتۇز"	"دەرىجىسى"^^xsd:int
owl:sameAs	UO:untitled-ontology-22:新疆大学
UO:"ئۇرۇمچى شەھرى تېبانشان رايونى غالىبىيەت يولى14-نۇمۇر: "	"ئونۋېرسىت_ئورنى"^^xsd:string

Figure4 SPAPRQL Query Results of UC-Ontology Three tuple

2. SPARQL Query on Class

According to the cross synonym standard of multilingual ontology, the concept should be independent of any language(In there, it at least out of Uyghur and Chinese). Therefore, in this paper, clustered the Uyghur and Chinese classes under the Equivalent English class(shown in with blue restriction in Figure3). Each English parent class has two children which Uyghur and Chinese. The English language used here is a symbol of a class, not English itself。Figure5 showed the results about classes using following SPARQL code. parent class “University” has two children “大学” and “ئونۋېرست”.

```
   Query_string= "PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>"
       +"PREFIX
UO:<http://www.semanticweb.org/administrator/ontologies/2016/0/UO#>"
   + " Select ?subclass ?relation"
   +"Where{"+
   "?subclass  ?relation UO:University  ."
     +"}";
```

subclass	relation
UO:大学	rdfs:subClassOf
UO:ئونۋېرست	rdfs:subClassOf

Figure5 SPARQL query the relation of parent and children

4 Conclusion

Ontology is an explicit formal specification of the domains and relations among them, and its goal is transforming the chaotic information into an orderly knowledge source for easy to use. This paper describes and designs the mixed method that building a new basic ontology from scratch, then get the multilingual ontology with University management domain through translation ,mapping and merging. At last, implement the three tuple and class query of the multilingual ontology using SPARQL. From the perspective of professional domain, it will provide basic support for the cross-lingual information retrieval of minority languages. However, there are some problems, that Uyghur words are not very standard on user interface of Protege and Jena. Because Uyghur is agglutinative language, writing format differ from Chinese and English, and writing it from right to left , so there are some difficulties about the word processing. It is also the one of the next step to study and solve the problems of the paper.

5 Acknowledgments

This work was supported by the National Social Science Foundation of China (13BYY062).

References

Dai Weimin.(2008). Technology and method of Semantic Web Information Organization[M]. Shanghai: Xue lin Press

Liu Yan and Lin Min.(2014). Research on Construction Method of Bilingual Domain Ontology Based on OWL [J]. Computer Technology and Development. 24(8):84-93

Zhang Chengzhi.(2012). Multilingual Domain Ontology Learning Research[M]. Nanjing University Press

Zhao Xiaobin, Qiu Lirong and Zhao Tiejun.(2011). Construction Technology of Ontology Knowledge Base in Minority Languages [J]. Journal of Chinese information Processing. 25(4): 71-74

Hankiz Yilahun, Seyyare Imam and Askar Hamdulla.(2015). A Survey on Uyghur Ontology[J]. International Journal of Database Theory and Application, 8(4):157-168.

Mirsalijan Sawut.(2015). Research on Query Extraction Technology Based on WordNet [D]. Xinjiang University

Eric Nichols, Francis Bond and Takaaki Tanka，et al.(2006). Multilingual Ontology Acquisition from Multiple MRDs[A],In Proceedings of the 2nd Workshop on Ontology Learning and Population(OLP2)[C]. Sydney，Australia:10-17

He Hu and Xiaoyong Du.(2007). Byuilding Bilingual Ontology from WordNet and Chinese Classified Thesaurus[A]. In: Proceedings of the Scholl International Conference on Knowledge Science，Engineering and Management(KSEM2007)[C]. Melbourne，Australia:649-654

Liu Yi.(2010). Imprecise Ontology Merging Research [D]. Dalian Maritime University

Tian hong and Ma Pengyun.(2011). A Reasoning And Query Method For Urban Transportation Domain Ontology Based On Jena [J]. Computer Applications and Software. 28(8):57-60

Ji Zhaohui.(2011). Ontology Searching and Reasoning[J].Journal of Microellectronics and Computer.28(10):52-55

Mining RDF Data for OWL 2 RL Axioms

Yuanyuan Li[1], Huiying Li[1], Jing Shi[1]

[1] School of Computer Science and Engineering, Southeast University,

Nanjing, China

{220141547, 101010166, 220151530}@seu.edu.cn

Abstract: The large amounts of linked data on the web are a valuable resource for the development of semantic applications. However, these applications often meet the challenges posed by flawed or incomplete schema, which would lead to the loss of meaningful facts. Association rule mining, as a successive way to discover implicit knowledge in RDF data, has been applied to learn many types of axioms. In this paper, we first make use of a statistical approach based on the association rule mining to enrich OWL ontologies. Then we propose some improvements according to this approach. Finally, we describe the quality of the automatically acquired axioms by evaluations on DBpedia datasets.

Keyword: Linked Data, RDF, OWL2, Association Rule Mining

1 Introduction

Nowadays, semantic applications are emerging continually which leads to a fast growing number of knowledge repositories on the web. Ontologies are an effective way to improve the quality of linked datasets but many datasets are short of the well-expressive schemas to infer potential information and to validate the consistency of datasets. In our work, we suggest the use of association rule mining methods for discovering ontological knowledge from the linked data itself. Our approach has three characteristics: first is the scalability to work on RDF repositories such as DBpedia, second is the fault-tolerance as we can accept a certain number of incorrect assertions. Third, our approach provides each generated axioms with a certain confidence value for its applications.

The structure of this paper is organized as follows: In Sect.2, we give a brief overview of related works. Afterwards, in Sect.3, we introduce the basics of OWL2 RL and notions of Association Rule Mining. Thereafter, in Sect.4, We describe the methods of getting axioms available in OWL2 RL and propose an improvement for it. In Sect 5, we describe our experiment results learned from two versions of DBpedia dataset. Sect 6 draws conclusions from our work and provides an outlook for future work.

2 Related Works

Several methods have been raised adapting machine learning methods. In [7], the Vector Space Model (VSM) was applied to recognize disjoint classes. However, if the dataset is too huge, a lot of time and memory will be spent storing class vectors, which will lead to low efficiency. Another research work [3] presents ORE which adopt supervised learning methods. As result, only the axioms of $A \equiv C$ and $A \subseteq D$ can be added into the ontology. Other methods mainly discuss the association rule mining. Nebot and Berlanga[2] take advantage of the schema-level knowledge encoded in the ontology to generate transactions which will later satisfy traditional association rules algorithms. Lorey[4] et al compare positive and negative association rules to existing schemas for indicating potential modeling errors.

Particularly related to our approach is the recent work by Johanna Völker et al, who have used association rule mining to learn disjoint axioms. Fleischhacker[5] presents a set of inductive methods to automatically enrich ontologies. One method is correlation computing, in which the correlation coefficients are computed to rate the strength of linear relationships between two classes. However, the

accuracy is not very fine. The other method is negative association rule mining, which takes precision and recall into consideration. But the storage space of the transaction table is too huge. Johanna Völker [8] et al use SPARQL queries to acquire the terminology firstly, construct transaction tables. Finally, mine axioms in the OWL2 EL. Paper [8] is a following work to mine multifarious property axioms.

Although there have some works on learning different types of axioms from linked dataset, few methods are developed to automatically enhance RDF repositories with complete OWL schemas.

3 Preliminaries

The OWL2 is an ontology language providing definitions of classes, properties, individuals, and data values. Several profiles of OWL2 have been described, each of which have different restrictions on the expressivity of OWL2. The OWL2 RL profile is aimed at applications that require scalable reasoning without sacrificing too much expressive power. Its semantics can be defined inductively from a set N_C of concept (or class) names, a set N_R of role (or property) names and a set N_I of individual names. Then the interpretation $I=(\Delta, \bullet^I)$ can be used to represent the actual semantics. The *domain* of I namely Δ^I is a non-empty set containing individuals. The interpretation function $\bullet^I$ maps concept names $A \in \Delta^I$ to a relation $A^I \in \Delta^I$, property names $r \in N_R$ to a binary relation $r^I \in \Delta^I \times \Delta^I$ and individual name $a \in N_I$ to an element $a^I \in \Delta^I$. Table 1 gives the syntax and semantics of OWL2 RL axioms.

The concept of association rules has been widely studied in the area of data mining. A lot of approaches can achieve this algorithm. In our work, we choose the Apriori[1] algorithm.

Table 1. Axioms available in OWL2 RL.

Name	DL Syntax	Semantics
SubClassOf	$C \subseteq D$	$\{ x \in C^I \Rightarrow x \in D^I \}$
EquivalentClasses	$C \equiv D$	$\{ x \in C^I \Rightarrow x \in D^I \wedge x \in D^I \Rightarrow x \in C^I \}$
DisjointClasses	$C \subseteq \neg D$	$\{ x \in C^I \Rightarrow x \notin D^I \}$
SubObjectPropertyOf	$r \subseteq s$	$\{(x, y) \in r^I \Rightarrow (x,y) \in s^I \}$
EquivalentObjectProperties	$r \equiv s$	$\{(x, y) \in r^I \Rightarrow (x, y) \in s^I \wedge (x, y) \in s^I \Rightarrow (x, y) \in r^I \}$
DisjointObjectProperties	$r \subseteq \neg s$	$\{(x, y) \in r^I \Rightarrow (x, y) \notin s^I\}$
ObjectPropertyDomain	$\exists r.T \subseteq C$	$\{(x, y) \in r^I \wedge x \in C^I \}$
ObjectPropertyRange	$\exists r^-.T \subseteq C$	$\{(x, y) \in r^I \wedge y \in C^I \}$
TransitiveObjectProperty	$r \circ r$	$\{(x, y) \in r^I \wedge (y, z) \in r^I \Rightarrow (x, z) \in r^I \}$
InverseObjectPropertyOf	r^-	$\{ (x, y) \in r^I \Rightarrow (y, x) \in r^I \}$
SymmetricObjectProperty	$\mathrm{Sym}(r)$	$\{(x, y) \in r^I \Rightarrow (y, x) \in r^I \}$
AsymmetricObjectProperty	$\mathrm{Asy}(r)$	$\{(x, y) \in r^I \Rightarrow (y, x) \notin r^I \}$
FunctionalObjectProperty	$T \subseteq (\leq 1\ r)$	$\{(x, y) \in r^I \wedge (x, z) \in r^I \Rightarrow y=z \}$
InverseFunctionalObjectProperty	$T \subseteq (\leq 1\ r^-)$	$\{(x, z) \in r^I \wedge (y, z) \in r^I \Rightarrow x=y \}$
IrreflexiveObjectProperty	$\mathrm{Irr}(r)$	$\{\{(x, x) \mid x \in \triangle^I\} \cap r^I = \emptyset \}$
DataPropertyDomain	$\exists R.T \subseteq C$	$\{(x, y) \in R^I \wedge x \in C^I \}$

4 Mining RDF data for OWL2 RL Axioms

In this paper, we use association rule mining approaches to learn OWL axioms from DBpedia datasets. We will first employ SPARQL query language to get expected ontology information. Afterwards, we translate the information into suitable transaction tables. Finally, we execute Apriori algorithm on such transaction databases to discover association rules which can be translated into OWL axioms eventually.

4.1 Transaction Table Construction

We will illustrate the methods of obtaining axioms through an extracted dataset from DBpedia shown in Fig. 1. For convenience, we use http://dbpedia.org/resource/ as default namespace, prefix rdf: for http://www.w3.org/1999/02/22-rdf-syntax-ns# and dbo: for the URI of DBpedia ontology (http://dbpedia.org/ontology). Firstly, we gather information of classes, properties and instances through SPARQL queries. In order to simplify the storage and utilization, we assign each class and instance a unique identifier as it has already done by Niepert[6]. Different types of axioms have different types of transaction tables. For class axioms, relationships of classes are needed. However, it is more complex for property axioms because properties specify how two individuals relate to each other. We will give two representative examples illustrating how to generate transaction table in class and property axioms.

```
Wave_Rock   rdf:type   dbo:PopulatedPlace .     Millennium_Final   dbo:previousEvent   Halloween_Havoc .
Wave_Rock   rdf:type   dbo:Place                WCW_Mayhem         dbo:previousEvent   Millennium_Final .
Machakos    rdf:type   dbo:PopulatedPlace .     WCW_Mayhem         dbo:previousEvent   Halloween_Havoc .
Machakos    rdf:type   dbo:Place .              SupperBrawl        dbo:previousEvent   Souled_Out .
Dominica    rdf:type   dbo:PopulatedPlace .     SupperBrawl        dbo:previousEvent   Starrcade .
Dominica    rdf:type   dbo:Country .            Souled_Out         dbo:previousEvent   Starrcade .
Dominica    rdf:type   dbo:Place .              Arum_alpinum       dbo:genus           Arum .
Doxey       rdf:type   dbo:PopulatedPlace .     Kiang              dbo:genus           Equus_(genus) .
Doxey       rdf:type   dbo:Location .          Kiang              dbo:genus           Asinus .
Doxey       rdf:type   dbo:Place .              Asinus             dbo:genus           Equus_(genus) .
Awre        rdf:type   dbo:PopulatedPlace .     Arum               dbo:genus           Carl_Linnaeus .
Awre        rdf:type   dbo:Place .
```

Fig.1. Triples extracted from DBpedia dataset 2015

The first example is disjoint class axioms. The task of association rule mining is to find rules like $A \Rightarrow \neg B$ and finally translate them into OWL axioms. In transaction tables, each class is labeled with an integer identifiers expressing if one instance belongs to one class (*1* for positive and *0* for negative). What' more, if instance *i* is not declared to be an instance of class *C*, we can have $i \in \neg C$, and the negative identifier will appear in item C and positive for $\neg C$. As a result, the transaction tables can be formed in Table 2. For property axioms, we take transitivity of object property as an example. Transitivity means that if property *r* is transitive and the statements *a r x* and *x r b* exist, and *a r b* must exist too. Hence, the item *r o r* which means *a r x* and *x r b* for an arbitrary instance $x \in N_I$ must be considered. Each transaction in tables represents one possible pair of instances (*a*, *b*) and contains all possible *r o r* and *r*. Thus, the transaction tables for transitivity are generated in Table 3.

Table 2. Transaction tables for class axioms.

URI	PopulatedPlace	Country	Place	Location	¬ PopulatedPlace	¬ Country	¬ Place	¬ Location
Dominica	1	1	1	0	0	0	0	1
Odanad	0	1	1	0	1	0	0	1
Wave_Rock	1	0	1	0	0	1	0	1
Machakos	1	0	1	0	0	1	0	1
Doxey	1	0	1	1	0	1	0	0
Awre	1	0	1	0	0	1	0	1

Table 3. Transaction tables for transitivity axioms.

URIs	previousEvent	genus	genus o genus	previousEvent o previousEvent
(SuperBrawl ,tarrcade)	1	0	0	1
(WCW_Mayhem, Halloween_Havoc)	1	0	0	1
(Asinus,Equus_(genus))	0	1	1	0
(Arum_alpinum, Carl_Linnaeus)	0	0	1	0

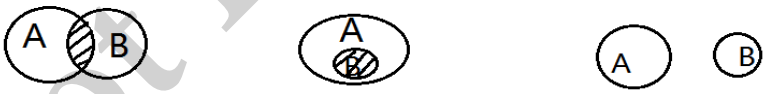
4.2 Class axioms generating

In our experiment, we suppose the confidence threshold to be 0.8. From Table 2, we find the itemset {*PopulatedPlace*, ¬*Country*} reaches a support value of 4. And the confidence value of rule *PopulatedPlace*⇒¬*Country* is 0.8. Likewise, the confidence value of rule *Country*⇒¬*PopulatedPlace* is 0.5. Hence, rule *PopulatedPlace*⇒¬*Country* can be mined, but *Country*⇒¬*PopulatedPlace* cannot. As we all know, the disjointness axioms are symmetrical. In our experiments, we also get 1066 pairs of classes having the form of *A*⇒¬*B* but no form of *B*⇒¬*A*. In addition, the confidence value of rule *Location* ⇒*Place* is 1.0 stating that *Location* is a subclass of *Place*. While rule *Place* ⇒ ¬ *Location* has the confidence value of 0.833. This leads to confliction too. We find 982 such contradictive rule pairs.

From the above analysis, we make a little adjustment to our method. Support($A \cup B$) means the number of instances both *A* and *B* have. In order to guarantee the symmetry, we choose the smaller one of support (*A*) and support(*B*). Three scenarios can be used to verify the rightness of the formula (3).

$$\text{confidence}(A \Rightarrow \neg B) = \text{confidence}(B \Rightarrow \neg A) = 1 - \frac{\text{support}(A \cup B)}{\min\{\text{support}(A), \text{support}(B)\}} \quad (3)$$

The first one is that class *A* and *B* are intersected depicted by Fig. 2(a). We can describe this scenario by example of two classes from DBpedia dataset 2015. Class *Automobile* has 8302 instances, while *MeanOfTransaportation* has 266 instances. They have 116 common instances. Thus, the confidence of *Automobile*⇒ ¬*MeanOfTransaportation* is 0.986, and *MeanOfTransaportation*⇒¬ *Automobile* is 0.563. As a result, asymmetrical axioms are got. But in our method, the confidence values of both are 0.564. They are not disjoint. The second one is depicted by Fig. 2(b). This can be explained by the example of class *Place* owning 10298 instance and *NaturalPlace* with 454 instances. All the instances belonging to *NaturalPlace* also belong to *Place*, which means *NaturalPlace* is the subclass of class *Place*. According to Apriori algorithm, the confidence value of *Place*⇒¬*NaturalPlace* is 0.955. As a result, conflicting rules are got. While in our method, confidence values of these two classes are both 0. Classes are not disjoint at all. The last one is that class *A* has no common instance with class *B* in Fig. 2(c) which can be explained by class *Comics* with 2173 instances and class *Event* with 7585 instances. The overlapped instance number is 0 and the confidence values are 1. So such two classes must be disjoint.



(a) Intersection (b) Inclusion (c) Non-intersection

Fig.2. Relations between two classes

4.3 Property axioms generating

In this section, we present methods about getting property axioms in OWL2 RL. Some properties may have similar restrictions for individuals, we will describe them into groups.

Object Property Transitivity: the representation of transitivity in transaction tables is described in Table 3. From it, the confidence value of rule *previousEvent o previousEvent*⇒ *previousEvent* is 1. And the confidence value of rule *genus o genus*⇒*genus* is 0.5. Thus, we can get the conclusion that object property *previousEvent* is transitive but object property *genus* is not.

Object Property Subsumption and Disjointness: These axioms are similar to the class axioms. Each transaction in the table represents one pair of instances (*a*, *b*) and contains all possible property items when tuple *a r b* holds in dataset. Association rule $r_i \Rightarrow r_j$ is used for subsumption. We extend the

disjointness by adding $\neg r$ into itemset I just as classes. Rule $r_i \Rightarrow \neg r_j$ is for disjointness. In addition, the conflictions happened in class disjointness is also applied to property. Adjustment is applied too.

$$\text{confidence}(r_i \Rightarrow \neg r_j) = \text{confidence}(r_j \Rightarrow \neg r_i) = 1 - \frac{\text{support}(r_i \cup r_j)}{\min\{\text{support}(r_i), \text{support}(r_j)\}}$$

Other properties: We have conducted other property axioms of OWL2 RL just like Fleischhacker[8] have already done. The characteristics are the rest of table 1 except characteristics mentioned above.

5 Experiments

We run our experiment on two DBpedia datasets depicted in Table 4. All experiments have been conducted on a Windows system equipped with an Intel Xeon e3-1225 3.20GHz processor and 16G main memory. Three different confidence thresholds are applied to study the relationships between higher thresholds and the correctness of axioms. We set the support threshold to be 1.

Table 4. Statistical data from different version of DBpedia.

	DBpedia Dataset 3.9	DBpedia Dataset 2015
# of classes	434	677
# of object properties	685	671
# of data properties	689	686
# of instances	8432070	7204698

We mined 14 types of axioms for each dataset. Too many axioms are generated so that it is difficult to check the rightness of these axioms one by one. We randomly chose 50 axioms for each type. If less than 50 axioms, we chose all. The chosen axioms were evaluated by three ontology engineers in the form of a natural language sentence like "The domain of object property *starring* is the class *Film*". They had two choices *right* or *wrong* to evaluate. The accuracy of the learned axioms is computed by averaging the number of correctness from the three engineers. Table 5 gives the results.

Table 5. Evaluation with different confidence thresholds. Number of axioms annotated by #num and accuracy as Acc

	DBpedia Dataset 2015						DBpedia Dataset 3.9					
Axiom	0.8		0.9		1.0		0.8		0.9		1.0	
Type	#num	Acc	#num	Acc	#num	Acc	#num	Acc	#num	Acc	#num	Acc
$C_i \sqsubseteq D_j$	1930	0.95	1857	0.93	1527	0.95	1945	0.93	1941	0.92	1927	0.92
$C_i \sqsubseteq \neg D_j$	485414	0.90	485130	0.89	480671	0.93	185811	0.91	185736	0.93	184539	0.92
$r_i \sqsubseteq r_j$	45	0.96	40	0.95	33	0.94	46	0.89	39	0.91	29	0.93
$r_i \sqsubseteq \neg r_j$	448017	0.93	447868	0.93	445367	0.90	466941	0.92	466796	0.90	464271	0.92
$\exists r.T \sqsubseteq C$	419	0.88	336	0.90	100	0.92	3492	0.88	3368	0.82	2846	0.86
$\exists r^-.T \sqsubseteq C$	71	0.87	44	0.91	21	0.90	598	0.90	313	0.92	112	0.91
$T \sqsubseteq (\leq 1\ r)$	398	0.40	290	0.32	107	0.45	405	0.30	292	0.31	103	0.35
$T \sqsubseteq (\leq 1\ r^-)$	256	0.28	170	0.32	77	0.45	246	0.30	164	0.32	72	0.38
Sym(r)	4	1.0	2	1.0	0	0.0	2	1.0	0	0.0	0	0.0
Asy(r)	652	1.0	640	1.0	488	1.0	672	0.94	659	0.95	505	0.94
$r_i \sqsubseteq r_j^-$	14	0.29	10	0.4	8	0.5	7	0.43	3	1.0	1	0.0
ro $r \sqsubseteq r$	71	0.30	54	0.30	48	0.32	78	0.26	65	0.28	59	0.31
Irr(r)	670	0.98	669	1.0	572	1.0	685	1.0	683	0.98	540	0.97
$\exists R.T \sqsubseteq C$	186	0.88	146	0.90	58	0.92	2000	0.88	1944	0.88	1717	0.91

According to results, we have some observations. It is noticeable that different confidence thresholds have little influence on the accuracy of our results. The accurate percentage tends to be stable in most cases. For the two DBpedia datasets, their numbers of each axiom are very similar except the domain and range axioms. For DBpedia dataset 2015, every property has at most one class as domain or range. While DBpedia dataset 3.9 has more than one class as domain or range and these classes have equivalence and inclusion relationships. The numbers of disjoint relationship of classes and properties are considerable. From DBpedia dataset 2015, there are 677 classes. When arranging them in a two value pair, we will get at least 400, 000 pairs, a lot of which have no common individuals. The same reason is for properties. What's more, low accuracy values for functional and inverse functional axioms come from an argument about the semantics of the properties, such as functional axioms for property color. One engineer thinks things may have at least one color while others think only one color is also ok sometimes. However, in our dataset, property color only occurs with the same subject one time.

6 Conclusion & Outlook

In this paper, we mainly discussed the acquisition of various types of axioms from RDF data. We did experiments on different DBpedia datasets by means of association rule mining. After analyzing the acquired axioms, we found some deficiencies and proposed an improvement. Finally, the learned axioms were evaluated by three ontology engineers. In future, we will take other datasets into consideration such as Wikidata to improve the quality of axioms learning. New approaches should also be proposed to deal with constant updated datasets.

7 Acknowledgements

The work is supported by the Natural Science Foundation of Jiangsu Province under Grant BK20140643 and the National Natural Science Foundation of China under grant No. 61502095.

Reference

1. Agrawal R, Srikant R. Fast algorithms for mining association rules[C]//Proc. 20th int. conf. very large data bases, VLDB. 1994, 1215: 487-499.
2. Nebot V, Berlanga R. Mining Association Rules from Semantic Web Data[C]// Trends in Applied Intelligent Systems -, International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, Iea/aie 2010, Cordoba, Spain, June 1-4, 2010, Proceedings. 2010:504-513.
3. Lehmann J, Bühmann L. ORE - A Tool for Repairing and Enriching Knowledge Bases[C]// The Semantic Web - ISWC 2010 -, International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010, Revised Selected Papers. 2010:177-193.
4. Lorey J, Abedjan Z, Naumann F, et al. RDF Ontology (Re-)Engineering through Large-scale Data Mining[J]. Semantic Web Challenge, 2011.
5. Fleischhacker D, Völker J. Inductive Learning of Disjointness Axioms[C]// Th Confederated International Conference on on the Move To Meaningful Internet Systems. Springer-Verlag, 2011:680-697.
6. Völker J, Niepert M. Statistical Schema Induction[C]// Extended Semantic Web Conference on the Semantic Web: Research and Applications. Springer-Verlag, 2011:124-138.
7. Töpper G, Knuth M, Sack H. DBpedia ontology enrichment for inconsistency detection[C]// International Conference on Semantic Systems. ACM, 2012:33-40.
8. Fleischhacker D, Völker J, Stuckenschmidt H. Mining RDF Data for Property Axioms[M]// On the Move to Meaningful Internet Systems: OTM 2012. Springer Berlin Heidelberg, 2012:718-735.

A Tableau-based Forgetting in ALCQ

Hong Fang[1] and Xiaowang Zhang[2,3,4]

[1] College of Arts and Sciences, Shanghai Polytechnic University, Shanghai 201209, China
[2] School of Computer Science and Technology, Tianjin University, Tianjin 300350, China
[3] Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin 300350, China
[4] Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, Nanjing 211189, China

Abstract. Forgetting is a useful tool for tailoring ontologies by reducing the number of concepts and roles. The issue of forgetting for general ontologies in more expressive description logics, such as $\mathcal{ALCQ}$ and $\mathcal{SHIQ}$, is largely unexplored. In this paper, we develop a decidable, sound, and complete tableau-based algorithm to implement the forgetting-based reasoning. Our tableau algorithm is technically feasibly extended to explore the forgetting in more expressive ontology languages.

1 Introduction

The Semantic Web [1], as an extension of the World Wide Web (WWW), becomes more constantly changing and highly collaborative. Ontologies in Semantic Web can be used by automated tools to provide advanced services such as more accurate web search, intelligent software agents and knowledge management. An example of large biomedical ontology is SNOMED CT. Ontology editing and maintaining tools, such as Protégé, are supported by efficient reasoners based on tableau algorithms for description logics (DLs) [1]. However, as shown in [1], the existing reasoners provide limited reasoning supports for ontology modifications, which largely restricts the wide use of ontologies in the Semantic Web.

Forgetting [3], as an important tool for tailoring ontologies by reducing the number of concepts and roles [3]. It is proven that forgetting can be applied in ontology revision [3], ontology repair [5], and ontology reasoning [6] etc. Though there are some approaches to characterize the forgetting-based reasoning over ontologies [5], it is still interesting to develop some algorithm to characterize the forgetting-based reasoning.

Moreover, it is also interesting to develop some approaches to computing the results of forgetting over ontologies. Recently, there exist some works addressed this issue. For instance, a rewriting approach is presented to compute uniform interpolation in DL-Lite. However, this approach is not direct to treat ontologies in expressive description logics even basic description logic $\mathcal{ALC}$. As an attempt, Wang et al [3] have firstly defined semantic forgetting about concepts and roles in $\mathcal{ALC}$ ontologies and have presented an algorithm to computing the result of forgetting where all concepts are required in disjunctive norm form (DNF). In [4], a tableau-based approach is proposed to compute the results of forgetting over $\mathcal{ALC}$ ontologies where concepts are required in negation normal form (NNF) instead of DNF.

In this paper, inspired from [4], we extend this tableau-based approach to characterize forgetting-based reasoning and generate the rolling-up technique to compute the result of forgetting over ontologies in expressive description logics. This paper focuses the description logic $\mathcal{ALCQ}$ since the number restriction $\mathcal{Q}$ is a most expressive operator in constructing many expressive description logics $\mathcal{SHIQ}$ [2]. Compared with the tableau-based approach introduced in [4], our proposal can further treat ontologies with the number restriction $\mathcal{Q}$.

2 Preliminaries

In this section, we briefly recall some preliminaries of $\mathcal{ALCQ}$ and the tableau algorithm for reasoning tasks. Further details of $\mathcal{ALCQ}$ and the tableau algorithm for $\mathcal{ALCQ}$ can be found in [1,2].

Description logic $\mathcal{ALCQ}$ First, we introduce the syntax of *concept descriptions* for $\mathcal{ALCQ}$. To this end, we assume that N_C is a set of *concept names*, N_R is a set of *role names* and N_I is a set of individuals.

Elementary concept descriptions consist of *concept names* and *role names*. So a concept name is also called *atomic concept* while a role name is also called *atomic role*.

Concepts description in $\mathcal{ALCQ}$ can be formed according to the following syntax:

$$C, D \rightarrow A \mid \top \mid \bot \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \forall R.C \mid \exists R.C \mid \leq nR.C \mid \geq nR.C$$

An interpretation $\mathcal{I}$ of $\mathcal{ALCQ}$ is a pair $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is a non-empty set called the *domain* and $\cdot^{\mathcal{I}}$ is an interpretation function which associates each atomic concept A with a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$ and each role R with a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. This function $\cdot^{\mathcal{I}}$ can be naturally extended to complex descriptions as normal [1]

An *assertional box* (or *ABox*) is a finite set of *assertions*. An assertion is a *concept assertion* of the form $C(a)$ or a *role assertion* of the form $R(a, b)$, where a and b are individuals, C is a concept and R is a role. An interpretation $\mathcal{I}$ *satisfies* a concept assertion $C(a)$ if $a^{\mathcal{I}} \in C^{\mathcal{I}}$, a role assertion $R(a, b)$ if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$. If an assertion ϕ, it is denoted $\mathcal{I} \models \phi$. An interpretation $\mathcal{I}$ is a *model* of an ABox $\mathcal{A}$, denoted by $\mathcal{I} \models \mathcal{A}$, if it satisfied all assertions in $\mathcal{A}$.

An *inclusion axiom* (simply *inclusion*, or *axiom*) is of the form $C \sqsubseteq D$ (C is *subsumed* by D), where C and D are concept descriptions. The inclusion $C \equiv D$ (C is *equivalent* to D) is an abbreviation of two inclusions $C \sqsubseteq D$ and $D \sqsubseteq C$. A *terminology box*, or *TBox*, is a finite set of inclusions. An interpretation $\mathcal{I}$ satisfies an inclusion $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. $\mathcal{I}$ is a *model* of a TBox $\mathcal{T}$, denoted by $\mathcal{I} \models \mathcal{T}$, if $\mathcal{I}$ satisfies every inclusion of $\mathcal{T}$.

Formally, an *ontology* $\mathcal{O}$ is a pair $(\mathcal{T}, \mathcal{A})$ of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. An interpretation $\mathcal{I}$ is a *model* of $\mathcal{O}$ if $\mathcal{I}$ is a model of both $\mathcal{T}$ and $\mathcal{A}$, denoted by $\mathcal{I} \models \mathcal{O}$. If ϕ is an axiom or an assertion, an ontology $\mathcal{O}$ *entails* ϕ, denoted by $\mathcal{O} \models \phi$, if every model of $\mathcal{O}$ is also a model of ϕ. Two ontologies $\mathcal{O}$ and $\mathcal{O}'$ are *equivalent*, denoted by $\mathcal{O} \equiv \mathcal{O}'$, if they have the same models. The equivalent relationship "$\equiv$" can be similarly defined for ABoxes and TBoxes.

The signature of a concept description C, written $sig(C)$, is the set of all concept names and role names in C. Similarly, we can define $sig(\mathcal{A})$ for an ABox $\mathcal{A}$, $sig(\mathcal{T})$ for a TBox $\mathcal{T}$, and $sig(\mathcal{O})$ for an ontology $\mathcal{O}$.

Tableau-based reasoning in $\mathcal{ALCQ}$ The tableau based algorithms have been developed to decide the consistency of general DL ontologies.

Given an ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$, we can assume without loss of generality that all of the concepts occurring in $\mathcal{T}$ and $\mathcal{A}$ are in NNF, i.e., that negation ($\neg$) is always in front of concept names. Note that an arbitrary $\mathcal{ALCQ}$ concept can be transformed into an equivalent one in NNF in polynomial time by applying the following rules:

$$\neg(C \sqcup D) \equiv \neg C \sqcap \neg D, \quad \neg\forall R.C \equiv \exists R.\neg C, \quad \neg \geq nR.C \equiv \leq n-1R.C,$$
$$\neg(C \sqcap D) \equiv \neg C \sqcup \neg D, \quad \neg\exists R.C \equiv \forall R.\neg C, \quad \neg \leq nR.C \equiv \geq n+1R.C.$$

where $\leq (-1)R.C \equiv A \sqcap \neg A$ for some $A \in N_C$. Given a concept C, we use $\doteq C$ to denote the NNF of $\neg C$.

The tableau algorithm works on a data structure called a *completion forest*. This consists a labeled directed graph, each node of which is the root of a *completion tree*. Each node x is labeled a set of concepts $\mathcal{L}(x)$ and each edge $\langle x, y\rangle$ is labeled a set of roles $\mathcal{L}(\langle x, y\rangle)$. If a role $R \in \mathcal{L}(\langle x, y\rangle)$, then we say x is an *R-predecessor* of y (and that y is an *R-successor* of x). A node y is an *ancestor* of a node x if they both belong to the same completion tree and either y is a predecessor of x, or there exists a predecessor z of x such that y is an ancestor of z.

Firstly, the completion forest is initialized $\mathcal{F}$ such that is contains a root node x_a, with $\mathcal{L}(x_a) = \{C \mid a : C \in \mathcal{A}\}$ for each individual name a occurring in $\mathcal{A}$, and an edge $\langle x_a, x_b\rangle$, with $\mathcal{L}(\langle x_a, x_b\rangle) = \{r \mid (a, b) : R \in \mathcal{A}\}$ for each pair (a, b) of individual names for which the set $\{R \mid (a, b) : R \in \mathcal{A}$ is non-empty.

The tableau algorithm applies the expansion rules presented in [2] where $R^{\mathcal{F}}(x, C) = \{y \mid y$ is R-successor of x and $C \in \mathcal{L}(y)\}$. The algorithm stops if it encounters a *clash*: a completion forest in which $\{A, \neg A\} \subseteq \mathcal{L}(x)$ for some node x and some concept name A or if there is some concept $\leq n\, R.C \in \mathcal{L}(x)$ and x has $n+1$ R-successors $y_1, \ldots, y_n$ with $C \in \mathcal{L}(y_i)$ and $y_i \neq y_j$ for all $0 \leq i < j \leq n$. A completion forest is *clash-free* if none of its nodes contains a clash, and it is *closed* otherwise. It is complete if no rule can be applied to it. And the algorithm answers "$\mathcal{O}$ is inconsistent" if the completion forest contains a clash; and it answers "$\mathcal{O}$ is consistent" otherwise.

Note that the tableau algorithm for $\mathcal{ALCQ}$ ABoxes (i.e., TBoxes are empty) would always terminate. However, when the GCIs of TBoxes are discussed in the tableau algorithm, the algorithm might not be terminable. For instance, the algorithm for the GCI $Person \sqsubseteq \exists$*HasParent.Person* runs perpetually. A so-called *blocking* technique is applied to guarantee termination of the expansion process even in the presence of GCIs. A node x is *blocked* if there is an ancestor y of x such that $\mathcal{L}(x) \subseteq \mathcal{L}(y)$ (called "y blocks x"), or if there is an ancestor z of x such that z is blocked; if a node x is blocked and none of its ancestors is blocked, then x is directly blocked.

We introduce a transformation $\sim$ defined as follows: (1) $\sim C(a) = \neg C(a)$; and (2) $\sim C \sqsubseteq D = C \sqcap \neg D(\iota)$ where ι is a special individual which does not occur before.

Lemma 1 *Let $\mathcal{O}$ be an ontology and ϕ a concept assertion or concept inclusion in $\mathcal{ALCQ}$. $\mathcal{O} \models \phi$ iff $\mathcal{F}$ is closed, where $\mathcal{F}$ is a complete forest of $\mathcal{O} \cup \{\sim \phi\}$ by applying the tableau algorithm.*

3 Forgetting in $\mathcal{ALCQ}$

In this section, following from forgetting $\mathcal{ALCQ}$ ontology presented in [3], we will simply give a semantic definition of what is means to forget about a set of variables in an $\mathcal{ALCQ}$ ontology.

As explained earlier, given an ontology $\mathcal{O}$ on signature $\mathcal{S}$ and $\mathcal{V} \subset \mathcal{S}$, in ontology engineering it is often desirable to obtain a new ontology $\mathcal{O}'$ on $\mathcal{S} - \mathcal{V}$ such that reasoning tasks on $\mathcal{S} - \mathcal{V}$ are still preserved in $\mathcal{O}'$. As a result, $\mathcal{O}'$ is weaker than $\mathcal{O}$ in general. This intuition is formalized in the following definition.

Definition 1 *Let $\mathcal{O}$ be an ontology in $\mathcal{ALCQ}$ and $\mathcal{V}$ a set of variables. An ontology $\mathcal{O}'$ over the signature $sig(\mathcal{O}) - \mathcal{V}$ is a result of forgetting about $\mathcal{V}$ in $\mathcal{O}$ if*

F1 *$\mathcal{O} \models \mathcal{O}'$;*
F2 *for each concept inclusion $C \sqsubseteq D$ in $\mathcal{ALCQ}$ not containing any variables in $\mathcal{V}$, $\mathcal{O} \models C \sqsubseteq D$ implies $\mathcal{O}' \models C \sqsubseteq D$;*
F3 *for each member assertion $C(a)$ or $R(a, b)$ in $\mathcal{ALCQ}$ not containing any variables in $\mathcal{V}$, $\mathcal{O} \models C(a)$ implies $\mathcal{O}' \models C(a)$ (resp., $\mathcal{O} \models R(a, b)$ implies $\mathcal{O}' \models R(a, b)$).*

If the result of forgetting about $\mathcal{V}$ in $\mathcal{O}$ is expressible as an $\mathcal{ALCQ}$ ontology, we say $\mathcal{V}$ is *forgettable* from $\mathcal{O}$.

Proposition 1 *Let $\mathcal{O}$ be an ontology in $\mathcal{ALCQ}$ and $\mathcal{V}$ a set of variables. If both $\mathcal{O}'$ and $\mathcal{O}''$ in $\mathcal{ALCQ}$ are resulting of forgetting about $\mathcal{V}$ in $\mathcal{O}$, then $\mathcal{O}' \equiv \mathcal{O}''$.*

This proposition says that the result of forgetting in $\mathcal{ALCQ}$ is unique up to ontology equivalence. Given this result, we write $\mathrm{forget}(\mathcal{O}, \mathcal{V})$ to denote any result of forgetting about $\mathcal{V}$ in $\mathcal{O}$ in $\mathcal{ALCQ}$. In particular, $\mathrm{forget}(\mathcal{O}, \mathcal{V}) = \mathcal{O}'$ means that $\mathcal{O}'$ is a result of forgetting about $\mathcal{V}$ in $\mathcal{O}$.

If the result of forgetting about $\mathcal{V}$ in $\mathcal{O}$ is expressible as an $\mathcal{ALCQ}$ ontology, $\mathcal{V}$ is called *forgettable* from $\mathcal{O}$.

The following property states that the definition of the result of forgetting $\mathcal{ALCQ}$ ontology is appropriate.

Proposition 2 *Let $\mathcal{O}$ be an ontology and $\mathcal{V}$ a set of variables in $\mathcal{ALCQ}$. If both $\mathcal{O}'$ and $\mathcal{O}''$ are the result of forgetting about $\mathcal{V}$ in $\mathcal{O}$, then $\mathcal{O}' \equiv \mathcal{O}''$.*

Forgetting in TBoxes is independent of ABoxes as the next result shows.

Proposition 3 *Let $\mathcal{T}$ be a TBox in $\mathcal{ALCQ}$ and $\mathcal{V}$ a set of variables. Then, for any ABox $\mathcal{A}$ in $\mathcal{ALCQ}$, $\mathcal{T}'$ is the TBox of $\mathrm{forget}((\mathcal{T}, \mathcal{A}), \mathcal{V})$ iff $\mathcal{T}'$ is the TBox of $\mathrm{forget}((\mathcal{T}, \emptyset), \mathcal{V})$.*

Proposition 4 *Let $\mathcal{O}$ be an ontology in $\mathcal{ALCQ}$ and $\mathcal{V}$ a set of variables. Then*

1. *$\mathcal{O}$ is consistent iff* $\mathrm{forget}(\mathcal{O}, \mathcal{V})$ *is consistent;*
2. *for any inclusion or assertion ϕ not containing variables in $\mathcal{V}$, $\mathcal{O} \models \phi$ iff* $\mathrm{forget}(\mathcal{O}, \mathcal{V}) \models \phi$.

This proposition shows that two major reasoning tasks, namely, consistency and query answering, can be preserved in the definition of forgetting. From the property, such two reasoning tasks in an ontology can be reduced into those tasks in the result of forgetting in the ontology. In this sense, we take advantage of forgetting to optimize reasoning tasks.

The following proposition shows that the forgetting operation can be divided into steps, with a part of the signature forgotten in each step.

Proposition 5 *Let $\mathcal{O}$ be an ontology in $\mathcal{ALCQ}$ and $\mathcal{V}_1, \mathcal{V}_2$ two sets of variables. Then we have* $\mathrm{forget}(\mathcal{K}, \mathcal{V}_1 \cup \mathcal{V}_2) \equiv \mathrm{forget}(\mathrm{forget}(\mathcal{K}, \mathcal{V}_1), \mathcal{V}_2)$.

For simplicity, forgetting in ontologies is independent of order of forgetting. Based on this idea, to compute the result of forgetting about $\mathcal{V}$ in $\mathcal{K}$, it is equivalent to forget in variables in $\mathcal{V}$ one by one.

4 Tableau-based forgetting in $\mathcal{ALCQ}$

In this section, we will compute the resulting of forgetting some variables based on the completion forest which is obtained by applying the tableau algorithm for $\mathcal{ALCQ}$.

Given an ontology $\mathcal{O}$ and a set of variables $\mathcal{V}$, the completion forest $\mathcal{F}$ which is obtained by applying the tableau algorithm may still contain some variables in $\mathcal{V}$. For instance, let $\mathcal{O} = (\{A \sqsubseteq B\}, \{A(a)\})$ and the completion forest $\mathcal{F}$ which is obtained by applying the tableau algorithm w.r.t. concept name A contains two branches $\mathcal{B}_1 = \{\mathcal{L}(a)\}$ where $\mathcal{L}(a) = \{A, \neg A\}$ and $\mathcal{B}_2 = \{\mathcal{L}(a)\}$ where $\mathcal{L}(a) = \{A, B\}$. However A still occur in $\mathcal{F}$. That is to say, in the completion forest $\mathcal{F}$, all variables forgotten are not deleted but ignored only. However, the result of forgetting does not contain any variable forgotten. Thus, to compute the result of forgetting from the $\mathcal{F}$, those variables forgotten are necessary to be deleted from $\mathcal{F}$. Since $\mathcal{F}$ are two different forms of the same result, we consider compute the result of forgetting based on $\mathcal{F}$ in this paper. In the following, we will delete variables in the completion forest by considering both nodes $\mathcal{L}(x)$ and edges $\mathcal{L}(\langle x, y\rangle)$ to a completion forest irrelevant to the variable set $\mathcal{V}$.

Definition 2 (Forgetting forest) *Let $\mathcal{O}$ be an ontology and $\mathcal{V}$ a set of variables. $\mathcal{F}$ is a completion forest by applying the tableau algorithm w.r.t. $\mathcal{V}$ on $\mathcal{O}$. We say the result of forgetting $\mathcal{V}$ in $\mathcal{F}$, written by* $\mathrm{forget}(\mathcal{F}, \mathcal{V})$*, is a forest obtained by forgetting nodes (written* $\mathrm{forget}(\mathcal{L}(x), \mathcal{V})$*) and forgetting edges (written* $\mathrm{forget}(\mathcal{L}(\langle x, y\rangle), \mathcal{V})$*) defined as follows:*

– *for every node $\mathcal{L}(x)$,* $\mathrm{forget}(\mathcal{L}(x), \mathcal{V})$ *is obtained from $\mathcal{L}(x)$ by*
  Step 1 *delete all the form $C \sqcup D$ or $C \sqcap D$ or $\exists R.C$ or $\geq nR.C$;*
  Step 2 *if $\{A, \neg A\} \subseteq \mathcal{L}(x)$ with $A \in \mathcal{V}$, then replace A and $\neg A$ by $\bot$;*
  Step 3 *if A or $\neg A$ or $A \sqcup C$ or $\neg A \sqcup C$ in $\mathcal{L}(x)$ with $A \in \mathcal{V}$, then delete A or $\neg A$ or $A \sqcup C$ or $\neg A \sqcup C$;*

Step 4 *if* $\forall R.C \in \mathcal{L}(x)$ *or* $\leq nR.C \in \mathcal{L}(x)$ *with* $R \in \mathcal{V}$*, then delete* $\forall R.C$ *or* $\leq nR.C$*;*

Step 5 *if* $\forall R.C \in \mathcal{L}(x)$ *or* $\leq nR.C \in \mathcal{L}(x)$ *with* $R \notin \mathcal{V}$*, then replace* C *with* $\mathrm{forget}(\{C\}, \mathcal{V})$ *and delete* $\forall R.(\top \sqcup C)$ *or* $\leq n\ R.C$*;*

– *for every edge* $\mathcal{L}(\langle x, y\rangle)$*,* $\mathrm{forget}(\mathcal{L}(\langle x, y\rangle), \mathcal{V})$ *is obtained from* $\mathcal{L}(\langle x, y\rangle)$ *by if* $R \in \mathcal{L}(\langle x, y\rangle)$ *with* $R \in \mathcal{V}$*, then* $\mathcal{L}(\langle x, y\rangle) - \{R\}$*.*

Note that (1) $\mathrm{forget}(\mathcal{L}(x), \mathcal{V})$ is recursive; and (2) $\mathrm{forget}(\mathcal{F}, \mathcal{V})$ is irrelevant to $\mathcal{V}$.

As will be readily seen, the forgetting forest algorithm w.r.t. nodes $\mathcal{L}(x)$ in completion forest $\mathcal{F}$ is similar to the algorithm of *compute C-forgetting* presented in [3]. It is quite natural that when we only consider each node $\mathcal{L}(x)$, the node $\mathcal{L}(x)$ can be taken as DNF of a *complex concept*. For instance, a node $\mathcal{L}(x) = \{A_1, A_2, \forall R.A_3\}$ can taken the DNF of the complex concept $C = A_1 \sqcap A_2 \sqcap \forall R.A_3$. We will apply the mechanism to compute the result of forgetting later. Forgetting forest algorithm w.r.t. edges $\mathcal{L}(\langle x, y\rangle)$ is directly deleting the roles in set of variables $\mathcal{V}$ from $\mathcal{L}(\langle x, y\rangle)$.

In fact, the forgetting forest algorithm holds the equivalence as follows.

Theorem 1 *Let* $\mathcal{O}$ *be an ontology and* ϕ *an axiom in* $\mathcal{ALCQ}$*. For any set of variables* $\mathcal{V}$ *irrelevant to* ϕ*, we have* $\mathrm{forget}(\mathcal{O}, \mathcal{V}) \models \phi$ *iff* $\mathrm{forget}(\mathcal{F}, \mathcal{V})$ *is closed. where* $\mathcal{F}$ *is a completion forest of* $\mathcal{O} \cup \{\sim \phi\}$ *by applying the tableau algorithm.*

Given an ontology $\mathcal{O}$ and a set of variables $\mathcal{V}$, Theorem 1 shows that the forest which does not contain any variable in $\mathcal{V}$ obtained by applying the forgetting forest algorithm could capture the consistency of $\mathcal{O}$ limited in the set of variables $sig(\mathcal{O} - \mathcal{V})$.

Acknowledgments

This work is supported by the program of Applied Mathematics Discipline of Shanghai Polytechnic University (XXKPY1604) and and the open funding project of Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education.

References

1. Baader F., Calvanese D., McGuinness D.L., Nardi D., & Patel-Schneider P.F. (2003). The description logic handbook: theory, implementation, and applications. Cambridge University Press.
2. Horrocks, I. & Sattler, U. (2004). Decidability of SHIQ with complex role inclusion axioms. *Artif. Intell.*, 160(1-2):79–104.
3. Wang, Z., Wang, K., Topor, R., & Pan, Z.J. (2010). Forgetting for knowledge bases in DL-Lite. *Ann. Math. Artif. Intell.*, 58(1-2):117–151.
4. Wang, Z., Wang, K., Topor, R., & Zhang, X.(2010). Tableau-based forgetting in ALC ontologies. In: Proc. of ECAI'10, pp. 47–52.
5. Zhang, X.(2016). Forgetting for distance-based reasoning and repair in DL-Lite. *Knowl.-Based Syst.*, 107:246-260, 2016.
6. Zhang, X., Wang, K., Wang, Z., Ma, Y., Qi, G., & Feng, Z. (2016). A distance-based framework for inconsistency-tolerant reasoning and inconsistency measurement in DL-Lite. *Int. J. Approx. Reason.*, `http://dx.doi.org/10.1016/j.ijar.2016.08.003`, 2016.

E-SKB: A Semantic Knowledge Base for Emergency

Chang Wen, Yu Liu, Jinguang Gu, Jing Chen, and Yingping Zhang

College of Computer Science and Technology,
Wuhan University of Science and Technology

Abstract. Although the number of knowledge bases in Linked Open Data has grown explosively, there are few knowledge bases about emergency, an important issue in the area of social management. In this paper, we introduce a semantic knowledge base of emergency, extracted from an authoritative website. According to the characteristics of the website, a framework is suggested to convert web into RDF. In order to help researchers acquire more knowledge, we follow the publishing rules of Linked Open Data—not only using URIs to label the objects in the semantic knowledge base, but also providing links to DBpedia. Finally, we employ Sesame to store and publish the semantic knowledge base, and develop a query interface to retrieve the knowledge base with SPARQL.

Keywords: Emergency, Linked Open Data, Semantic Knowledge Base, SPARQL

1 Introduction

Emergency, an unexpected event, may cause serious social harm and bring a great loss to human life [1]. It can be divided into four categories, natural disasters, accidents disasters, public health and social security [2]. Due to the uncertainty and paroxysm of emergency, it is necessary for us to integrate the scattered data into a knowledge base, which contributes to collect information efficiently and conveniently.

As one of the widely used technologies, the semantic knowledge base is suitable for developing an application to deal with emergencies. Some popular knowledge bases were constructed successfully such as GeoNames [3], DBpedia [4], FOAF [5], etc. If the knowledge base about emergency has collected a great amount of events in detail, it would be easy to work out potential results and feasible solutions by searching the knowledge base when an emergency happens. The E-SKB can be adopted to construct an expert system to handle emergencies, improve the query accuracy and realize the linguistic diversity through linking with DBpedia.

Since the most of knowledge bases in Linked Open Data (LOD) [6] do not cover the specific knowledge about emergency, we extract the web information from a case database of emergency management, maintained by Jinan University, which has collected 574 cases and 2275 resources of emergency (`http://decm.`

jnu.edu.cn/) and can be accessed by the browser. Then the web information is converted into RDF triples by following the principles of LOD, so that researchers can retrieve the knowledge with semantic web technologies, such as SPARQL.

2 Construction Process of E-SKB

In order to construct the E-SKB, the main processing procedures can be divided into three parts. First of all, we introduce a crawler that is applied to collect data from the web. Secondly, we extract the concepts according to the classification tree, then define the properties by the labels on the news pages, and link the data set to DBpedia following the LOD rules. Finally, we employ the Sesame to store the data and develop a query interface to acquire E-SKB by SPARQL.

2.1 Extracting Data from Web

Given there are a wide variety of methods to develop a crawler, we just give a brief description of the crawler that is used by JSOUP and HttpClient.

Due to a large quantity of URLs need to be solved, the technologies of queue and multithreading are applied to the crawler. As a result, the crawling process can be functioned more efficiently. While an URL is added into the queue, we collect the news information with HTML filter and convert it to a JSON string. Then the first URL will be removed and we repeat the previous step until the queue is empty. Finally, the emergency information is presented in the JSON format.

2.2 Processing Data with Certain Rules

The Linked Data is a group of best practices for publishing and interlinking structured data on the web. It was introduced by Tim Berners-Lee in his website [7] and has become known as the Linked Data principles, which can be concluded as follows:

- Using URIs to present things.
- Using HTTP URIs, so that people have access to resources.
- When someone looks up the URI, information is found by SPARQL query language.
- URIs are linked with each other, helping users discover more resources.

In the process of dealing with the data set, the principles mentioned above should be obeyed in order that we can keep the data normatively.

Concepts Extraction. Concepts are utilized to describe a set of entities, which possess the same types and can be linked to the existed ones on the Internet. The emergencies are classified into several kinds based on the classified layer tree on the web. Fig. 1 shows some event classes and the relations between them, the

"Accident Disaster" can be regarded as a concept that has nine sub classes, each one of them is a unique concept as well. All the entities belong to the nine sub classes are parts of the "Accident Disaster". The main relation of these concepts is presented by the property "subClassOf" in RDFS, which means one concept is a subset of another.

According to the Linked Data principles, each concept has a unique URI so that there is no confliction in defining the emergencies. We construct the concept's URI by adding the class name behind the namespace. The class names are extracted from the classified layer tree in the home page, and the namespace is defined as "`http://decm.jnu.edu.cn/class#`". Finally, we can build a concept model to show the taxonomic hierarchies between different emergencies.

Fig. 1. Some event classes and the relaions between them in E-SKB

Properties Extraction. The properties are relations between the subject resources and object resources, which can be deemed to the predicates in the sentences. The properties are divided into two groups, the system properties and user defined properties. System properties are the internal properties of the RDF and RDFs, which have XML Schema data type vlaues. User defined properties are the attributes defined to present the specific relations. For each defined properties, we need to assign its domain and range to indicate the subject and object.

The definition of property is the same as the concept, which includes namespace and property names. The namespace is defined as "`http://decm.jnu.edu.cn/property#`". Since most of the pages are constituted in a uniform way, all of them include the same labels to present the emergency contents. Fig. 2 shows an emergency case entitled "Spraying pesticide poisoned 9 people". The seven

labels are nation, area, location, start time, end time, loss and relevant resources. We regard these labels as the property names and define a "content" to present the description. So the "content" is expressed as "`http://decm.jnu.edu.cn/property#content`", which is abbreviated to "depr:content".

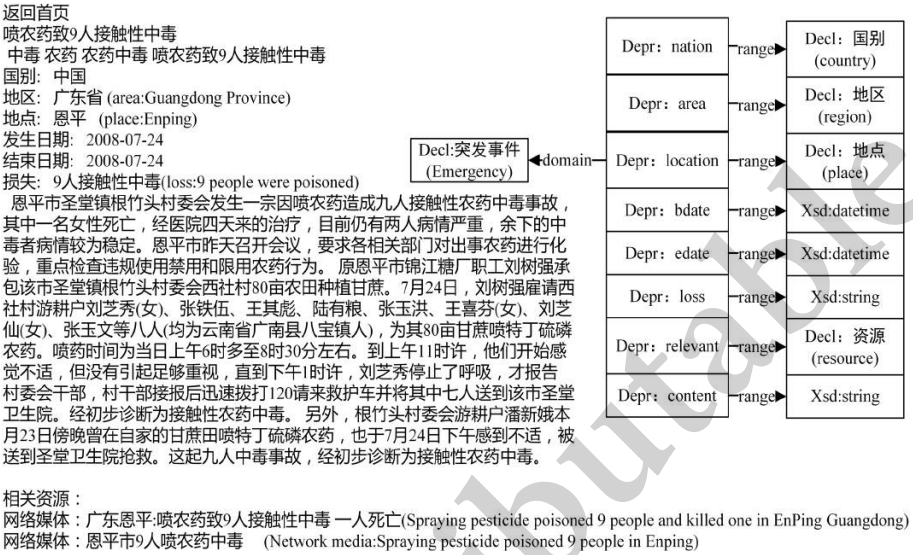


Fig. 2. The properties of an event instance

Instances Extraction. According to the definitions of concepts and properties, we extract the instances from the web. They can be divided into two types, emergency news and related resources. The former is the news that is described in the page, and the latter is related news about the topic. The relation between two instances is represented as the property "depr:relevant" as we have mentioned above.

Take the news of "Spraying pesticide poisoned 9 people" for example, the knowledge graph of the instance is shown in Fig. 3. Resources are connected with the instance by properties: therefore they constitute triples that can be formatted into RDF.

2.3 Linking E-SKB to DBpedia

Linked Data is the core technology in exposing, sharing and connecting web information, which uses RDF and URI to present things and the relations between resources. The characteristics of LOD include simple structures, standardized information and low-cost interaction between the mankind and the machine. In this paper, the geographical concepts can be associated with the resources in the DBpedia for the sake of data sharing.

As mentioned before, the instances have the properties of nation, area and location, and the property values are the resources of the specific information. We can connect these geographical values with the resources that have the same meanings in DBpedia. Since the values in E-SKB are Chinese, we need to find the corresponding resources and use "owl:sameAs" to link them together. The steps to link E-SKB to DBpedia are as follows:

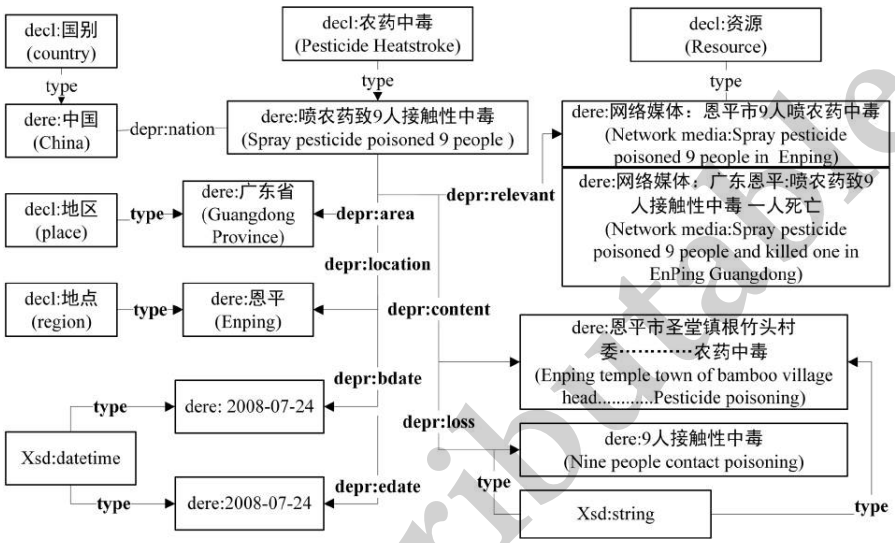


Fig. 3. The knowledge graph of an event instance

1. Constructing the geographical resources according to the format in DBpedia. The prefix of resources is defined as "`http://dbpedia.org/resource/`", so we can add the Chinese geographical names after the prefix to construct the resources.
2. Querying the related resources by using the SPARQL endpoint in DBpedia. We use the resources built in step one as the objects and the "`http://dbpedia.org/ontology/wikiPageRedirects`" as the predicate to construct the query statements. The related resources in DBpedia such as "`http://dbpedia.org/resource/Beijing`" will be returned.
3. Linking the geographical resources to DBpedia. We use "`http://www.w3.org/2002/07/owl#sameAs`" as the property to link E-SKB to the resources returned in step two.

According to the steps discussed above, we can get 891 linking results of the geographical resources in E-SKB. In the future work, we will expand the E-SKB by extracting other news websites and linking more resources to DBpedia.

3 Publishing E-SKB into Sesame

We store the data as RDF triples and publish it into the Sesame server. The Sesame files are downloaded and deployed to the tomcat server. Finally, the RDF file is uploaded to the Sesame server, which can be accessed by the query interface.

4 Web-based Query System

In the web-based query system, we can get the detailed information of E-SKB by SPARQL. Fig. 4 shows the result of querying instance "Spraying pesticide poisoned 9 people", the properties and objects are returned.

http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://decm.jnu.edu.cn/class#农药中毒
http://decm.jnu.edu.cn/property#Relations	http://decm.jnu.edu.cn/class#网络媒体：广东恩平:喷农药致9人接触性中毒 一人死亡
http://decm.jnu.edu.cn/property#Areas	http://decm.jnu.edu.cn/class#广东省
http://decm.jnu.edu.cn/property#StartTime	http://decm.jnu.edu.cn/property#2008-07-24
http://decm.jnu.edu.cn/property#Relations	http://decm.jnu.edu.cn/class#网络媒体：恩平市9人喷农药中毒
http://decm.jnu.edu.cn/property#Damages	http://decm.jnu.edu.cn/property#9人接触性中毒
http://decm.jnu.edu.cn/property#Countrys	http://decm.jnu.edu.cn/class#中国
http://decm.jnu.edu.cn/property#EndTime	http://decm.jnu.edu.cn/property#2008-07-24

Fig. 4. Query results of an event instance

Acknowledgments. This work was partly supported by the National Science Foundation of China (No. 61502359), the National Students' Innovative Entrepreneurship Training Program under Grant (No. 201510488016).

References

1. Xue Lan, Zhong Kaibin. The Category, Classification and Periodization of Emergency Events: the Based Management System of emergency. Administrative Management of China, 102-107(2005)
2. An Yu. The theoretical framework of Emergency Response Law. Law Science Magazine. 27(4):28-31(2006)
3. Yoshioka M, Kando N. Issues for Linking Geographical Open Data of GeoNames and Wikipedia. Semantic Technology(2013)
4. Auer S, Bizer C, Kobilarov G, et al. DBpedia: A Nucleus for a Web of Open Data(2010)
5. Dan B. L.: FOAF Vocabulary Specification 0.9. Computer Science & Communications Dictionary, 23(3):165(2007)
6. Heath T, Bizer C. Linked Data: Evolving the Web into a Global Data Space. Molecular Ecology, 22(3):670684(2011)
7. Tim Berners-Lee. Linked Open Data Design Issues. `http://www.w3.org/DesignIssues/LinkedData.html`(2006)

An Initial Ingredient Analysis of Drugs Approved by China Food and Drug Administration

Haodi Li, Qingcai Chen, Buzhou Tang[1], Dong Huang,
Xiaolong Wang, Zengjian Liu

Key Laboratory of Network Oriented Intelligent Computation, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China 518055
{haodili.hit, qingcai.chen, tangbuzhou, donghuang2012}@gmail.com,
wangxl@insun.hit.edu.cn, liuzengjian.hit@gmail.com

Abstract. Drug is an important part of medicine. Drug knowledge bases that organize and manage drugs have attracted considerable attention, and have been widely used in human health care in many countries and regions. There are also a large number of electronic drug knowledge bases publicly available. In China, however, there is hardly any publicly available well-structured drug knowledge base, may due to two different types of medicine: Chinese traditional medicine (CTM) and modern medicine (ME). In order to build an electronic knowledge base of drugs approved by China Food and Drug Administration (CFDA), we developed a preliminary ingredient drug analysis system. This system collects all drug names from the website of CFDA, obtains their manuals from three medical websites, extracts the ingredients of drugs, and analyses the distribution of the extracted ingredients. Totally, 12,918 out of 19,490 drug manuals were collected. Evaluation on randomly selected 50 drug manuals shows that the system achieves an F-score of 95.46% on ingredient extraction. According to the distribution of the extraction ingredients, we find that ingredient multiplexing is very common in medicine, especially in herbal medicine, which may provide a clue for drug safety as taking more than one type of drug that contains partially the same ingredients may cause overtaking the same ingredients.

Keywords: drug knowledge base, Chinese traditional medicine, drug ingredient extraction

1 Introduction

In human's history, medicine always attracts considerable attention. Until now, it has made great progress and various types of medicine with different types of drugs appear such as Chinese traditional medicine (CTM) and modern medicine (ME). In a country or region, there may be more than one type of medicine. For example, in China, CTM and ME coexist. Most drugs in ME consist of only one chemical substance, while most drugs in CTM consists of multiple medicinal herbs. The elementary units of drugs in ME are different from that of drugs in CTM. For drugs in ME, there have been a large number of public electronic knowledge bases in the United States of America (USA), which have been widely used in human health care. However, few electronic knowledge bases of drugs in other types of medicine such as CTM are available.

[1] Corresponding authors

In order to build a well-structured electronic knowledge base of drugs approved by China FDA (CFDA), we collect all drug names from the website of CFDA (http://www.sda.gov.cn), obtain their manuals from some medical websites, and analyse them briefly. Among these drugs, manuals of 12,918 drugs are collected from medical websites. In order to analyse the drugs, we build an automatic ingredient extraction system based on manuals of 320 randomly selected out of the 12,918 drugs. Evaluation on manuals of the other randomly selected 50 drugs shows that the ingredient extraction system achieves a precision of 96.51%, a recall of 94.44% and an F-score of 95.46%. With this system, all ingredients are extracted from the 12,918 manuals. Based on the extracted ingredients, we find that the ingredient multiplexing is very common in medicine, especially in herbal medicine.

2 Related Work

A large number of drug knowledge bases have been developed for different applications such as medication information exchange, clinical decision support, etc. In the USA, both government departments and academically institutions have been involved in building and maintaining various types of drug knowledge bases. The representative drug knowledge bases include the FDA Terminology, NDF-RT[1], RxNorm[2], DrugBank[3], medical databases in UMLS[4] and so on. The FDA Terminology is developed by US FDA and used to support medication information exchange between government agencies by using the Unique Ingredient Identifier (UNII) codes that uniquely identify all ingredients of marketed drugs in the USA to control terminology in medication information area. NDF-RT is a drug database made and maintained by the Veterans Health Administration (VHA). RxNorm provides normalized names for clinical drugs and links them to many drug vocabularies and databases. DrugBank is a database of FDA-approved drugs, nutraceuticals and experimental drugs. UMLS is developed and maintained by the US National Library of Medicine (NLM). It indexes and links various dictionaries through a simple semantic network. All these drug knowledge bases are digitized and most of them are publicly available.

In China, the related research on drug knowledge base construction started later. The early studies mainly focused on how to interpret each term of drug dictionaries. For example, the Chinese Pharmacopoeia edited by the National Pharmacopoeia Committee of China uses the active ingredients of drugs as the basic units to describe the drugs' chemical structure, properties, detect methods and so on [5]. The Dictionary of Chinese Pharmacy uses drug ingredients as the basic units to describe drugs' aliases and comments [6]. The Contemporary Drug's Names and Tradenames Dictionary edited by the China Association of Traditional Chinese Medicine also uses the active ingredients of drugs as the basic units to describe the drugs' category, relative diseases, aliases and production name [7]. In recent years, some researchers have begun to use semantic relations to construct drug knowledge bases such as the Traditional Chinese Medicine Language System developed by the China Association of Traditional Chinese Medicine [8]. Most of drug knowledge bases only focus on

drugs in herbal medicine, and there is hardly any publicly available electronic drug knowledge base.

3 Method

Figure 1 shows the overview of our preliminary drug ingredient analysis system. It consists of five components as follows:

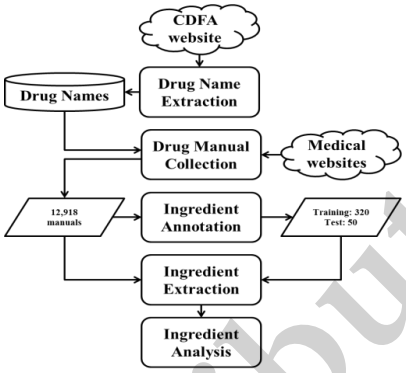


Figure 1. Overview of our preliminary drug ingredient analysis system.

(1) Drug Name Extraction: extract drug names from the CDFA website (http://www.sda.gov.cn) by a customized crawler. 19,490 drugs have been approved by CFDA in total until January 2015, and have been classified into seven categories: herbal medicine (9,914), chemical medicine (8,879), accessory (30), biologicals (555), pharmaceutic adjuvant (6) and other (106).

(2) Drug Manual Collection: collect drug manuals from three medical websites, i.e., http://ypk.39.net, http://www.yaopinnet.com and http://db.yaozh.com. We collect all manuals in text form, and finally obtain 16,882 manuals.

(3) Ingredient Annotation: randomly select 370 drug manuals for annotation. Among them, 320 manuals are used as a training set, and the reminding 50 manuals are used as a test set.

(4) Ingredient Extraction: extract ingredients of drugs from their manuals. This task is recognized as a sequence labeling problem, and Conditional Random Fields (CRF) is used to solve it. The first step of ingredient extraction is to split every manual into sentences. After sentence split and tokenization, each ingredient is represented by BILO tags, where B, I, L and O denote a Chinese character at the beginning, in the middle, at the ending and outside of an ingredient respectively. An example of the ingredient representation is shown in Figure 2. A CRF model is trained on the training set, and all collected drug manuals are labeled by the model. The features used in the CRF-based system only include N-grams of tokens (N=1, 2, 3 in a window of [-3, 3]), segmentation and part-of-speech. Precision, recall and F-score are used to measure the performance of the ingredient extraction system.

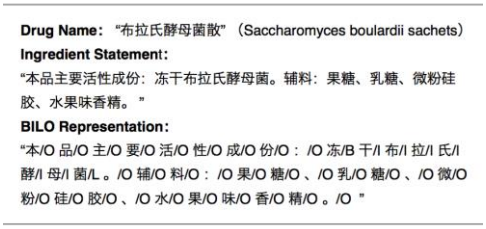


Figure 2. Example of the ingredient representation.

(5) Ingredient Analysis: analyse the distribution of ingredients in the drugs approved by CFDA according to the results of the ingredient extraction module.

4 Result

The precision, recall, F-score of our ingredient extraction system on the test set are 96.51%, 94.44% and 95.46% respectively. On the 25 drug manuals in herbal medicine, the ingredient extraction system achieves a precision of 96.47%, a recall of 95.00% and an F-score of 95.71% respectively, while it achieves a precision of 96.88%, a recall of 91.18% and an F-score of 93.94% on the 25 drug manuals in chemical medicine. Obviously, the ingredient extraction system shows better performance in herbal medicine than chemical medicine.

On all 12,918 drug manuals in text format, the ingredient extraction system obtains 5,107 types of ingredients, including 3,420 types of herbal ingredients and 2,102 types of chemical ingredients. To further understand the distribution of the ingredients, we list the most common 5 ingredients of drugs in herbal medicine in Table 1 and the most common 5 ingredients of drugs in chemical medicine in Table 2 respectively. The most common ingredient of drugs in herbal medicine is liquorice ("甘草" in Chinese), which occurs in 1,659 drugs, and the most common ingredient of drugs in chemical medicine is acetaminophen ("对乙酰氨基酚" in Chinese), which occurs in 158 drugs. It seems that ingredient multiplexing in herbal medicine is more common than chemical medicine. To validate it, we further investigate the relationship between the number of drugs and the number of ingredients as shown in Figure 3, where x axis is the number of ingredients sorted by the count they occur in drugs and y axis is the number of drugs containing the corresponding ingredient. It is clear that the ingredient multiplexing in herbal medicine is more common than chemical medicine.

Table 1. Most common 5 ingredients of drugs in herbal medicine

Ingredients		**Counts**
Name	Chinese name	
Liquorice	甘草	1716
Angelica sinensis	当归	1556
Astragalus membranaceus	黄芪	1081
Poria cocos	茯苓	1069
Ligusticum chuanxiong hort.	川穹	927

Table 2. Most common 5 ingredients of drugs in chemical medicine

Ingredients		Counts
Name	Chinese name	
Acetaminophen	对乙酰氨基酚	158
Chlorpheniramine maleate	马来酸氯苯那敏	151
Vitamin B	维生素 B	145
Glycerin	甘油	121
Sodium chloride	氯化钠	116

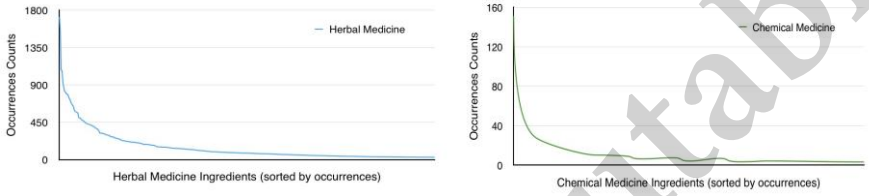


Figure 3. Relationship between the number of drugs and the number of ingredients.

5 Discussions and Conclusion

In this study, we analyse the distributions of ingredients of drugs approved by CFDA, where the ingredients are extracted by a CRF-based classifier. As the CRF-based ingredient extraction system achieves an F-score of 95.46% on an independent test set, the analysis would be worthy of trust.

We notice that a number of drug manuals (6,572 out of 19,490) cannot be collected from the three medical websites. Most of them are not available on the internet, and a small number of them are only available in non-plain text format. Therefore, drug information needs to be further digitized. In our future work, we will manually add the missed drug manuals to our database.

Although the manuals of drugs are well formatted, it is not easy to extract ingredients from them by simple rules. At the beginning of this study, we have ever attempted to extract ingredients from the first sentences in the "ingredients" field of drug manuals by splitting the sentences by punctuations and treating each part as an ingredient. However, this rule-based method achieves only a precision of 84.68%, a recall of 85.04% and an F-score of 84.86% on the test set, which are much worse than the CRF-based classifier. The main challenge lies in that the ingredients of some drugs are not given directly.

There are some interesting findings from the extracted ingredients. Firstly, two drugs may have the same ingredients such as "JuBanZhiKe Granule" and "JuHong Pill" ("橘半止咳颗粒" and "橘红丸" in Chinese), both of which consist of 14 herbs. Secondly, one drug may consist of a subset of ingredients of another drug. For example, "ShaYao" ("痧药" in Chinese), a drug in herbal medicine, is composed of 11 herbs, and another drug in herbal medicine "ChanSuDing" ("蟾酥锭" in Chinese) is

composed of 4 out of the 11 herbs of “ShaYao”. These two drugs look similar according to their ingredients but their indications are greatly different.

This study is a preliminary step of other studies such as medical knowledge graph construction, but it is can be widely used in several medical applications. It may guide the suitable usage of drugs. For example, drugs that contain the same ingredients had better be taken separately as overdosing one ingredient may cause potential side effects such as polygonum multiflorum[9] (“何首乌” in Chinese).

Ingredient multiplexing is very common in medicine, especially in herbal medicine. Based on the results of the drug ingredient extraction system, we can further link drugs through their common ingredient(s), which is a part of knowledge graph of drugs and is one case of our future work.

Acknowledgments. This paper is supported in part by grants: National 863 Program of China (2015AA015405), NSFCs (National Natural Science Foundation of China) (61402128, 61473101, 61173075 and 61272383) and Strategic Emerging Industry Development Special Funds of Shenzhen (JCYJ20140508161040764 , JCYJ20140417172417105 and JCYJ20140627163809422).

References

[1] S.H. Brown, P.L. Elkin, S.T. Rosenbloom, C. Husser, B.A. Bauer, M.J. Lincoln, J. Carter, M. Erlbaum, M.S. Tuttle, VA National Drug File Reference Terminology: a cross-institutional content coverage study, Medinfo, 11 (2004) 477–481.
[2] S. Liu, W. Ma, R. Moore, V. Ganesan, S. Nelson, RxNorm: prescription for electronic drug information exchange, IT professional, 7 (2005) 17–23.
[3] V. Law, C. Knox, Y. Djoumbou, T. Jewison, A.C. Guo, Y. Liu, A. Maciejewski, D. Arndt, M. Wilson, V. Neveu, others, DrugBank 4.0: shedding new light on drug metabolism, Nucleic acids research, 42 (2014) D1091–D1097.
[4] O. Bodenreider, The unified medical language system (UMLS): integrating biomedical terminology, Nucleic acids research, 32 (2004) D267–D270.
[5] C.P. Commission, others, Chinese pharmacopoeia, Chemical Industry Press, Beijing, 328 (2005) 547.
[6] Guan Xie, Zhōngyī Dàcídiǎn, Beijing: People's Health Publisher, (1998).
[7] Zhigang Zhao,Contemporal Drug's Names and tradenames dictionary, Chemical Industry Press , (2006).
[8] Ai-ning Yi, Nuen Zhang,A Stduy on Unified Traditioanl Chinese Medicine Language System,Chinese Journal Of Information On Traditional Chinese Medicine, 10 (2003) 90–92.
[9] X. Lei, J. Chen, J. Ren, Y. Li, J. Zhai, W. Mu, L. Zhang, W. Zheng, G. Tian, H. Shang, Liver Damage Associated with Polygonum multiflorum Thunb.: A Systematic Review of Case Reports and Case Series, Evidence-Based Complementary and Alternative Medicine: eCAM, 2015 (2015) 459-749.

Position Paper: The Unreliability of Language - A Common Issue for Knowledge Engineering and Buddhism

Zhangquan Zhou[1] and Guilin Qi[1]

[1]School of Computer Science and Engineering, Southeast University, China,
{quanzz,gqi}@seu.edu.cn

Abstract. According to the studies of Kurt Gödel and Ludwig Wittgenstein, both of formal languages and human languages are unreliable. This finding inherently influences the development of artificial intelligence and knowledge engineering. On the other hand, their finding, i.e., the unreliability of languages, was early discussed by Gautama Buddha who founded Buddhism. In this paper, we discuss the issue of the unreliability of language by bridging the perspectives of Gödel, Wittgenstein and Gautama. Based on the discussion, we further give some philosophical thoughts from the perspective of knowledge engineering.

Keywords: knowledge engineering, artificial intelligence, language, unreliability, Gödel, Wittgenstein, Buddhism

1 Introduction

The core of *knowledge engineering* is to apply different kinds of formal languages (or models) to represent and manage human languages (or knowledge) [6]. Researchers in the filed of knowledge engineering develop and optimize methods for automatic knowledge management, and even for making knowledge machine-understandable, which is also one target of *artificial intelligence*. A question then arises naturally: "*Is it possible that all kinds of human knowledge can be represented and handled by machines?*" Unfortunately, the answer turns out to be "NO" from a theoretical perspective. The reason can be traced back to Gödel's incompleteness theorems [3], which state that there is not a complete and reliable system for proving all mathematical consequences. This result was further extended to general formal languages by Alfred Tarski [7]. Since all current methods of representing and handling human knowledge are based on formal languages or models, they submit to Gödel's incompleteness theorems.

From a philosophical perspective, even human languages are unreliable, i.e., they are full of contradictions and mistakes. This was claimed by Ludwig Wittgenstein, whose theories essentially laid the foundations of the *linguistic philosophy*. The findings of Gödel and Wittgenstein inherently influence the development of artificial intelligence and knowledge engineering. However, their finding, i.e., the unreliability of languages, was early discussed by Gautama Buddha who founded

Buddhism. Gautama said that we cannot rely on languages to understand the truth of the world. In summary, researchers in the filed of knowledge engineering have to face an issue: *(formal and human) languages are unreliable.*

The aim of this paper is not to address the above issue, i.e., the unreliability of languages, but to highlight this issue by bridging the perspectives of Gödel, Wittgenstein and Gautama. Based on the discussion of their views, we give some philosophical thoughts from the perspective of knowledge engineering.

2 The Incompleteness of Mathematical Languages

In 1931, Kurt Gödel published his incompleteness theorems (known as Gödel's incompleteness theorems) [3], which are important in both of the mathematical logic and the philosophy mathematics. As shown in the name of Gödel's incompleteness theorems, the famous theorems indicate an important property of mathematical languages: *incompleteness*. That is, one cannot prove all mathematical consequences by using the axioms expressed in mathematical languages[1]. We give a simple case of incompleteness in the following example.

Example 1. Consider a mathematical system δ where set *is the unique atomic element represented by capital letters. It is allowed that a set can be a member of other sets. The symbols "$:=, \{, |, \}, \notin$" and "$\in$" are also allowed in δ. These symbols have standard mathematical semantics and can help us to describe the relations between sets ($\notin$ or $\in$) and define new sets. We now define a set X by $X := \{S | S \notin S\}$ where S is a set.*

The mathematical system δ is such simple that it contains sets as its unique elements. However, δ is incomplete. Consider the question "*Is X a member of itself?*" (formally $X \in X$?). If $X \in X$, then it contradicts the definition given in Example 1. Thus X should not be in X; if $X \notin X$, then X satisfies the above definition and should be in X. From the standard semantics of $\in$ and $\notin$, any two sets should have a binary relation of either $\in$ or $\notin$. However, both of the two results ($X \in X$ and $X \notin X$) would result in contradictions. The problem in Example 1 is known as *Russell's paradox* (or more popular, *Barber paradox*) that was proposed by Bertrand Russell in 1903. One intuitive cause of this kind of problems is *self-reference* [1], i.e., X is also defined by itself.

Russell's paradox indirectly resulted in *the Third Mathematical Crisis* when the completeness of mathematical languages was under suspicion. In fact, to find a complete and perfect mathematical system was the dream of several famous mathematicians, like David Hilbert, who devoted a large part of his life to this work. However, Kurt Gödel finally broke the dream with a simple fact: mathematical languages are unreliable due to incompleteness.

[1] Alfred Tarski extended the results to more general formal system five years later. We refer the readers to Tarski's undefinability theorem [7].

3 Language-game

Ten years before the publication of the incompleteness theorems, a young man, in his doctoral thesis [8], claimed that *even human languages are unreliable*.

The young man is Ludwig Wittgenstein, who was the protege of Bertrand Russell, and the classmate of Adolf Hitler. In his doctoral thesis [8], Wittgenstein analyzed the *contradictions*, *vagueness* and *woven* of human languages. The finding of Wittgenstein really resulted in the crisis of philosophy, and built a new branch of philosophy: the *linguistic philosophy*. The importance of linguistic philosophy lays in that, it fundamentally queries all the other schools of western philosophy. This is because that, all philosophical theories are described by human languages. Since human languages are unreliable and full of contradictions, philosophical theories cannot hold even from the level of language. To understand this, we give the following example.

Example 2. A Martian asked Wittgenstein a question: "Sir, how many toes do philosophers have?" Wittgenstein answered:"Of course ten!" The Martian raised his feet of only six toes and said sadly:"Does that mean, we Martians cannot be philosophers?"

In daily life, we use a large amount of *concepts* (or *terms*) in our languages to communicate with each other, like the concept "*philosopher*" in the above example. However, we rarely doubt the exact meaning of these used concepts (more precisely, their *intensions* and *extensions*). This is due to two reasons: (1) We tend to easily believe what we see and hear, which would further be reflected in our languages; (2) The meanings of concepts in our languages are established by people who live in the same environment around us. Recall Example 2. In our common sense, a philosopher should first be a human. Thus, we undoubtedly treat "*having ten toes*" as one of the extensions of the concept "philosopher".

Wittgenstein argued that, philosophical theories cannot be built on the concepts without exact definitions and specifications. For example, when one asks the question "*Who am I? Why am I in this world?*", he should first give the exact definitions of the concepts "*I*", "*world*" and the semantics of the interrogatives, "*who*" and "*why*". He also found that, many concepts were even defined by themselves (this is like the case of self-reference mentioned in Section 2). In this sense, using languages is similar to playing games: given a bunch of words without meanings, we first set the rules of how to use these words, and then we use these words to communicate, to describe our ideas and react to the word usage of other people. This is also called *language-game* by Wittgenstein [8]. From his theories, language-game is such a process when the meanings of words are not static, but are dynamically changed according to different situations and different people. He also said that language-game is being played in every family where children learn to use languages from their parents.

The idea of language-game gives a negative signal: we can never find the truth of the world and ourselves through languages, since we are just being in a game where the rules of how to use languages are full of mistakes and contradictions. This finding brought Wittgenstein a huge suffering at the end of his life.

4 The Influence of Gödel's Incompleteness Theorems to Computer Science and Artificial Intelligence

Different from the suffering of Wittgenstein, Gödel's incompleteness theorems actually benefited scientists a lot. The incompleteness theorems and the contributions of many mathematicians for solving the Third Mathematical Crisis virtually gave birth to the strongest tool in human history, *computer*.

At the same time of Gödel, there was an American mathematician called Alonzo Church. He and Gödel contributed a lot to *recursion theory*. Driven by the similar dream of Hilbert, they started their journey to a different destination: to build a universal machine that can describe and solve mathematical problems. However, a Ph.D student of Church was the first one to reach the destination. The name of this student is Alan Turing. The universal machine described by Turing is also well known as *Universal Turing machine* [2], which is supposed to be the prototype of computer.

The prototype of computer further encouraged scientists to investigate whether a computer can solve all problems with termination. By referring to Gödel's incompleteness theorems, scientists immediately found the answer: "NO". It is proved that there exists a large group of problems that cannot be solved with termination [2]. These problems are also called *uncomputable problems*. The uncomputability can also be ascribed to the issue of self-reference (see the related content in Section 2). Many uncomputable problems are proved using the technique called *diagonalization*[2], which is essentially a formulation of self-reference.

Gödel's incompleteness theorems also influenced the development of artificial intelligence (AI for short). On one hand, the *symbolic logic* as the classical approach of AI suffers that, computers cannot solve some problems when using highly expressive logic languages with termination. With regard to techniques, the notion of self-reference is always used to identify the completeness of a logic language, e.g., introducing *canonical models* to identify completeness for *model theoretic logic*. On the other hand, many researchers pay more attention on *statistical models* rather than *symbolic logics*. The basic idea of statistical techniques is to make machines behave like men by *leaning* human behavior. The related techniques are known as *statistical learning* or *machine learning*. However, for both of symbolic logics and statistical models, researchers just choose to weaken the influence of the unreliability of formal languages or models, but not completely solve it.

5 Everything with Form is Unreal

From the previous sections, we can conclude that, both of formal languages and human languages are unreliable in some sense. Further, the strongest tool, computer, is not as such strong as we imagine.

During the time (the latter half of the 20^{th} Century) when many AI researchers turned to statistic methods, western philosophers also found a fact

[2] The details of diagonalization can be found at [2].

that many ideas in the linguistic philosophy were early discussed in Buddhism [4]. Gautama Buddha, who built Buddhism, said in different Buddhist sutras that language is unreliable and is really an obstruction for us to the Enlightenment.

According to the opinions of Gautama, we human beings begin to understand the world by mapping different meanings to what we see and hear. These meanings, also called *forms* by Buddhists, are always our subjective thoughts which are incomplete, full of contradictions, and cannot reflect the reality of the world. However we always tend to believe that such forms are real. For example, our ancestors believed that the earth was in the center of the universe for a long time, since the sunrise and sunset looked like that the sun was just moving around the earth (similar to the moon). Here, "*the earth was in the center of the universe*" is such a form that our ancestors mapped to what they saw.

It is obvious that human language is also a kind of form. We map different concepts (or words) to what we see and hear. As time passes, we tend to rely on such concepts to understand this world. However, Gautama said that we cannot define *truth* and *reality* using forms, since forms are just reflections of our mind. Further, Gautama gave a strong claim that *everything with form is unreal*. This claim is deduced in *The Diamond Sutra* and cited in other sutras.

Gautama underlined to his proteges that, if someone believes that there exists the Enlightenment, he will never reach the Enlightenment. In other words, "Enlightenment" is just a word consisting of 13 letters. Gautama also argued that there is even no "*self*", i.e., "self" is just an illusioned concept in our mind. Gautama said that almost all kinds of *sufferings* came from our persistence in "self". However, "self" is just a concept, but not a real existence according to Gautama. Thus, in many practices of Buddhists, e.g., *meditation*, people train themselves to jump beyond the bound of language, the constraint of "self", and all the other forms.

Backing to the unreliability of language, it seems that we have not any progress on the question "*Who am I? Why am I in this world?*". However, we indeed have a deeper understanding of this question and our languages. That is, our languages are unreliable.

6 Discussion

Due to the generation of data by sensor networks, social media and different organizations, there is an exponential growth of structured or semi-structured data [5]. In this background, the techniques of AI and knowledge engineering are being widely used to represent and manage data (or knowledge) for different domains. On the other hand, the issue of the unreliability of formal languages and human languages has to be faced as well. In this part, we try to give some philosophical thoughts from the perspective of knowledge engineering.

First, it is not appropriate to find the exact meaning of "intelligence". From the perspective of the linguistic philosophy, there does not even exist an exact and static definition of "intelligence". According to the idea of Gautama, "in-

telligence" may just be a word created by human and turns out to be a wishful thinking of human, rather than a nature existence. In this sense, it is not appropriate to use any formal language or model to explain what is "intelligence".

Second, we should trade off between completeness and incompleteness. Completeness is an important property to show whether the utilized formal languages and models are reliable. However, *incompleteness* is inevitable in the sense that we utilize highly expressive formal languages. There has been work where researchers carefully sacrifice the completeness (reliability) of the utilized logic languages to achieve a better computational efficiency for logic reasoning. The related method is also called *incomplete reasoning* or *approximate reasoning*.

Third, we should combine different forms for representing knowledge. According to the arguments of Gautama, any kind of form is unilateral, subjective, and a partial reflection of our mind. Thus, it is rewardless to rely on any form to understand this world and ourselves. However, we have to use languages, or different forms to represent and manage knowledge, and to communicate with other people. Therefore, it is better for us to combine different forms, i.e., different formal languages or models to represent and manage knowledge, rather than to be constrained in only one formal language or representation of knowledge.

7 Conclusions

In this paper, we briefly discussed the findings of Gödel and Wittgenstein. That is, both of formal languages and human languages are unreliable. We further strengthened this claim by introducing some views of Gautama. We finally discuss this issue from the perspective of knowledge engineering.

References

1. S. J. Bartlett. *Reflexivity: A Source Book in Self-reference*. Amsterdam: North-Holland/Elsevier Science Publishers, 1992.
2. M. D. Davis and E. J. Weyuker. *Computability, complexity, and languages - fundamentals of theoretical computer science*. Computer science and applied mathematics. Academic Press, 1983.
3. K. Gödel. Über formal unentscheidbare sätze der principia mathematica und verwandter systeme i. *J. Monatshefte für Mathematik Physik*, 38:173–198, 1931.
4. C. Gudmunsen. Wittgenstein and buddhism. *J. The International Association of Buddist Studies*, 3:122–126, 1980.
5. N. Kleiner, S. Sejdovic, S. Zander, T. Setzer, R. Studer, and S. Jähnichen. Big data, smart data and semantic technologies (BDSDST). In *Proc. of GI-Jahrestagung*, pages 1169–1170, 2015.
6. R. Studer, V. R. Benjamins, and D. Fensel. Knowledge engineering: Principles and methods. *J. Data Knowl. Eng.*, 25(1-2):161–197, 1998.
7. A. Tarski and J. Woodger. The concept of truth in formalized languages. *J. Corcoran*, 8:153–278, 1931.
8. L. Wittgenstein. Tractatus logico-philosophicus. *London: Routledge and Kegan Paul*, 7, 1922.

TEDL: A System for CCKS2016 Domain-Specific Entity Discovery and Linking Task

Feng Zhang, Tao Yang, Xiao Li, Qianghuai Jia, Ce Wang

Tencent Inc., Beijing China
{jayzhang, rigorosyang, chinali, jasonqhjia, fordwang}@tencent.com

Abstract. This paper describes the TEDL system for the entity discovery and linking, which compete the CCKS2016 domain-specific entity discovery and linking task. Given one review text and one pre-constructed movie knowledge base(MKB) from the douban website, we need to firstly detect all the entity mentions, then link them to MKB's entities. The traditional named entity detection(NED) and entity linking(EL) techniques cannot be applied to domain-specific knowledge base effectively, most of existing techniques just take extracted named entities as the input to the following EL task without considering the interdependency between the NED and EL and how to detect the Fake Named Entities(FNEs)[1]. In this paper, we employ one novel method described in [1] to joint model the 2 procedures as our basic system. Besides it, we also used the basic system's output as features to train models. Finally we ensemble all the models' output to predict FNE. The experiment results show that 80.30% NED F1 score and 93.45% EL accuracy, which is better than that of traditional methods.

KeyWords: Fake Named Entity, Entity Linking, Domain-specific Knowledge Base

1 Task Overview

Named Entity Detection(NED) and Entity Linking(EL) is one key step to bridge unstructured text with structured knowledge base(KB). It is widely studied in this area but mostly for the general KB, and the wikipedia is the most popular study target. Recently domain-specific KB has been found more effective and useful to manage and query knowledge with a specific domain, such as IMDB douban and mtime[1]. The domain specific KB contains more concrete entities.

One of the CCKS2016 task is the Domain-Specific Entity Discovery and Linking. It gives one movie knowledge base(MKB) from the douban website, wich contains about 100 thousand star and about 100 thousand movies. The input linking texts are the real people review for people or movies, including short comments, long reviews(more than 1000 characters), topics and synthetic reviews. The training data contains about 870 texts, and the test data contains about 420 texts. Besides this, it also contains 10+ concepts, 30+ properties.

2 System Design

Figure 1 is our whole NED and EL system overview for both offline and online process. The system including both offline and online process. The offline is mainly for mining more and more entities' alias to increase the coarse-grained recall. The online process is that given one input text, do the NED and EL steps and get the final result.

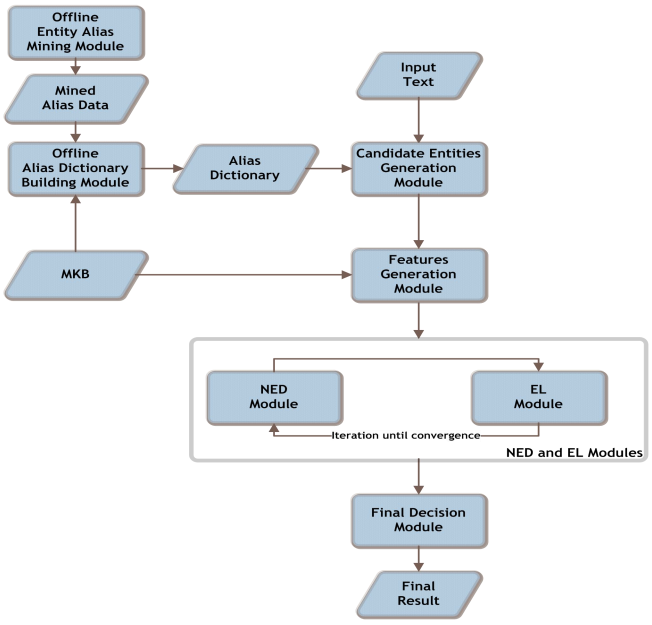


Fig.1. the TEDL system design overview

2.1 Offline Mining Alias and Dictionary Building Module

the entity alias mining is the key step for the whole system because it directly affect the subsequent modules for its coarse-grained recall. We tried below methods to build our alias dictionary:

Building the Initial Dictionary from the original MKB we build the initial entity alias dictionary. About 290 thousand entries.

Removing the Noise from Initial Dictionary there are much noise existing in the initial dictionary, such as “西游记(新版)”, “绝望的主妇第二季”, we should clear them.

Removing some very generic Alias there are some very generic entity names in the initial dictionary, such as “这个”, “时间”, we should remove them to avoid bring in much noise in the subsequent modules.

Mining Some Alias from Baidu Baike from baike’s info box, we could mine some alias, and also using the baike’s anchor we can also mine some.

Mining Some Alias from Search Query from the search engine’s queries we can mine some entity aliases[7].

Correcting the Spelling Error this method is implemented during the online. The main idea is the edit distance algorithm.

Generate alias for the foreign people. Such as the “尼古拉斯.凯奇”， split their names and keep the “凯奇” “尼古拉斯”, and then remove some very generic names.

After finishing all the above steps, the final entry number in the alias dictionary is about 460 thousand.

2.2 Candidates Generation Module

After building the alias dictionary, we use it to generate the candidates for one input entity mention. The main data structure is the trie tree and the edit distance algorithm for speller error detection.

2.3 Feature Generation Module

We treat the NED as the binary classification problem and the EL as the ranking problem. So we create about 56 features for NED model and about 17 features for EL model.

EL model features including below features:

a. The popular, WLM, jaccard and content similarity, 5 features in all[1].

b. The entity’s in-link, out-link, is people or not, 3 features in all.

c. Whether the movie’s actor, director occur in its context, and whether there is movie occurring among the actor’s context, 9 features in all.

NED model features including below features:

a. The link probability, WLM, jaccard and link certainty, 5 features in all[1].
b. Mean WLM and jaccard, 6 features in all.
c. Some segment feature, such as whether it is one phrase.
d. CRF features. We trained one CRF model using the training data.
e. Some context feature. Such as the context mention number, and above EL's c features.
f. Some mined popular people and video as feature.

2.4 NED and EL Module

For the domain specific KB, the key issue is the Fake Named Entity(FNE)[1], so to overcome this we employ the iteration process describe in [1]. More details about the iterative NED and EL models' training and evaluation in [1].

2.5 Final Decision Module

After the iteration process, we leveraged the boosting idea to train some other models to predict jointly using different training algorithm. So for the EL model, we trained one GBDT classification model and one learning to rank model. One is use the same features as the iterative EL model, and the other uses the features with NED dependent features removed. For the NED models we did the same things, one SVM model and one GBDT model, one EL dependent and one EL independent.

3 Experiments and Evaluation

To assess our system's performance, we build 2 baselines to compare. We refer to our TEDL as Treatment, Baseline1 is make the max iteration number as 1 and removed the final decision module, which is equivalent to the traditional process; baseline2 is the same as the treatment except removing the final decision module, which means that we use the iteration's result as the final result.

Approach	NED			EL	Overall(NED + EL)		
	precision	recall	F1	accuracy	precision	recall	F1
baseline1	74.00%	76.41%	75.19%	91.00%	67.34%	69.53%	68.42%
baseline2	76.85%	79.21%	78.01%	92.41%	71.02%	73.18%	72.08%
treatment	**79.33%**	**81.30%**	**80.30%**	**93.45%**	**74.13%**	**75.98%**	**75.43%**

Table.1. the TEDL system design overview

Table 1 shows the results. From the table we could see that the treatment has the best performance. Comparing the baseline2 and the baseline 1 we can see that the iteration process achieve +3.66% overall F1 score; comparing the treatment and the baseline2 we can see that adding the +3.35% overall F1 score, and for NED both the precision and recall increased. From the results we could draw the conclusion that the iteration process and the boosting method(final decision module) help a lot. The treatment's result is the final result we submitted.

4 Related Work

The NED and EL problem attracts a lot of people study recent years because it is the key step for many KB applications. The first system to figure out this problem is described by Bunescu and Pasca[2]. The system uses the wikipedia articles as the KB and view all the links as the unambiguous mentions of entity.[3] and [4] uses the learning to rank method to perform the EL's candidates ranking, and gets good results. [3] formulates the whole EL process as 4 sub modules: query processing, candidates generation, candidates ranking and top1 candidate validation. Most of existing approaches focus on the general purpose knowledge bases[1]. Many previous systems employed a pipeline frameworks[5, 6]. But in this paper we employed one novel method to model the 2 steps jointly, which is described in[1]. But besides the basic system, we create other subsequent models to predict FNE jointly, achieve good performance.

5 Conclusion

The current traditional EL system focus on the general KB instead of specific domain KB, which has many FNEs. So we employ one novel method which model the NED and EL jointly, which obtain the better result than the traditional methods. We also used the basic system's output as features to train models to predict FNE, the experiment shows that it can achieve better result.

References

1. Jiangtao Zhang.: Domain-Specific Entity Linking via Fake Named Entity Detection. In Database Systems for Advanced Applications Volume 9642 of the series Lecture Notes in Computer Science pp 101-116 (2015)

2. Razvan Bunescu and Marius Pasca: Using encyclopedic knowledge for named entity disambiguation. In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL), pages 9–16. Association for Computational Linguistics, Trento, Italy. (2006)
3. Z. Zheng, F. Li, M. Huang, and X. Zhu.: Learning to link entities with knowledge base. In NAACL, pp. 483-491. (2010)
4. D. Ceccarelli, C. Luccchese, S. Orlando, R. Perego, and S. Trani.: Learning relatedness measures for entity linking. In CIKM pp. 139-148 (2013)
5. Ratinov, L., Roth, D., Downey, D., Anderson, M.:Local and global algorithms for disambiguation to wikipedia. In: HLT'11. pp. 1375-1384
6. Sil, A., Cronin, E., Nie, P., Yang, Y., Popescu, A.M., Yates, A.: Linking named entities to any database. In:EMNLP-CoNLL' 12. pp. 116-127
7. Bei shi, Le Sun, Xianpei Han, Graph based alias extraction using query log. In journal of chinese information processing Vol 27, No. 5 Sep, 2013

Knowledge Graph Embedding for Link Prediction and Triplet Classification

Shijia E, Shengbin Jia, and Yang Xiang

Tongji University, Shanghai 201804, P.R. China,
e.shijia@gmail.com, {shengbinjia,shxiangyang}@tongji.edu.cn

Abstract. The link prediction (LP) and triplet classification (TC) are important tasks in the field of knowledge graph mining. However, the traditional link prediction methods of social networks cannot directly apply to knowledge graph data which contains multiple relations. In this paper, we apply the knowledge graph embedding method to solve the specific tasks with Chinese knowledge base *Zhishi.me*. The proposed method has been successfully used in the evaluation task of CCKS2016. Hopefully, it can achieve excellent performance.

Keywords: knowledge graph, distributed representation, entity embedding

1 Introduction

In traditional social networks, the link prediction task is one of the important technologies to discover the relationships among users [1]. Within the link prediction of social network, the *connection* between two users is often said to be a friend relationship. However, in the knowledge graph, the knowledge network is composed of entities and relations. A connection with two entities can be denoted as a triplet (h, r, t), where h is the head entity, t is the tail entity, and the relation between them is represented as r. Different from the social networks, the *connection* in the knowledge graph is usually with a direction, *e.g.* for the triplet *(Yao Ming, born in, Shanghai)*, the relation *born in* is a way from *Yao Ming* to *Shanghai*, but we could not say *Shanghai* was born in *Yao Ming*. Therefore, the traditional link prediction methods used in social networks are not suitable for the link prediction task in the knowledge graph. In addition, because of the flexibility of Chinese language, the rule based natural language precessing (NLP) methods often require a lot of manual intervention.

In this paper, we adopt the representation learning to understand the knowledge graph provided by *zhishi.me*, and embed the entities and relations of the knowledge graph into a low dimensional vector space. The vector representation of the entities and the relations will contain the semantic relationships among them.

The rest of this paper is structured as follows. In section 2, we describe our model architecture used in the evaluation task. In section 3, we summarize the

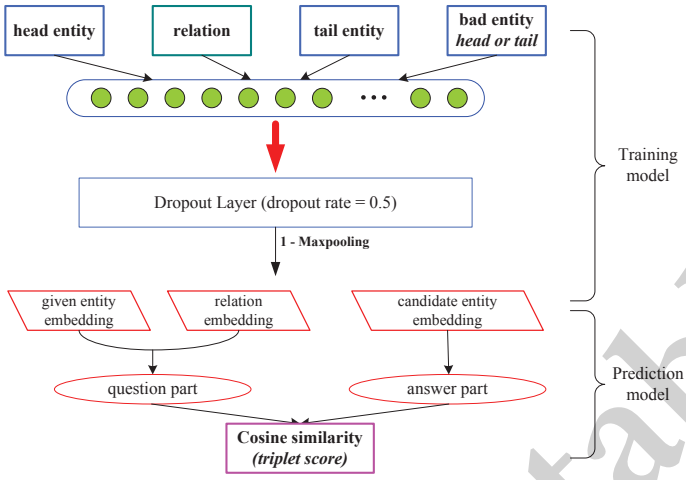


Fig. 1. The overall neural network architecture of our model

experiment setup of our model. The application of our model is presented in Section 4. Section 5 contains related work and finally we give some concluding remarks in Section 6.

2 The Embedding Model for Knowledge Graphs

In this section we describe the proposed deep neural networks to solve the LP and TC problems. Figure 1 shows the overall framework of our model. The training part aims to learn the semantic relationships among entities and relations with the negative entities (bad entities), and the goal of the prediction part is giving a *triplet score* with the vector representations of entities and relations. The following is a detailed description.

2.1 Data preprocessing

The dataset of the evaluation task is from the Chinese knowledge base *zhishi.me*, and the basic statistics of the data are shown in Table 1. In order to meet the requirement of the evaluation task, we first number the entities and relations in turn. During the training time, different IDs represent different entities and relations. This kind of representation can be convenient for us to do the vectorize operations.

2.2 Core architecture of knowledge graph embedding

For a given triplet (h, r, t) in the training set, our model will learn the vector representations of h and t as well as the r, denoted as $\mathbf{h}$, $\mathbf{t}$ and $\mathbf{r}$. The core idea

Table 1. Data set used in the evaluation task

Dataset	#Entities	#Relations	#Triplets (Train)
zhishi.me	644699	3512	7063189

of the model is that transforming the link prediction problem into a question and answer mode, *i.e.* $\mathbf{h} + \mathbf{r}$ expresses the question, and $\mathbf{t}$ is the answer, or $\mathbf{t}$ - $\mathbf{r}$ is question, and $\mathbf{h}$ expresses the answer.

Based on the above ideas, in order to learn the proper vector representations, our neural networks are trained to minimize the following loss function with the training data (illustrated by the example of tail entity prediction):

$$L = max\{0, m - cos(\mathbf{h} + \mathbf{r}, t^+) + cos(\mathbf{h} + \mathbf{r}, t^-)\} \quad (1)$$

where $m > 0$ is the margin hyper-parameter, t^+ and t^- denote the correct tail entity and wrong tail entity respectively. Unlike the TransE [2] or TransM [3] model that use the L_1 or L_2 norm as the dissimilarity measure, we use the cosine similarity (cos) to judge the matching degree of *question* and *answer* which can be called as **matching score**. After training with the loss function, it turns out that the loss value of the correct triplet is less than its corresponding wrong triplets. m is used to control the degree of deviation.

During the training process, at every epoch, we randomly sample a wrong entity which is from the whole entity set to each correct triplet in the training set. As a result, the four tuple (h, r, t^+, t^-) (or (h^-, h^+, r, t)) forms a training sample. As Figure 1 shown, we add a *Dropout* layer after the *Embedding* layer to improve the generalization ability of the model and prevent overfitting [4]. Besides that, we add a 1-MaxPooling layer. The vector representation after the pooling layer is treated as the final embedding of the entity or relation which will be used in the loss function.

3 Experiment Setup

In this section, we describe the parameters and experiment environment used in this evaluation task. The parameters need to be fine tuning with different tasks.

3.1 Parameter settings

In this evaluation task, the margin value m was 0.05, and the embedding dimension of entities and relations was 100. We also tried 200 or 1000 dimensions, and it can get better result on a small dataset (split from training set). However, on the whole dataset, it was more costly. The optimization method employed was Adam [5], and it was more computationally efficient than basic stochastic objective function (SGD). The learning rate was 0.001, and the batch size was 512 per epoch. We trained 200 epochs for the predictions of head entity and tail entity respectively.

3.2 Training environment

The model used in this evaluation task was implemented with Keras[1]. We used a Tesla K20c GPU device to train the model. Due to time constraints, we believe our model can get better results after longer training time.

4 Applications of the Model

In the triple link prediction tasks, our model would treat all available entities as the candidates for each test sample in the test set ($(h, r, \_)$ or $(\_, r, t)$). The trained model would give the matching scores to each *question* and *answer* pairs, and entities ranked at the highest top 200 could be saved as the submitted results.

As to the triplet classification task, we adopted the tail entity prediction model as the test model. For the triplet given by the test set, the model would give the matching score of the test samples. Our strategy was that if the triplet's score was greater than or equal to 0.55, it was considered to be valid, otherwise we tagged it as an invalid one.

5 Related Work

The model used in this evaluation task is related to the following two research areas.

Distributed Representation Learning. It plays an important role with the development of deep learning. The related methods can been applied to various fields, such as NLP, computer vision and image processing [6]. Especially, models based on word embedding have been achieved good performance in the field of text classification [7]. It makes it possible to train on large scale data with limited resources. Inspired by the word embedding model, such as *word2vec*[2], a lot of similar models have emerged recently. *Paragraph Vector* and *Doc2vec* [8] are extensions of *word2vec*, and they learn the vector representations of paragraphs and documents. Essentially, the core of the ideas is to make a good text representation which can express proper semantic information in a specific environment. The embedding models on knowledge graph data also try to catch the key semantic relationships hidden in the numbers of entities, and we can absorb the advantages of those models to help us learn the structure of knowledge graphs.

Knowledge Graph Completion. It aims to predict relations between entities of an existing knowledge graph. It has been several *translation* based methods, such as TransE, TransM, TransR [9] and Hole model [10]. The knowledge graph embedding models with the representative of the TransE have made remarkable achievements in the knowledge graph completion task with the specific datasets. In essence, all of them try to find out a comprehensive and effective

[1] `https://keras.io`

[2] `https://code.google.com/archive/p/word2vec/`

rule which *translates* head entities to tail entities. For the evaluation task in the paper, the scale of the data is far beyond the dataset used in existing experiments. Therefore, we should develop a more effective method to tackle this problem.

6 Conclusion

We describe a deep neural network method with distributed representation to solve the triplet prediction and triplet classification evaluation tasks. Our model can be trained fast with advanced GPU devices and easily extended to other similar tasks.

In addition, the entity candidates in the task is really large. If we can figure out a way to reduce the size of search space, maybe the test result will be better.

References

1. Liben-Nowell, D., Kleinberg, J.: The link-prediction problem for social networks. Journal of the American society for information science and technology **58**(7) (2007) 1019–1031
2. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems. (2013) 2787–2795
3. Fan, M., Zhou, Q., Chang, E., Zheng, T.F.: Transition-based knowledge graph embedding with relational mapping properties. In: Proceedings of the 28th Pacific Asia Conference on Language, Information, and Computation. (2014) 328–337
4. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15**(1) (2014) 1929–1958
5. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
6. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Cognitive modeling **5**(3) (1988) 1
7. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
8. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: ICML. Volume 14. (2014) 1188–1196
9. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: AAAI. (2015) 2181–2187
10. Nickel, M., Rosasco, L., Poggio, T.: Holographic embeddings of knowledge graphs. arXiv preprint arXiv:1510.04935 (2015)

Knowledge Base Completion via Rule-Enhanced Relational Learning

Shu Guo[1,2], Boyang Ding[1,2], Quan Wang[1,2⋆], Lihong Wang[3], and Bin Wang[1,2]

[1]Institute of Information Engineering, Chinese Academy of Sciences
[2] University of Chinese Academy of Sciences
[3]National Computer Network Emergency Response Technical Team Coordination Center of China

Abstract. Traditional relational learning techniques perform the knowledge base (KB) completion task based solely on observed facts, ignoring rich domain knowledge that could be extremely useful for inference. In this paper, we encode domain knowledge as simple rules, and propose rule-enhanced relational learning for KB completion. The key idea is to use rules to further refine the inference results given by traditional relational learning techniques, and hence improve the inference accuracy of them. Facts inferred in this way will be the most preferred by relational learning, and at the same time comply with all the rules.

Keywords: Knowledge base completion, relational learning, rules

1 Introduction

Knowledge bases (KBs) are extremely useful resources for many NLP tasks. They provide large collections of facts about entities and their relations, typically stored as triples, *e.g.*, (`Beijing`, `capitalOf`, `China`). Although such KBs can be very large, they are still quite incomplete. KB completion, *i.e.*, automatically inferring missing facts from existing ones, has thus attracted increasing attention. Various relational learning techniques have been proposed for this task [1–6].

Most existing relational learning techniques, *e.g.*, the embedding-based TransE model [2] and the path ranking algorithm (PRA) [3] make inferences based solely on facts in KBs. They ignore rich domain knowledge which might also be useful for inference. For example, given the fact (`Beijing`, `capitalOf`, `China`), one can easily infer that `Beijing` cannot be the capital of any country other than `China`, by using the domain knowledge about `capitalOf`. Domain knowledge is usually encoded as rules, and has been applied in a variety of inference tasks [7–9].

In this paper, we propose rule-enhanced relational learning, specifically rule-enhanced TransE and PRA for KB completion. The key idea is to incorporate additional rules (*i.e.*, domain knowledge) to further refine the inference results given by TransE and PRA, and hence enhance the inference accuracy of them. Facts inferred in this way will be the most preferred by the relational learning techniques, and at the same time comply with all the rules.

⋆ Corresponding author: Quan Wang (wangquan@iie.ac.cn).

2 Our Approach

Our approach consists of two key components: 1) relational learning techniques of the TransE model and the path ranking algorithm (PRA); 2) rules imposed to further refine inference results.

2.1 TransE Model

TransE [2] is an embedding-based technique which is simple and efficient while achieving state-of-the-art predictive performance. The key idea of TransE is to embed entities and relations in a KB into a continuous vector space, and make inferences in that space.

Specifically, TransE represents entities and relations as vectors in the embedding space. Given a triple (e_i, r_k, e_j) and the embeddings $\mathbf{e}_i, \mathbf{e}_j, \mathbf{r}_k \in \mathbb{R}^d$, TransE assumes that $\mathbf{e}_i + \mathbf{r}_k \approx \mathbf{e}_j$. A score function $f(e_i, r_k, e_j) = -\|\mathbf{e}_i + \mathbf{r}_k - \mathbf{e}_j\|_1$ is further defined on each triple. Plausible triples are assumed to have high scores. To learn these embeddings, a margin-based ranking loss is minimized, *i.e.*,

$$\min_{\{\mathbf{e}\},\{\mathbf{r}\}} \sum_{t^+ \in \mathcal{O}} \sum_{t^- \in \mathcal{N}_{t^+}} \left[\gamma - f(e_i, r_k, e_j) + f(e'_i, r_k, e'_j)\right]_+ .$$

Here, $t^+ = (e_i, r_k, e_j) \in \mathcal{O}$ is a positive (observed) triple; $\mathcal{N}_{t^+}$ denotes the set of negative triples constructed for t^+, and $t^- = (e'_i, r_k, e'_j) \in \mathcal{N}_{t^+}$; $\gamma > 0$ is a margin separating positive and negative triples; and $[x]_+ = \max(0, x)$. Stochastic gradient descent (in mini-batch mode) is adopted to solve this problem. In each stochastic iteration, we generate two negative triples for each t^+, one by replacing the head entity and the other the tail entity. To replace a position (head or tail), we use only entities that have appeared in that position (with the same relation).

2.2 Path Ranking Algorithm

PRA [3] is an inference technique that uses paths connecting two entities to predict potential relations between them. Here a path is a sequence of relations that link two entities. For example, `bornIn` $\rightarrow$ `capitalOf` is a path linking `ZhangZiyi` to `China`, through an intermediate node `Beijing`. Such paths are then used as features to predict the presence of specific relations, *e.g.*, `nationality`.

Specifically, for each target relation, PRA first generates a set of training instances, *i.e.*, pairs of entities that are linked by the relation (positive instances) or not (negative instances). Then, we employ depth-first search [10] to enumerate all paths with bounded lengths linking the two entities in each training instance. Besides paths, path bigrams are also included as features [11]. The feature values are simply determined by frequency. Finally, we use two-level stacking [12] to combine multiple classifiers, so as to judge whether two entities should be linked by the target relation. We choose 7 base-level classifiers: 1) three decision forest models of random forest [13], ExtraTree [14], and XGBoost [15]; 2) four logistic regression models with different seeds. A meta-level logistic regression classifier is then trained by taking predictions of the base-level classifiers as input features.

Dataset	# Ent.	Train-I		Train-II		# Test-lph	# Test-lpt	# Test-tc
		# Rel.	# Trip.	# Rel.	# Trip.			
Baidu	86,272	6	40,967	381	566,028	24,613	20,252	75,991
Hudong	418,529	5	328,927	298	4,679,917	114,928	74,857	176,598
Zhwiki	144,314	9	17,266	2,819	1,163,405	72,719	86,607	155,772

Table 1. Statistics of data sets.

2.3 Rules Imposed

We further introduce three types of rules to refine the inference results given by TransE and PRA.

Rule 1 (simple implication). Suppose relation r_1 implicates relation r_2, denoted as $r_1 \mapsto r_2$. Then, any two entities linked by r_1 should also be linked by r_2. For example, `capitalOf` $\mapsto$ `locatedIn`.

Rule 2 (argument type restriction). Arguments of a relation should be entities of certain types. For example, the tail argument of the relation `capitalOf` need to be `Country` entities.

Rule 3 (at-most-one restriction). For 1-To-Many/Many-To-1 relations, the head/tail argument can take at most one entity; for 1-To-1 relations, both arguments can take at most one entity.

By applying these rules directly on observed facts, we obtain additional evidence which can be used to refine the inference results given by TransE and PRA.

3 Experimental Setups

3.1 Data Sets

The released Zhishi corpus consists of three KBs: Baidu, Hudong, and Zhwiki. For each KB, we split it into two parts, Train-I and Train-II by relation. Train-I contains name-related relations like `chineseName`. Such relations can be handled simply by string matching, and hence are not included in relational learning. The other relations are contained in Train-II. We further split Train-II into a training set and a validation set with nearly 5000 triples, used for model training and parameter tuning respectively. Test data is released separately. Table 1 gives some statistics of the data sets, where # Test-lph/# Test-lpt/# Test-tc denotes the number of test triples used for link prediction of head entities, link prediction of tail entities, and triple classification respectively.

We manually create 5/8/4 simple implication rules for Baidu, Hudong, and Zhwiki. For Rule 2, by following the closed-world assumption, we assume that the head/tail argument of a relation can take only entities that have appeared in the same position with that relation. For Rule 3, to identify the relation type (*i.e.*, 1-To-Many, Many-To-1, or 1-To-1), we compute the average number of heads (tails) per tail (head). If the average number is smaller than 2, we label the head (tail) argument as "1" or "Many" otherwise.

3.2 Link Prediction

This task is to complete a triple (e_i, r_k, e_j) with e_i or e_j missing, i.e., predict e_i given (r_k, e_j) or predict e_j given (e_i, r_k). TransE is used for this task.

Evaluation protocol. For each test record $(?, r_k, e_j)$ or $(e_i, r_k, ?)$, we take every entity e' in the dictionary as a candidate answer and calculate its plausibility. If r_k is name-related, the plausibility is defined as the string similarity between e' and e_j/e_i. Otherwise, the plausibility is the score given by TransE. Ranking the plausibility in descending order, we get a list of candidate answers. For each candidate answer, if the resultant triple can be directly inferred by Rule 1, we boost it to the top of the list; and if the triple violates Rule 2 or Rule 3, we remove it from the list. We then return the top 200 candidates and record the rank of the correct answer (not released).[1] Aggregated over all test records, we report: 1) the averaged rank (Mean), and 2) the proportion of ranks no larger than n (Hits@n).

Implementation details. We create 100 mini-batches on each KB. The best model is selected by early stopping on validation sets (by monitoring $S = 30\% \times \big(1 - \frac{\text{Mean}}{200}\big) + 30\% \times \text{Hits@10} + 10\% \times \text{Hits@3}$), with a total of at most 1000 iterations. The optimal configurations are: the dimension of the embedding space $d = 70$, the margin $\gamma = 4$, the learning rate for entity $\eta_e = 0.005$, and for relation $\eta_r = 0.0001$ on BAIDU; $d = 70$, $\gamma = 2$, $\eta_e = 0.005$ and $\eta_r = 001$ on HUDONG; $d = 70$, $\gamma = 5$, $\eta_e = 0.001$ and $\eta_r = 0001$ on ZHWIKI.

3.3 Triple Classification

This task is to verify whether a given triple $\langle e_i, r_k, e_j \rangle$ is correct or not. Both TranE and PRA are used for this task.

Evaluation protocol. Given a test triple (e_i, r_k, e_j), we take it as positive if it can be directly inferred by Rule 1, and negative if it violates Rule 2 or Rule 3, without further prediction. For name-related relations, we simply use string matching. A triple is predicted to be positive if the string similarity between the two entities is higher than 0.7. The other relations are handled by either TransE or PRA. For TransE, a triples is predicted to be positive if its score is above a relation-specific threshold δ_r; while for PRA, we can just use the (meta-level) classifier trained for each relation. We choose accuracy (Acc) as the evaluation metric. Relations with Acc higher than 75% on validation sets are handled by PRA, and the others by TransE.

Implementation details. For TransE, δ_r is determined by maximizing Acc on validation sets, again, with a total of at most 1000 iterations. The other hyperparameters are set to the optimal configurations as used in link prediction. For PRA, during training, we generate two negative instances for each positive one, one by corrupting the head, and the other the tail. The maximum path length is set to 3. On BAIDU, we use stacking to train a meta-level classifier. The number of trees nt is set to 300 for random forest, 300 for ExtraTree, and

[1] If the correct answer is not included in the 200 candidates, we give it a rank of 201.

Test-lph			Test-lpt			Test-tc	Overall
Mean	Hits@3 (%)	Hits@10 (%)	Mean	Hits@3 (%)	Hits@10 (%)	Acc(%)	
4.15	95.50	96.98	20.30	51.954	68.09	67.69	80.61

Table 2. Link prediction and triple classification results on the test data of ZHISHI.

1000 for XGBoost in the base-level classifiers. On HUDONG and ZHWIKI, we use standard random forest, with *nt* set to 1000. All the classifiers are implemented using publicly available tools [2].

4 Results and Conclusion

The experimental results on the three KBs are aggregated and summarized in Table 2. We can see that our approach performs quite well on both tasks, achieving the best overall performance in the CCKS 2016 competition. (The overall performance is evaluated as $30\%\times(1-\frac{\text{Mean}}{200})+30\%\times\text{Hits@10}+10\%\times\text{Hits@3}+30\%\times\text{Acc.}$) The results demonstrate the superiority of incorporating domain knowledge into traditional relational learning.

References

1. Nickel, M. , Nickel, V, Kriegel, H. -P.: A three-way model for collective learning on multi-relational data. In: Proceedings of ICML, pp. 809–816 (2011)
2. Bordes, A., Usunier, N., GarciaDurán, A., Weston, J. and Yakhnenko, O.: Translating embeddings for modeling multirelational data. In: Proceedings of NIPS, pp. 2787–2795 (2013)
3. Lao, N., Cohen, W. W.: Relational retrieval using a combination of path-constrained random walks. MACH LEARN. 81(1), pp. 53–67 (2010)
4. Richardson, M., Domingos, P: Markov logic networks. MACH LEARN, 62(1-2), pp. 107–136 (2006)
5. Wang, Q., Liu, J., Luo, Y., Wang, B., Lin, C.: Knowledge Base Completion via Coupled Path Ranking. In: Proceedings of ACL, pp. 1308-1318 (2016)
6. Guo, S., Wang, Q., Wang, B., Wang, L., Guo, L.: Semantically smooth knowledge graph embedding. In: Proceedings of ACL, pp. 84C94 (2015)
7. Rocktäschel, T., Singh, S., Riedel, S: Injecting logical background knowledge into embeddings for relation extraction. In: Proceedings of NAACL, pp. 1119–1129 (2015)
8. Wang, Q., Wang, B, Guo, L.: Knowledge base completion using embeddings and rules. In: Proceedings of IJCAI, pp. 1859–1865 (2015) Zhuoyu Wei, Jun Zhao, Kang Liu, Zhenyu Qi, Zhengya
9. Wei, Z., Zhao, J., Liu, K., Qi, Z., Sun, Z., Tian, G.: Large-scale knowledge base completion: inferring via grounding network sampling over selected instances. In: Proceedings of CIKM, pp. 1331–1340 (2015)
10. Shi, B., Weninger, T.: Fact checking in large knowledge graphs: A discriminative predict path mining approach. In: arXiv:1510.05911 (2015)
11. Gardner, M., Mitchell, T.: Efficient and expressive knowledge base completion using subgraph feature extraction. In: Proceedings of EMNLP, pp. 1488–1498 (2015)
12. Wolpert, D. H.: Stacked Generalization. NEURAL NETWORKS, 5, pp. 241–259 (1992)
13. Breiman, L.: Random Forests. MACH LEARN. 45(1), pp. 5–32 (2001)
14. Geurts,P., Ernst, D., Wehenkel, L.: Extremely randomized trees. MACH LEARN, 63(1), pp. 3–42 (2006)
15. Chen T., He T.: XGBoost: A Scalable Tree Boosting System. In: Proceedings of KDD (2016)

[2] http://scikit-learn.org/stable/

Product Prediction with Deep Neural Networks

Shijia E and Yang Xiang

College of Electronics and Information Engineering,
Tongji University, Shanghai 201804, P.R. China,
e.shijia@gmail.com, shxiangyang@tongji.edu.cn

Abstract. In this paper, we give a solution to the product prediction shared task of CCKS 2016. The main purpose of the task is to determine the product categories for the import and export transaction record data. For this specific dataset, we apply deep neural networks to solve the multi-label classification problem. On the training set, our proposed method achieves a precision of 0.90, and the proposed model can have a good performance on the test set.

Keywords: multi-label classification, neural networks, product prediction

1 Introduction

For the classification problem, traditional methods are focus on learning from a set of examples with only single label, called the binary classification. Nowdays, more classification tasks are often multi-label classification problems. In those tasks, the examples usually belong to more than two categories, even hundreds of categories. In this evaluation task, the training data contains seven basic attributes, of which there are two numeric fields: *Quality* and *Price*, five discrete attributes: *Enterprise, Destination, Origin, Custrom* and *Product*. The *Product* field is the target field for the prediction task, and the remainder of the attributes is known to the training attribute. However, the test set of the evaluation task does not contain the attribute of *Quality*. Therefore, for this product prediction task, we have not used the *Quality* attribute as an input feature during the model training process.

In this paper, according the existing data size, we directly use a multi-layer perceptron (MLP) neural network architecture. After 5000 epochs, the accuracy of the training data can reach 90%.

The rest of this paper is structured as follows. In Sect. 2, we describe our model architecture used in the evaluation task. In Sect. 3, we summarize the experiment setup with a discussion of our model. Section 4 contains related work and finally we give some concluding remarks in Sect. 5.

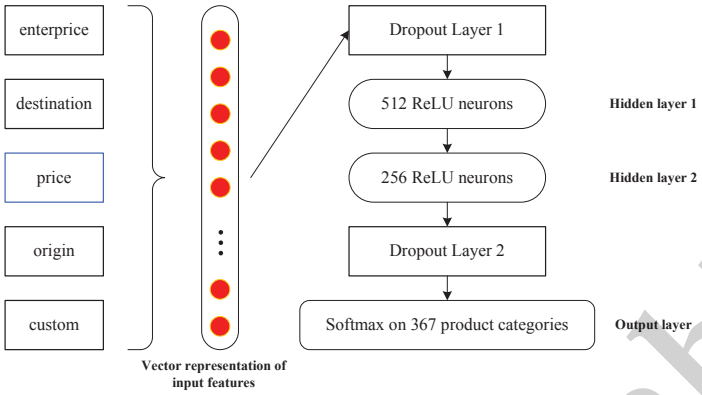


Fig. 1. Architecture of our proposed model

2 The Deep Neural Network Model for Product Prediction

In this section we describe our solution to this specific task. Our main idea is to design an end-to-end method with as little feature engineering as possible, even with no model ensemble. We will give a detail discussion of several models in Sect. 3.2. The final architecture of our model is demonstrated in Fig. 1.

2.1 Data preprocessing

In order to allow the data to be trained with the deep neural networks, the numeric attribute (*Price*) and discrete attributes need to be unified into vector representations. The preprocessed data will be treated as the input layer of the neural networks.

There are 18279 training samples and 767 test samples in the provided dataset. Due to the discrete attributes of the current dataset only contain 855 values, we directly apply the one-hot encoding method to the input attributes, *i.e.* each discrete attribute can be expressed as a vector with 855 dimensions, and the continuous numerical attribute *Price* is directly used as another dimension of the vector. Therefore, the input features of each sample can be used a vector with 856 dimensions to express.

2.2 Model description

The core architecture of our model is based on a MLP which is one of the simplest neural network architectures [1]. Specifically, a MLP consists of an input layer, one or more hidden layers, and an output layer. The input layer is always with fixed size representation of input variables, and the hidden layer is used

to calculate the intermediate representation of the input variables. Finally, the output layer is used to give the prediction of the output value.

In the MLP architecture, we use the pre-processed data as the input layer of the neural network, then we add the two hidden layers, and the last layer is output layer based on *Softmax* which is a generalization of the logistic function to fit the multi-label classification. In addition, we add two *Dropout* layers [2] before the first hidden layer and the output layer to prevent the model overfitting.

Based on the above ideas, the objective function used in our neural networks is multi-class log loss, also known as the categorical cross-entropy. It is really a common used loss function in the field of multi-label classification. It can be optimized by stochastic gradient descent. The overall model is just like a linear stack of layers, simple but effective.

2.3 The output of our model

There are 364 categories of the output products. As a result, there are 364 neurons in the output layer based on *Softmax* function. The output of that is a probability distribution over the 364 target categories, and the sum of these probabilities is 1. Therefore, for any given sample in the test set, the model is able to return the probability that the sample belongs to any category, and the categories with the top 3 probabilities among the 364 targets are selected as the final prediction.

3 Experiment Setup and Discussions

In this section, we describe the parameters and experiment environment used in this evaluation. In addition, we give some discussions of the models we have ever tried.

3.1 Parameter settings

In this evaluation task, we used two hidden layers, the number of neurons in the first hidden layer was 512, and the second was 256. The activation function we used was $ReLU$, and we initialized the network weights with the normal distribution. The optimization method we choose was *Adam* [3], and it was a variant of the typical stochastic gradient descent (SGD). As mentioned before, we added two *Dropout* layers. The dropout rates are 0.25 and 0.5 respectively. The learning rate was set to be 0.001, and the batch size was 128 per epoch. We trained 5000 epochs with a Tesla K20c GPU device. It just took a few minutes to complete the training phase.

3.2 Discussions

For this specific task, we also tried more sophisticated neural network architectures, such as the embedding model inspired by natural language processing

(NLP) and something relates to long-short term memory networks (LSTM). The more advanced neural networks didn't get better results than the original MLP.

For the embedding model, we treated the discrete attributes within a sample as the words in a sentence. We wanted to learn the hidden relationships among those attributes and hoped that the relationships can reflect some key features of the product to help the model do the prediction. But the results showed the semantic relationships among these attributes were not much valuable. Because the relevance among these attributes was not particularly strong, the embedding model couldn't play its unique role. As for the LSTM models, we tried to convert the task into a sequence prediction problem, but we didn't make a good performance with a longer training time. It was because the product category was not a input sequence item in the provided samples. Therefore, the memory network couldn't learn a good understanding of the transaction data.

We could figure out that even a simple model can achieve a satisfiable result, and to solve certain specific problems, complex models are not always necessarily required.

4 Related Work

The product prediction of the task is just a type of multi-label classification. There are several related methods in this research area. [4] proposes a system based on the k-NearestNeighbor (kNN) classifier for multi-label document classification. Its main shortcoming, however, is for real-world use, where the number of labels of a new document is indeterminate. Liu and Chen [5] have made a detailed empirical study of different multi-label classification methods on sentiment classification. We can see that the method with best performance is rely on a high quality sentiment dictionary. It needs more extra resources to do the multi-label classification.

Besides the traditional methods, the deep neural networks (DNNs) also have made a good progress in the field of multi-label classification. Ciregan and Meier et al. apply the DNNs to image classification [6] and traffic sign classification [7]. [8] uses the deep convolutional neural network (CNN) for fine-grained image classification. Apart from the image processing area, the DNNs play a import role in the field of NLP as well. [9] and [10] use the CNN for sentiment classification. [11] uses word embeddings for document classification. All these methods show that the DNNs can make a better performance with large dataset than the traditional rule based methods. The model proposed in this paper is also an effective attempt in the multi-label classification tasks.

5 Conclusion

In this paper, we have introduced a effective deep neural network model to solve the product prediction task. Our model can perform prediction on any import and export transaction records without product categories. The model is able to deal arbitrary size of data. In addition, our results show that we don't have to

be obsessed with complex models. In practice, often simple and effective models can also be achieved satisfiable results.

References

1. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by error propagation. Technical report, DTIC Document (1985)
2. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15**(1) (2014) 1929–1958
3. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
4. Luo, X., Zincir-Heywood, A.N.: Evaluation of two systems on multi-class multi-label document classification. In: International Symposium on Methodologies for Intelligent Systems, Springer (2005) 161–169
5. Liu, S.M., Chen, J.H.: A multi-label classification based approach for sentiment classification. Expert Systems with Applications **42**(3) (2015) 1083–1093
6. Ciregan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for image classification. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE (2012) 3642–3649
7. CireşAn, D., Meier, U., Masci, J., Schmidhuber, J.: Multi-column deep neural network for traffic sign classification. Neural Networks **32** (2012) 333–338
8. Xiao, T., Xu, Y., Yang, K., Zhang, J., Peng, Y., Zhang, Z.: The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 842–850
9. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. arXiv preprint arXiv:1404.2188 (2014)
10. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
11. Kusner, M.J., Sun, Y., Kolkin, N.I., Weinberger, K.Q.: From word embeddings to document distances. In: Proceedings of the 32nd International Conference on Machine Learning (ICML 2015). (2015) 957–966

ICRC-DSEDL：基于知识图谱的影视领域实体发现与链接系统

李昊迪， 汤步洲[1], 陈清财，胡江鹭，张广鹏

（哈尔滨工业大学深圳研究生院 智能计算研究中心， 深圳， 518000）
lhd911107@gmail.com, tangbuzhou@gmail.com,qingcai.chen@gmail.com
hujianglu.hit@gmail.com, zhangguangpeng_hit@163.com

Abstract. 命名实体是文本中的重要单元,，正确分析存在歧义的命名实体对理解文本起着关键作用。影视领域的命名实体发现和链接任务相比于传统新闻领域的实体发现和链接有所异同，例如需要区分演员姓名和角色姓名。本文面向影视领域的评论文本，抽取其中的影视作品和人物并且对应到知识图谱的标准实体上，依此将主要任务分为实体识别和实体链接。在实体识别部分，我们主要采用了序列标注方法作为主要方法，通过豆瓣实体知识库、拼音、词性以及深度学习方法抽取特征，抽取出影视作品和人物。在实体链接部分，我们通过排序学习的方法构建模型，结合知识图谱与百度百科、豆瓣等开放知识库产生候选实体条目，同时从评论文本中抽取特征，对候选实体进行排序，最终找到目标实体。在CCKS官方测试数据集上，我们的系统分别取得了76.12%（F1值，实体识别）、86.53%（正确率，实体抽取）以及65.87%（F1值， Overall）的性能。

Keywords: 实体识别，实体链接，条件随机场，序列标注，排序学习

1 引言

命名实体识别(Named Entity Recognition, NER)[1]主要是研究如何从文本中将人名、地名以及机构名等专有名词识别出来，并且将他们分类。命名实体识别属于未登录词识别的范畴，对于这类未登录词的识别，一直是中文信息处理领域研究的热点问题之一[2]。

基于统计的命名实体识别方法是目前的主流方法，其基本思想是通过对人工标注的语料进行统计分析，从中学习到相应的知识，构建出标注器，然后利用构建出来的标注器去对文本进行标注，常用的方法包括隐马尔科夫模型，决策树等。

排序学习（Learning to rank） [3]方法在实体链接领域效果较为理想。实体链接问题可以转化为排序问题，对于一个给定实体，在知识库中先找出候选实体，然后在文本中提取特征，用排序学习方法进行学习排序，最后返回最优结果。

[1] 通信作者

adfa, p. 1, 2011.

本评测任务为限定领域的实体发现与实体链接，简称DSEDL (Domain-Specific Entity Discovery and Linking)。即对于给定的一组限定领域的纯文本文件，任务的目标是识别并抽取出与领域相关的实体提及(mention)，并将它们链接到给定知识库对应的实体(entity)。实体名字具有歧义性和变异性，也就是同一个实体名字,有可能指代多个实体，需要根据上下文消歧；此外，同一个实体可能有多个实体名字与之对应，比如别名、绰号、昵称等等，这些所有的名字变型均需识别。

CCKS 2016 Task 1的评测任务[2]限定在影视领域, 由清华大学计算机系知识工程实验室、豆瓣、微软亚洲研究院联合举办。影视评论中出现的与影视相关的实体名字分为两大类：影视人物及影视作品。影视人物包括演员、导演、制片人、编剧、主持人等，影视作品包括电影、连续剧、综艺节目等。据此，我们对实体识别和实体抽取任务分别构建了两个独立的子模块，结合给定的知识图谱以及开放的知识库，构建一个流水线系统。在CCKS官方测试数据集上，我们的系统分别取得了76.12%（F1值，实体识别）、86.53%（正确率，实体抽取）以及65.87%（F1值， Overall）的性能。

2 相关工作

从上个世纪末开始，消息理解会议（Message Understanding Conference, MUC）、自动内容抽取会议（Automatic Content Extraction, ACE）、多语言实体任务会议（Multilingual Entity Task, MET）等多种会议不断被开展，信息抽取（Information Extraction, IE）的研究逐渐被发展并被推广。

信息理解会议在信息处理的研究上有着重要的推动作用，命名实体识别作为一项任务被研究，最早可以追溯到1991年的第7届IEEE人工智能应用会议，在这届会议上，Ran发表了一篇关于“抽取和识别公司名称”的文章[4]，文章中，Ran介绍了一个能够识别、抽取公司名称的系统，该系统在实现时，采用了基于规则的方法，同时使用到了启发式算法。到1996年，命名实体识别作为信息抽取的子任务被正式引入到MUC-6。在MUC-6会议上正式提出了“命名实体”这个概念，并引入了信息抽取研究的评价指标体系，并定义了命名实体包括了：人名（Person）、地名（Location）、机构名（Organization）、日期（Date）、时间（Time）、百分数（Percentage）、货币（Monetary Value）。而MUC-7 [5]定义了，信息抽取包括3种任务：模板元素（Template Element, TE）、模板关系（Template Relation, TR）、脚本模板（Scenario Template, ST）。

命名实体识别的研究领域已取得了很多成果，在基于上述成果的基础上，我们将命名实体识别的方法应用于影视领域，进行影视领域命名实体的识别。

命名实体链接的输入通常为一段文本中的一个实体的提及（mention）[6]，命名实体链接的任务就是要从指定知识库中找到查询实体提及所指代的实体。

[2] http://ccks2016.cn/ccks-ch/tasks/

命名实体链接的任务通常包括两个主要阶段[7]：候选实体生成和候选实体排序。候选实体生成就是在指定知识库中找到可能是查询文本指代的实体。然后在对这些实体提取特征排序，返回最优结果。

3 方法

CCKS 2016 Task One包含两个子任务：影视实体识别（发现）和实体链接任务。因为影视知识库实体较多，容易将本不是影视实体的词条链接为影视实体，将两个任务采用联合学习的方式容易导致过多的错误识别率，因此本次评测我们采用的方法也将二者作为独立任务以流水线方式组合，不将其作为联合任务合并学习。

在影视实体识别任务中，我们主要采用了CRFs序列标注方法作为实体识别的主要方法，通过豆瓣实体知识库、拼音、词性[4]以及深度学习方法抽取特征。我们还采用了SSVMs序列标注方法作为辅助，合并得到新的结果。

在影视实体链接任务中，我们首先采用基于筛（sieved）的层次过滤，得到每个词条的候选实体列表，然后采用排序学习的方法对这些候选实体进行排序得到链接的结果。其总体流程如下图1所示：

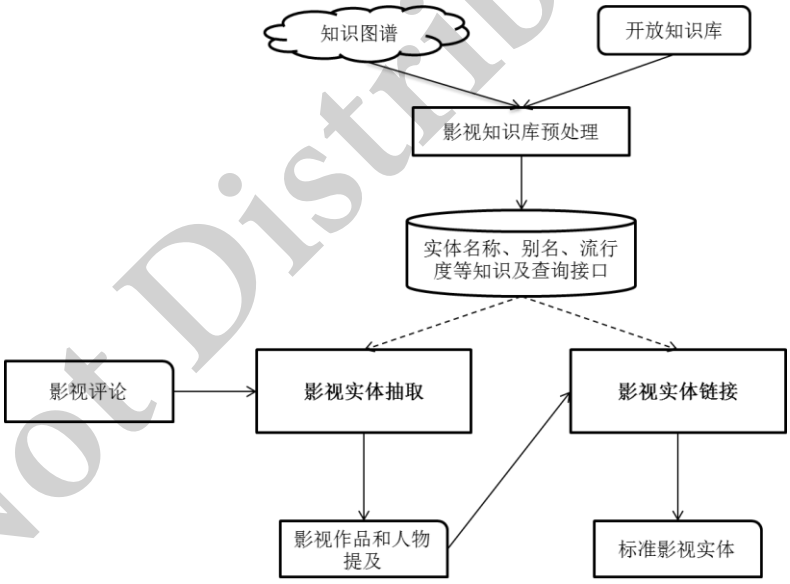


图1 ICRC- DSEDL系统整体架构

首先，本文先对给定影视知识图谱和相关的开放知识库进行处理解析和构建，得到实体名称、别名以及其查询接口，同时利用开放知识库获取实体的流行度等信息。然后采用影视实体抽取模块对影视作品和人物进行抽取，之后对抽取的结果进行对应到给定的影视知识图谱的标准实体上，得到实体识别和链接的结果。

3.1 知识图谱解析与相关知识库构建

本模块将文本化表示的知识图谱解析成规整的关联数据格式，抽取其中的电影名称及与其相关的导演名称、演员名称，构建根据电影名称抽取导演、演员信息的接口。

实体流行度构建，通过豆瓣的开放API接口，获取影视作品的流行度信息，包括影视作品评价数目、影视作品评论数目、影视人物收藏数目。

实体别名库构建，通过豆瓣、百度等开放互联网数据，构建影视作品的别称（如外文名、港台译名、简称等）、构建影视人物的别称（如人物的外文名、中文译名、常见粉丝昵称等）。

3.2 影视实体识别模块方法描述

影视实体识别模块分为以下几个步骤：(1) 预处理和格式转换；(2) 特征提取；(3) 训练和测试；(4) 获得影视作品和任务在评论中的提及。

其中，预处理和格式转换模块完成以下功能：

1) 将文档按字符为单位切割；
2) 将繁体文本转换为简体文本；
3) Tokenization，将不规则的字符标准化；

之后，特征提取模块抽取以下表1所示特征：

表 1 影视实体识别模块特征列表

特征名称	特征类型	特征说明
字符	文本特征	语料数据经过预处理模块处理后的单个字符、上下文字符以及由这些字符组成的 N-gram 字符串 (N<5)
拼音		每一个字符对应的拼音
词边界		字符所在的词的边界
词性		字符所在词的词性
词聚类		采用word embedding方法得到的词向量的聚类特征
句来源		根据句子的来源，用于区分不同类型的评论
句类型		采用RNN学习和判别句子是否可能描述影视作品、影视人物
影视人物	知识特征	是否与知识库中的影视人物词典匹配
影视作品		是否与知识库中的影视作品匹配
相关影视人物		根据评论的标题，获取相关作品的影视人物列表，是否匹配与该列表匹配
相关影视作品		根据评论的标题，获取相关作品的影视作品列表，是否匹配与该列表匹配
别称		是否与知识库中的别称词典匹配
流行度		该影视作品或者影视人物的流行度区间
字符+词边界	组合特征	/
字符+词性		/
词性+词边界		/
字符+句类型		/

其中，特征中的词聚类特征是通过影评数据和知识图谱中的影视知识库介绍文本作为语料来源，采用SkipGram方式进行训练。句类型特征是采用stacked-LSTM对训练集中出现影视作品、影视人物的句子进行学习其分类，输入为评论中任意一个句子，其类标为对应是否有影视作品、影视人物。

对评论文本采用序列标注方法进行序列标注，采用BIOES标注体系，得到标注结果，获得影视作品和人物提及。

3.3 影视实体链接模块方法描述

影视实体链接分为以下两个步骤：(1) 候选实体集合抽取；(2) 候选实体排序。

候选实体集合抽取采用了筛模式的抽取方法，将分以下层次分别抽取，当某一层次能够获得候选实体时，不进行下一层次的候选实体抽取:

1） 完全匹配，对于能够直接匹配知识库（包括全称和别名）中的词条，将所有匹配到的条目作为候选集合
2） 部分匹配，对于能够部分匹配知识库（包括全称和别名）中的词条，将所有匹配到的条目作为候选集合
3） 对于抽取的电影实体，如若以上两级就匹配结果为空，对于知识库中每个电影全称是否包含电影实体中每个字，将所有匹配到的条目作为候选集合。
4） 编辑距离匹配，对抽取实体 mention 长度小于 4 的阈值设置为 1，否则为 2。计算实体 mention 与知识库（包括全称和别名）中的词条的编辑距离，将结果小于阈值的条目作为候选集合。
5） 拼音编辑距离匹配，对抽取实体 mention 长度小于 4 的阈值设置为 1，否则为 2。计算实体 mention 的拼音与知识库（包括全称和别名）的拼音编辑距离，将结果小于阈值的条目作为候选集合。

在获取候选实体集合后，对候选实体与文本中提取的实体进行相关排序，得到最相关的实体，本系统采用排序学习方法进行学习排序规则，其中的采用的特征如下

拼音编辑距离特征：

对于候选实体列表中的每一个实体，分别计算mention与其全称，别名，全称拼音和别名拼音的编辑距离，并返回最小的编辑距离。

1）流行度特征：

在豆瓣网上可以获取每个电影实体的评分人数和每个影人实体的粉丝数。

对于电影实体，每个候选实体的流行度特征计算公式如下：

$$P_m(e) = \frac{\text{Ratings}(e)}{\sum_{k=1}^{n} \text{Ratings}(e_k)}$$

其中$\text{Ratings}(e)$表示e候选实体的评分人数。对于影人实体，每个候选实体的流行度特征计算公式如下：

$$P_h(e) = \frac{\text{Fans}(e)}{\sum_{k=1}^{n} \text{Fans}(e_k)}$$

其中$\text{Fans}(e)$表示候选实体e的影迷人数。

2）基于关键字的相似度特征：每个电影实体，可以在豆瓣网上获取该电影的关键字，如《大白鲨》的关键字列表是“惊悚、 美国、 灾难、 经典、 恐怖、 1975、 剧情、 科幻”，影人实体则以其作品列表作为其关键字。

对于候选实体e对应的关键字列表K的相似度特征计算公式如下:

$$\mathrm{Sim}(e, K) = \frac{\mathrm{counts}(e)}{\mathrm{length}(K)}$$

其中$\mathrm{counts}(e)$代表e实体关键字在评论中出现的次数，$\mathrm{length}(K)$代表e实体对应的关键字列表中关键字的个数。

3）关联特征，抽取评论集合中的电影名称（如果有），根据电影名在知识库中匹配到相应的电影实体和与其相关联的电影主演导演等影人实体。如果候选实体在其中该特征值为1，否则为0。

4 实验及结果分析

影视知识库(Keg-Movie-Ontology)是由清华大学计算机系知识工程实验室构建的完全结构化的双语影视本体,包括23个概念，91个属性，70余万个实体以及1000多万个三元组。本次评测发布的数据是一个子集,仅包含豆瓣的词条。

影视知识库(KMO)共包括以下几个文件：

1) artist：影视人物实体
2) movie：影视作品实体
3) concept.ttl：概念及其上下位关系
4) actornode.ttl：影视作品中的演员信息

通过从影视知识库中进行一系列的数据提取和解析，将文本化表示的知识图谱解析成规整的关联数据格式，抽取其中的电影名称及与其相关的导演名称、演员名称，构建根据电影名称抽取导演、演员信息的接口。

随后通过查阅大量的开放互联网资源和开放API接口进行知识库的构建，这些都为后面的实体识别和实体链接打下了很好的基础。

命名实体识别可以看作是一种序列标注问题，序列标注问题在自然语言处理领域是一类很典型的问题。条件随机场是其中性能较好的模型之一，因而在影视实体识别任务中，我们主要采用了CRFs[8]序列标注方法作为实体识别的主要方法，通过豆瓣实体知识库、拼音、词性以及深度学习方法抽取特征。同时我们还采用了SSVMs[9]序列标注方法作为辅助，合并得到新的结果。

基于已发布的训练集和测试集，我们对实体识别系统进行了实验评测，利用训练数据来训练模型，利用测试数据来检测模型的性能。实体识别采用精确率(Precision)、召回率(Recall)以及 F1-Measure 作为评价指标。其实验结果如下表2所示：

表2 ICRC-DSEDL系统实体识别性能

NED	Precision	Recall	F1-Measure
ICRC-DSEDL	84.24%	69.43%	76.12%

实体链接可以看作是一种排序问题。先对给定实体在知识库中找出对应的候选实体，然后通过计算特征对这些实体进行排序，返回排序最高的那个结

果。SVM-rank[3]是一个常用的学习排序工具，通过编辑距离，拼音编辑距离，实体的流行度以及关联特征来确定候选实体的排序结果。

在实体链接部分，采用精度(Precision)作为评价指标，实验结果如下表3所示：

表3 ICRC-DSEDL系统实体链接性能

EL	Precision
ICRC-DSEDL	86.53%

最后，将命名实体识别和实体链接联合起来，在端对端的层面上，对整个系统做一个综合评价，评价结果如下表4所示：

表4 ICRC-DSEDL系统端对端整体性能

Overall	Precision	Recall	F1-Measure
ICRC-DSEDL	72.90%	60.08%	65.87%

5 结论

本文对影视领域电影评论的命名实体识别和将实体与知识库相链接的任务进行了研究，主要内容包括知识图谱解析与相关知识库构建、影视实体识别和影视实体链接等。首先，对给定的影视知识库进行了一系列的数据提取和解析，将文本化表示的知识图谱解析成规整的关联数据格式。随后通过查阅大量的开放互联网资源和开放API接口进行知识库的构建。然后采用了CRFs序列标注方法作为实体识别的主要方法，通过豆瓣实体知识库、拼音、词性以及深度学习方法抽取特征。同时还采用了SSVMs序列标注方法作为辅助，合并得到新的实体识别结果。最后，将实体识别的结果链接到影视知识库中，我们首先采用基于sieved的层次过滤，得到每个词条的候选实体列表，然后通过SVM Ranking模型，在文本中提取了诸多特征，对候选实体进行排序，最终得到最优实体链接结果。

但由于影视领域的实体识别和链接的研究不同于其他领域包括通用领域的研究，其表现在影视领域中人物和电影名称都存在大量的别名，而且实体名字具有歧义性和变异性，也就是同一个实体名字，有可能指代多个实体；此外，同一个实体可能有多个实体名字与之对应，所以这给实体识别和链接都带来了一定的阻碍。同时也因为训练语料较少的缘故，故此次的结果还没有达到预期的成熟状态，还有待将来进行更加深入的研究。

[3] https://www.cs.cornell.edu/people/tj/svm_light/svm_rank.html

参考文献

[1] C. N. Santos and R. L. Milidiú, “Named entity recognition,” *Entropy Guid. Transform. Learn. Algorithms Appl.*, pp. 51–58, 2012.

[2] 张祝玉, 任飞亮, and 朱靖波, “基于条件随机场的中文命名实体识别特征比较研究[C],” in *见: 第4届全国信息检索与内容安全学术会议论文集*, 2008.

[3] H. Li, “Learning to rank for information retrieval and natural language processing,” *Synth. Lect. Hum. Lang. Technol.*, vol. 7, no. 3, pp. 1–121, 2014.

[4] R. Grishman and B. Sundheim, “Message Understanding Conference-6: A Brief History.,” in *COLING*, 1996, vol. 96, pp. 466–471.

[5] N. Chinchor and E. Marsh, “Muc-7 information extraction task definition,” in *Proceeding of the seventh message understanding conference (MUC-7), Appendices*, 1998, pp. 359–367.

[6] W. Shen, J. Wang, and J. Han, “Entity linking with a knowledge base: Issues, techniques, and solutions,” *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 2, pp. 443–460, 2015.

[7] J. Yuan, Y. Yang, Z. Jia, H. Yin, J. Huang, and J. Zhu, “Entity recognition and linking in Chinese search queries,” in *National CCF Conference on Natural Language Processing and Chinese Computing*, 2015, pp. 507–519.

[8] J. Lafferty, A. McCallum, and F. Pereira, “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data,” *Dep. Pap. CIS*, Jun. 2001.

[9] Y. Altun, I. Tsochantaridis, T. Hofmann, and others, “Hidden markov support vector machines,” in *ICML*, 2003, vol. 3, pp. 3–10.

基于平均互信息量和知识图谱的产品预测

邹震，张昀，刘君艺，周子力
曲阜师范大学物理工程学院， 山东曲阜， 273165
996736369@qq.com 935089344@qq.com 595037416@qq.com zlzhou999@163.com

摘要 ： 随着大数据时代的到来以及其快速的发展趋势，对大数据的分析已经成为一个热门的技术。本文介绍了利用平均互信息量、知识图谱等方法对包含 19046 个数据的训练样本进行分析建模的过程。首先计算出产品对应每一个属性的条件概率，然后针对每个属性取值对属性权值的贡献程度的差别，提出基于平均互信息量的分类模型，然后针对测试结果加入知识图谱模型，通过本次测试任务发布的数据测试后发现，基于平均互信息量和知识图谱的分类模型可以有效地提高分类算法的预测精度和准确率。

关键词： 条件概率；平均互信息量；权重因子；知识图谱

0 引言

大数据是国内外最热门的研究方向之一，研究大数据建立一个合适的模型是必不可少的。我们建立的模型是基于平均互信息量和知识图谱的分类模型，并用该模型进行产品与给定属性的分析匹配，得到产品与不同属性之间的联系，从而达到当给定任务测试数据之外的属性数据时就可以自动匹配出相应的产品类型。该模型的特点就是将知识图谱引入进来，利用知识图谱先将数据范围进行缩小然后进行分类匹配，从而使得模型的精确度在原有的基础上得到了提高同时也大大减少了模型的计算量。

例如：当给定某种产品的五种属性（Enterprise（供应商编码），Destination（买方国家编码）， Price（平均价格），Origin（原产地编码），Custom（通关海关编码））或者其中几种时，该模型就可以通过计算后得出所对应的产品类型，本文在接下来的内容当中，将详细叙述此次评测任务的建模过程。

1 任务分析

要对进出口交易记录数据进行产品判别，其中样本数据的属性均为实体。首先将所给的有类别属性的 18279 条记录作为训练样本，通过数据分析，确定 Enterprise（供应商编码），Destination（买方国家编码）， Price（平均价格），Origin（原产地编码），Custom（通关海关编码）5 个属性对产品类别的影响程度。根据 5 个属性的影响程度及产品类别之间的相关性，判断出其他数据库中给出不同 Enterprise（供应商编码），Destination（买方国家编码），Price（平均价格），Origin（原产地编码），Custom（通关海关编码）所对应的产品类别。

2 解决方案

2.1 初始方案

产品类别属性有 364 个不同值，即｛P001,…,P364｝，因此有 364 个不同的类，设 a_1 对应 P001，而 a_2 对应 P002…以此类推， a_{364} 对应 P364。决策集 A 有 364 个类，$A=\{a_1,a_2,...a_{364}\}$。

2.1.1 步骤：

（1） 首先计算 Enterprise 属性与产品类别之间的条件概率，Enterprise 有 560 种状态，

即｛1102919313, 1105919182,..., 440316946J｝,设 $E=\{e_1,e_2,...,e_{560}\}$。对应条件概率为 $p(a_i \mid e_j)$，其中 $1\le i\le 364$，$1\le j\le 560$。计算得到 560*364 个结果。

(2) 同理求出其他属性与产品类别之间的条件概率，分别为 Destination： $p(a_i \mid d_k)$，$1\le k\le 144$；Price： $p(a_i \mid p_l)$，$1\le l\le 67$；

注：对价格属性进行排序，发现数据具有连续性，因而对其作适当的简化处理。即根据产品数量将价格区间按照升序划分为 67 个部分，每一部分可依次命名为 $p_1,p_2,...,p_{67}$。区分数据记录中的价格具体属于哪一部分。

Origin： $p(a_i \mid o_m)$，$1\le m\le 131$；Custom： $p(a_i \mid c_n)$，$1\le n\le 20$

(3) 假设供应商、买方国家、平均价格、原产地、通关海关对名称类别的影响等价，即权值均为 1。对训练样本中的记录数据，根据

$$p(a_i)=p(a_i \mid e_j)+p(a_i \mid d_k)+p(a_i \mid p_l)+p(a_i \mid o_m)+p(a_i \mid c_n) \qquad (1)$$

计算出 P001，P002，…，P364 对应的值，值越大说明概率越大。按照概率大小进行排序，找出最有可能的 product 类别，从而得到预测结果。

(4) 将预测结果和实际结果进行比较，发现出错率较高。

2.1.2 缺陷：

每个变量对结果的影响是不同的，而笼统的将权值视为 1，认为各变量贡献相等，势必会大大降低结果的正确率。

2.2 改进方案

考虑到变量对结果影响的差异性，根据平均互信息量的大小确定变量对应的权重。

基于互信息量的分类模型，可以充分考虑条件属性（Enterprise，Destination， Price，Origin，Custom）对决策属性（Product）的影响，计算出每个属性的具体取值对权重的影响程度，即权重因子。

2.2.1 步骤：

(1) 根据式

$$I(E;a_i)=\sum_{j=1}^{560} p(e_j a_i)\lg\frac{p(a_i \mid e_j)}{p(a_i)} \qquad (2)$$

可求出Enterprise（供应商编码），对 $1\le i\le 364$ 范围内每一个产品类别的平均互信量。利用同样的方法求出Destination（买方国家编码）， Price（平均价格），Origin（原产地编码），Custom（通关海关编码）这四个属性与产品类别之间的平均互信息量。即: $I(D;a_i)$, $I(P;a_i)$, $I(O;a_i)$, $I(C;a_i)$。

求出的平均互信息量的值反映了属性间的影响程度，且值越小说明影响越小，即相关性越小。根据这个特点可以判断出Enterprise（供应商编码），Destination（买方国家编码），Price（平均价格），Origin（原产地编码），Custom（通关海关编码）这五个属性对产品类别的影响大小。

（2）设产品类别为a_i时五个条件属性对应的权重因子分别为f_i, g_i, x_i, y_i, z_i。根据五个属性与产品类别的平均互信息量进行归一化处理，得到属性权重向量$(f_i, g_i, x_i, y_i, z_i)$。则

$$f_i = \frac{I(E;a_i)}{I(E;a_i)+I(D;a_i)+I(P;a_i)+I(O;a_i)+I(C;a_i)} \quad (3)$$

$$g_i = \frac{I(D;a_i)}{I(E;a_i)+I(D;a_i)+I(P;a_i)+I(O;a_i)+I(C;a_i)} \quad (4)$$

$$x_i = \frac{I(P;a_i)}{I(E;a_i)+I(D;a_i)+I(P;a_i)+I(O;a_i)+I(C;a_i)} \quad (5)$$

$$y_i = \frac{I(O;a_i)}{I(E;a_i)+I(D;a_i)+I(P;a_i)+I(O;a_i)+I(C;a_i)} \quad (6)$$

$$z_i = \frac{I(C;a_i)}{I(E;a_i)+I(D;a_i)+I(P;a_i)+I(O;a_i)+I(C;a_i)} \quad (7)$$

（注：f_i, g_i, x_i, y_i, z_i均为常数。）

3）根据式

$$P(a_i) = f_i p(a_i \mid e_j) + g_i p(a_i \mid d_k) + x_i p(a_i \mid p_l) + y_i p(a_i \mid o_m) + z_i p(a_i \mid c_n) \quad (8)$$

计算出 P001，P002，…，P364 对应的值，计算所得的值越大说明对应是此类产品的概率越大。

然后按照概率大小进行排序，取排序结果的前三个产品类别便可得到预测结果。部分结果见表 2-1:

表 2-1 概率预测结果表

Enterprise	Destination	Price	Origin	Custom	Product	Product1		Product2		Product3	
320496506X	D012	105	OR051	C16	P055	P055	0.1470833	P187	0.1220126	P050	0.1169166
3204960521	D079	173	OR051	C01	P055	P065	0.1679658	P050	0.1551496	P055	0.1459081
3206946104	D079	29.35	OR057	C16	P056	P056	0.2989533	P187	0.1141611	P226	0.1015451
3204960521	D012	22.1	OR051	C16	P056	P065	0.3025068	P187	0.1250381	P095	0.1144981
3206946104	D079	29.35	OR057	C16	P056	P056	0.2989533	P187	0.1141611	P226	0.1015451
3206946104	D079	29.35	OR057	C16	P056	P056	0.2989533	P187	0.1141611	P226	0.1015451
3206946104	D079	29.35	OR057	C16	P056	P056	0.2989533	P187	0.1141611	P226	0.1015451
3707933849	D012	14.3	OR119	C16	P065	P006	0.1690672	P187	0.1424123	P032	0.1212233
3301910048	D034	18.32	OR126	C16	P065	P065	0.380949	P187	0.1202494	P008	0.1043618
5107230024	D012	18.05	OR056	C16	P065	P065	0.4748296	P187	0.1357677	P163	0.0685474

4）通过对比，发现结果虽仍有出错的情况，但准确率相比于之前的测试结果已大大提高。

2.2.2 优势：

基于平均互信息量的样本预测可以预测未知样本的产品类别，计算方法比较简单，并具有较高的预测精度和分类准确率。

2.3 方案优化

知识图谱是结构化的语义知识库，用于以符号形式描述物理世界中的概念及其相互关系．其基本组成单位是“实体－关系－实体”三元组，以及实体及其相关属性－值对，实体间通过关系相互联结，构成网状的知识结构，其旨在描述真实世界中存在的各种实体或概念，能够利用可视化的图谱形象地展示多个实体或概念之间的相互联系。

考虑到知识图谱在这一方面的优点以及实际的情况(一个供应商提供的产品类别是有限的。同样地，某一个买方国家、平均价格、原产地、通关海关对应的产品类别也是有限的)，因而利用知识图谱的概念，构建变量之间的联系，找出变量中共同对应的产品类别，就能够缩小所提供训练样本的范围，减少匹配过程中的计算量，更加重要的是可以提高测试结果的准确性。

例如，验证样本的第一条记录

$e_j = 1301930930, d_k = \mathrm{D110}, p_l = 4.6, o_m = \mathrm{OR028}, c_n = \mathrm{C18}$，在所给训练样本中分别对应的产品类别如表 2-2 所示：

表 2-2 产品类别对应表

Enterprise	1301930930	P185，P292，P184
Destination	D110	P226，P173，P228，P185，P291，P092，P184，P201
Price	4.6	P003，P006，P007，P009，P025，P027，P031，P032， P033，P035，P036，P038，P041，P046，P047，P069，P084，P086，P096，P098，P101，P104，P107，P114，P115，P117，P126，P135，P140，P144，P150，P163， P164，P167，P184，P185，P187，P196，P199，P200， P203，P215，P224，P226，P228，P234
Origin	OR028	P292，P119，P187，P185，P257，P291，P351，P073， P092，P234，P350，P352，P207，P279，P323，P314，P096，P086，P356，P355，P259，P184，P117，P313， P252，P012，P126，P082，P196，P077，P150，P232， P318，P110，P065，P254，P263
Custom	C18	P203，P185，P291，P292，P263，P259，P184，P064， P267
Common Product	P185，P184	

可以看到，第一条记录对应的产品类别只可能是 P185 或 P184，因此只需计算出 $P(a_i = \mathrm{P185})$ 和 $P(a_i = \mathrm{P184})$ 的值，进行比较后即可得出预测结果。

利用该方法简化以后，只需计算个别产品类别的概率，大大减少了运算量。

3 参数调试

（1）根据求出的权重向量 $(f_i, g_i, x_i, y_i, z_i)$，对训练样本进行预测，并将预测结果与实际结果进行比较，发现准确率约为 50.2%。这是由于利用平均互信息量所求的是一个属性对某一产品类别的整体影响，不可能准确适用于每一个属性值，从而产生错误的预测结果。

（2）分析结果出错主要是由哪个属性主导的。通过不断调整该属性的权重因子，实现正确率的提高。

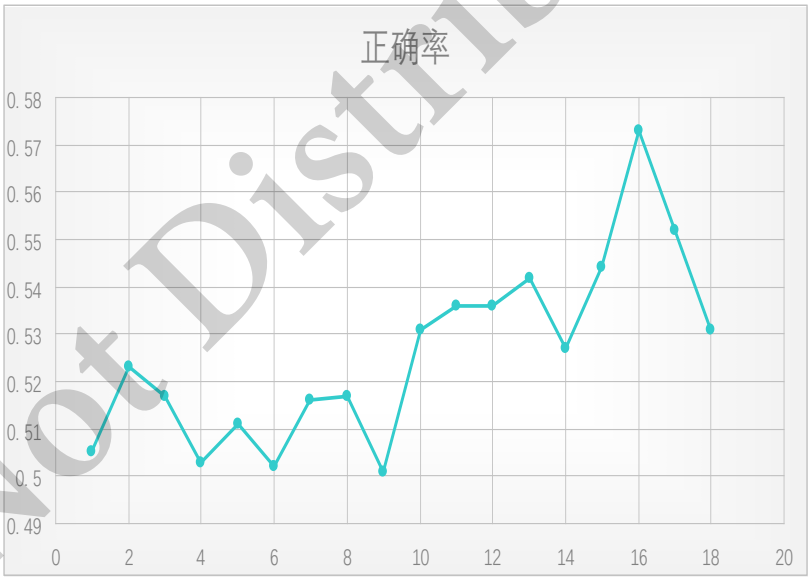
表 3-1 方案预测结果

方案	正确率	f	g	x	y	z

1	0.505	0.352941	0.352941	0.235294	0.029412	0.029412
2	0.523	0.315789	0.315789	0.315789	0.026316	0.026316
3	0.517	0.352941	0.235294	0.352941	0.029412	0.029412
4	0.503	0.32	0.32	0.32	0.013333	0.026667
5	0.511	0.336323	0.336323	0.336323	0.019058	0.028027
6	0.502	0.26087	0.26087	0.434783	0.021739	0.021739
7	0.516	0.336323	0.336323	0.336323	0.028027	0.019058
8	0.517	0.352941	0.235294	0.352941	0.029412	0.029412
9	0.501	0.336323	0.336323	0.336323	0.028027	0.019058
10	0.531	0.307692	0.307692	0.307692	0.025641	0.051282
11	0.536	0.3	0.3	0.3	0.05	0.05
12	0.536	0.352941	0.352941	0.176471	0.058824	0.058824
13	0.542	0.428571	0.214286	0.214286	0.071429	0.071429
14	0.527	0.5	0.083333	0.25	0.083333	0.083333
15	0.544	0.461538	0.153846	0.230769	0.076923	0.076923
16	0.573	0.3	0.2	0.3	0.1	0.1
17	0.552	0.222222	0.222222	0.333333	0.111111	0.111111
18	0.531	0.272727	0.181818	0.272727	0.090909	0.181818

不同方案对应的正确率如图 3-1 所示：

图 3-1 方案正确率折线图



（3）由上图可知，取方案 16 对应的权重因子时，训练样本的准确率最高。故最终选择权重向量（0.3,0.2,0.3,0.1,0.1）对验证样本进行产品名称类别的预测。

4 结论

本文简介了评测任务的基本情况，提出了基于平均互信息量并加入只是图偶的的分类模型，该方法利用知识图谱大大缩小了计算的范围和复杂程度，得到了良好的测试效果。

参考文献

[1] 张震, 胡学钢. 基于互信息量的分类模型[J]. 计算机应用, 2011, 31(6):1678-1680. 第 1679-1680 页.

[2] 刘峤, 李杨, 段宏, 等. 知识图谱构建技术综述[J]. 计算机研究与发展, 2016(3):582-600.

[3] 郭云峰,韩龙,皮立华,等.知识图谱在大数据中的应用[J].电信技术,2015(6):25-29.

基于位置的知识图谱链接预测

张宁豫　陈曦　陈矫彦　陈华钧

浙江大学计算机科学与技术学院
浙江省大数据智能计算重点实验室
{zhangningyu,huanjunsir,xichen,jiaoyanchen}@zju.edu.cn

摘要 链接预测是知识图谱的补全和分析的基础。由于位置相关的实体和关系本身拥有丰富的位置特征，本文提出了一种基于位置的知识图谱链接预测方法。该方法首先通过分析实体和关系的语义特征对关系进行分类，然后提出了一种基于位置的实体和关系的位置特征和规则的挖掘方法；其次，通过挖掘出的实体位置特征和规则，对实体和关系的向量化方法的预测结果进行约束，得到最终的结果。本文通过对WikiData、FB和WN数据集的实验，证明本方法针对基于位置的关系和实体链接预测拥有较好的效果。

关键词 位置特征，知识图谱，链接预测，知识图谱补全

1　引言

知识图谱例如FreeBase、YAGO等是很多人工智能应用的重要数据来源。它包含了海量的实体和关系以三元组的形式进行存储。然而，大多数知识库的数据都是缺失的。所以知识库补全，也就是从现有的知识库进行链接预测新的关系和实体是一项重要的工作。

现有的知识图谱链接预测方法大多都是直接利用实体和关系本身或图的特征来进行链接预测。对于给定的知识图谱，实体和关系通常会被映射成低维的向量。通过定义一个打分函数来对每一对实体和关系的三元组进行预测。实体和关系的向量可以通过最大化已知正确三元组的打分函数来训练获得。然而，在训练实体和关系向量和打分函数的过程中，这类方法并没有利用实体和关系本身隐藏的位置特征。此外,由于实体和关系的向量化方法的数据驱动特点，如果训练结果中某一关系或者实体数据量很小，训练出的这一关系或实体的向量针对打分函数可能会造成过拟合等问题。

事实上，现有的知识库中储存着海量的位置相关的实体和关系。例如，在三元组（鲁迅，WasBornIn，绍兴）中，实体“绍兴”有明确的位置特征。利用实体”绍兴“的属性可以获得位置特征，进而可以推测实体“鲁迅”的隐含

的位置特征，利用位置的隐含特征构造规则约束。例如在判断三元组（鲁迅，WasBornIn，浙江）是否成立时，利用实体“鲁迅”的位置特征和空间位置的规则判断，可以约束判断的最终结果。

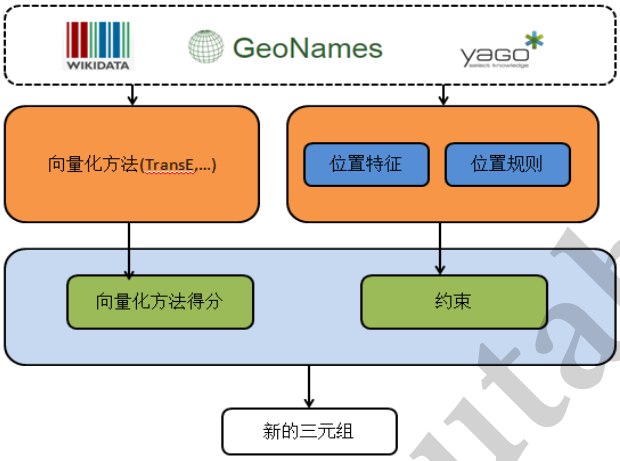


图 1. 基于位置的向量化和规则链接预测方法。

在本文中, 我们提出了一种针对位置关系的基于向量化和规则的链接预测方法。位置相关的关系指的是三元组中至少含有一个实体,其属性或者本身含义带有位置的特点。例如，至少有一个实体是一个地名、一个区域名称、兴趣点名称等。首先，针对基于位置的三元组，我们根据其特点把基于位置的关系分成了三类：包含关系、相邻关系、相交关系。包含关系意思是两个实体本身的地理坐标范围是相互包含的，例如LoactedIn。相邻关系是指两个实体本身地理坐标范围是相互分离的但在一定距离内，例如NearBy。相交关系是指两个实体本身地理坐标范围是相互交叉的，例如HasSameHometown。针对不同的实体，我们提取出不同的隐藏的位置特征。针对不同的关系类型，我们提取不同的规则。实体的隐藏位置特征主要由实体本身的位置如经纬度或地名和它的辐射范围组成。规则主要分成两类：一类是通用的规则。例如两个实体间拥有NearBy 关系必然会存在HasNeighbour 关系，同时NearBy 关系的实体必须是属于Location类型的。另一类是位置规则。例如实体h 和实体t的隐藏位置特征是后者包含前者，则两个实体间有可能存在包含这类的关系。最后，我们利用规则对向量化方法结果进行约束，得到最终的结果如图1所示。我们的方法有以下优点：1）规则的使用降低了计算空间和提高了准确度2）保留了向量化方法的优点，同时

加入了隐藏的位置信息3）它是一个通用的框架，能够试用各种通用的向量化方法和规则。

综上所述，本文的贡献如下: 1)针对基于位置的三元组，我们提出了挖掘实体和关系位置特征的方法。2)我们提出了一种针对位置关系的基于向量化和规则的链接预测方法。3)我们利用WikiData、FB和WN的数据集进行实验，证明针对位置相关的链接预测，本方法比其他方法准确度有所提高。

2 相关工作

知识图谱的链接预测通常是指给定一组三元组，预测其成立的可能性。根据Nickel Maximilian[11]的研究，知识图谱链接预测通常分为三大类：1）通过实体和关系的隐含特征将其转换成低维向量的方法[12][3]；2)基于图特征的方法[8][5]；3) 基于马尔科夫概率图利用一阶谓词逻辑[6]或者软逻辑(Probabilistic Soft Logic)[13] 来预测。

基于向量化的知识图谱链接预测方法的核心是用向量来表达实体和关系隐藏的特征。RESCAL[12]和TransE[2]是两个典型的方法。他们通过最小化结构风险或边界误差来学习隐藏的向量。然而，在学习和预测的过程中，这类方法都没有利用潜在的位置特征和应用规则。TRESCAL[4]将规则和RESCAL整合在了一起，但他仅能使用单一规则（例如某种关系的实体必须是特定的类型）。Rockt ¨ aschel et al.[14]提出了将一阶谓词逻辑映射成低维向量。但是他们的方法中规则并没有直接起到链接预测的作用，也没有降低预测的复杂度。Wang Quan[16]提出了一种基于整数线性规划(ILP)的方法将向量化结果和规则整合起来进行链接预测。但是他们并没有利用潜在的位置特征和基于位置的规则。基于图的方法核心是挖掘知识图谱图结构所有的特征。Linyuan[9]挖掘节点之间的相似度来进行链接预测。Path Ranking Algorithm（PRA）[7]是利用节点之间不同通路包含的特征来进行预测，也可以提炼出规则来约束结果。但是，基于图的特征的方法通常很适合局部的链接预测，不一定能挖掘出全局的隐藏特征。我们的方法的不同点在于我们提供了一个通用的利用位置特征和规则的预测框架，可以整合各种向量化方法和规则。

在马尔科夫网络中，规则已经被大量使用，代表性的研究有利用一阶谓词逻辑[6]和软逻辑(Probabilistic Soft Logic)[13]。本文中，我们利用规则来约束向量化方法的结果，将整合问题变成一个整数规划问题。此外，我们挖掘出了隐藏的位置特征，构造了位置特征的规则。

3 方法

3.1 定义

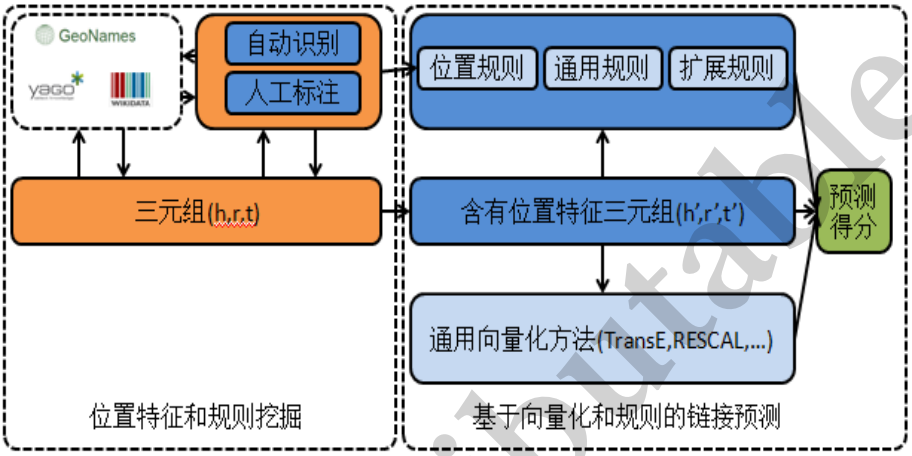


图 2. 系统框架。

定义1(实体位置特征):如果实体e能够在当前知识库或外部数据库如Yago, geoname, linkgeodata, Wikidata中匹配到相应的位置（经纬度）和大致范围或所属上级的范围，则e有位置特征f_e=[lng,lat,D]，lng是经度，lat是维度，D是一个描述实体包含范围的数值，通常情况由实体本身的行政地域半径或上级所属区域半径最小值确定。

定义2(位置相关三元组):三元组(h,r,t)的实体h,t中至少有一个实体含有位置特征。

定义3(包含关系):实体h 和t的位置特征存在$\sqrt{|(h_{lng}-t_{lng})^2-(h_{lat}-t_{lat})^2|} < |h_D - t_D|$ 则两者存在包含关系$HasContain(h,t)$。

定义4(相邻关系):实体h 和t的位置特征存在$\sqrt{|(h_{lng}-t_{lng})^2-(h_{lat}-t_{lat})^2|} \geq |h_D + t_D|$ 则两者存在相邻关系$HasAdjacent(h,t)$。

定义5(相交关系):实体h 和t的位置特征存在$|h_D - t_D| \leq \sqrt{|(h_{lng}-t_{lng})^2-(h_{lat}-t_{lat})^2|} < |h_D + t_D|$ 则两者存在相交关系$HasIntersect(h,t)$。

3.2 框架

如图2所示，我们的系统由两部分组成。1）位置特征和规则挖掘。这部分首先对三元组中实体进行位置特征提取，然后对基于位置的三元组的关系进行自动识别或者人工标注分类，最后提取出其他可能存在的位置特征和规则。2）基于向量化和规则的链接预测。这部分首先对三元组利用向量化方法进行训练，然后利用规则对结果进行约束。

3.3 隐含的位置特征和规则挖掘

给定一个基于位置的三元组(h, r, t), 首先我们需要提取出三元组中实体可以直接获得的位置特征。例如三元组（鲁迅，WasBornIn, 绍兴）中，通过对实体“鲁迅”和“绍兴”的类型和本地数据库以及外部数据库Yago, geoname, linkgeodata, Wikidata的匹配得到，实体“绍兴”是一个地名。我们可以获得该实体的经纬度、面积、相邻城市等信息。通过近似计算（利用面积或相邻区域经纬度），我们可以获得实体"绍兴"的位置特征。然后我们需要获得关系"WasBornIn"的类别, 即它属于包含、相邻、相交哪类。一般地说，有两种做法。1）自动识别。遍历所有三元组中两个实体都含有位置特征的三元组，通过反向计算实体位置特征的差异，推导出此三元组拥有的关系。对常见的如locatedin, nearby等关系，此方法可以方便的判别。2）人工标注。事实上，基于位置的关系总数并不多，再者，通常整个知识图谱需要预测的关系的数量级也不是很大，远小于实体个数的数量级。所以可以采取人工标注的方法来解决额外的关系的分类。最后，我们通过已经获得的关系"WasBornIn"属于包含关系，判断实体“鲁迅”隐藏位置特征，该特征和实体“绍兴”的位置特征存在包含关系。这个知识可以作为规则为后续的未知链接预测做约束。

具体的说, 对于任意三元组(h, r, t), 如果只有实体t可以直接获得位置特征f_t=$[t_{lng}, t_{lat}, t_D]$，根据关系r我们可以推测实体h隐含的位置特征。如果r属于包含关系，则h可能存在隐含位置特征$[t_{lng}, t_{lat}, t_D - \mu]$，其中$0 < \mu < t_D$。如果$r$属于相交关系，则$h$可能存在隐含位置特征$[h_{lng}, h_{lat}, h_D]$ ，其中$|h_D - t_D| \leq \sqrt{|(h_{lng} - t_{lng})^2 - (h_{lat} - t_{lat})^2|} < |h_D + t_D|$，也就说是$h$ 位于一个环状区域范围内。如果r属于相邻关系，则h可能存在隐含位置特征$[h_{lng}, h_{lat}, h_D]$ ，其中$\sqrt{|(h_{lng} - t_{lng})^2 - (h_{lat} - t_{lat})^2|} \geq |h_D + t_D|$。反之，如果实体$h$含有隐藏位置特征来推到$t$也是如此。事实上，对于相交和相邻关系，大多数三元组的两个实体都本身可以直接获取位置关系。以上的隐藏特征都是近似特征。

由此，我们可以获得海量的实体隐藏的位置特征和规则。事实上，我们可以获得以下规则[16]：

规则1(实体类型匹配):特定的关系拥有特定类型的实体。例如，关系LocatedIn拥有的两个实体一定是Location 类型的；关系WasBornIn的两个实体一定是一个是Person类型，一个是Location类型。

规则2(参数个数匹配):一对多和多对一的关系中特定实体的数目有一定限制。例如CityLocatedInCountry是一个多对一的关系。给定一个城市实体，在知识图谱中最多存在一个国家实体与之对应。

规则3(相似关系匹配):如果关系r_1和r_2存在一定的牵连或同属于同一个类型(同是包含类型)同时不违背规则1，2的前提下，则拥有r_1 关系的实体可能存在r_2 关系。例如, $CityCapitalOfCountry->CityLocatedInCountry$。

规则4(位置包含关系):如果两个实体的位置特征存在包含关系，则两个实体可能存在包含关系。例如，实体"鲁迅"和实体"浙江"的位置关系存在包含关系，则两个实体很大程度上存在包含关系。

规则5(位置相邻关系):如果两个实体的位置特征存在相邻关系，则两个实体可能存在相邻关系。例如，实体“西湖”和实体“浙江大学”的位置关系存在相邻关系，则两个实体很大程度上存在相邻关系。

规则6(位置相交关系):如果两个实体的位置特征存在相交关系，则两个实体可能存在相交关系。例如，实体“金庸”和实体“徐志摩”的潜在的位置特征存在相交关系，则两个实体可能存在相交关系。

规则7(位置包含传导):如果实体e_2的位置特征包含实体e_1的位置特征，实体e_3的位置特征包含实体e_2的位置特征，则实体e_3和e_1存在包含关系。包含关系可以一直连续传递，相邻和相交关系不能传递。例如实体“鲁迅”和实体“浙江”存在包含关系，实体“浙江”和实体“中国”存在包含关系，则实体“鲁迅”和实体“中国”存在包含关系。

此外，如果未知的一对一关系的三元组中，其中一个实体和关系存在于已知三元组正样本中，那这个三元组很可能是不成立的。

3.4 基于向量化和规则的链接预测

给定一个知识图谱，其包含n个实体，m个关系。我们可以获得三元组集合$O=\{h,r,t\}$。向量化方法的目的在于：1）通过隐含的特征把实体和关系映射到一个向量2）利用训练好的向量来预测新三元组成立的可能性。本文中。我们利用了三种成熟的向量化方法：RESCAL[12],TRESCAL[4]，TransE[2]。

RESCAL讲每个实体e_i看做一个向量$e_i \in R^d$，每个关系r_k都是一个矩阵$R_k \in R^{d\times d}$。给定三元组$e_i r_k e_j$，它打的打分函数是：

$f(e_i, r_k, e_j) = e_i^T R_k e_j$

$\{e\}$ 和$\{r_k\}$是通过最小化下面的结构损失函数来获得的：

$$\min_{\{e_i\},\{R_k\}} \sum_k \sum_i \sum_j (y_{ij}^{(k)}) - f(e_i, r_k, e_j))^2 + \lambda R$$

其中如果三元组(e_i, r_k, e_j)成立则$y_{ij}^{(k)}$等于1，反之为0。R是正则项。

TRESCAL是RESCAL算法的一个扩展，需要对给定关系的实体的类型进行约束。例如，给定关系r_k和分别包含特定类型的实体的集合H_k,T_k。则问题变成优化问题如下：

$$\min_{\{e_i\},\{R_k\}} \sum_k \sum i \in H_k \sum_{j \in T_k} (y_{ij}^{(k)}) - f(e_i, r_k, e_j))^2 + \lambda R$$

TransE将三元组(e_i, r_k, e_j)映射称以下的三个向量$e_i, r_k, e_j \in R^d$，它是使用了以下的打分函数来计算三元组成立的可能性：

$$f(e_i, r_k, e_j) = ||e_i + r_k - e_j||$$

其中$\{e_i\}, \{r_k\}$ 是通过优化以下边缘损失函数（正确样本得到更高的得分，错误样本得分更低）来得到：

$$\min_{\{e_i\},\{r_k\}} \sum_{t^+ \in O} \sum_{t\_ \in N} [\gamma - f(e_i, r_k, e_j) + f(e_i^{'}, r_k, e_j^{'})]_+$$

其中t^+是正样本，N是负样本集合，由正样本中替换掉实体来构建。在替换掉过程中我们未采用随机替换，而是替换之后确保新的三元组在原始的数据集中存在确定的关系，但关系不是r_k, 这很大程度上确保了样本是负样本。我们利用随机梯度下降的方法来求解优化问题。

利用上述方法，对为止的三元组，打分高的一般情况下成立的可能性较高，反之较低。我们将向量化方法得分的输出记为$y_{ij}^{(k)} = f(e_i, r_k, e_j)$，每个实体的位置特征记为$f_i, f_j$，标记相交关系集合$R_{intersect}$ ，含三元组s 对，相邻关系集合$R_{adjacent}$，含三元组p 对，包含关系集合$R_{contain}$, 含三元组q 对，标记一对多、多对一、一对一关系集合$R_{1-M}, R_{M-1}, R_{1-1}$, 标记特定关系所属实体种类的集合H_k, T_k 。用逻辑变量$x_{ij}^{(k)}$ 来标记这个三元组成立的最终可能。根据[16]我们把规则约束向量化结果的问题定义为一个整数规划的问题如下：

$$\max_{x_{ij}^{(k)}} \sum_k \sum_j \sum_j y_{ij}^{(k)} x_{ij}^{(k)}$$

$$s.t. R1. x_{ij}^{(k)} = 0, \forall k, \forall i \notin H_k, \forall j \notin T_k,$$

$$R2. \sum_i x_{ij}^{(k)} \leq 1, \forall k \in R_{1-M}, \forall j; \sum_i y_{ij}^{(k)} \leq 1, \forall k \in R_{M-1}, \forall i;$$

$$\sum_i y_{ij}^{(k)}, \sum_j y_{ij}^{(k)} \leq 1, \forall k \in R_{1-1}, \forall i, \forall j;$$

$$R3. x_{ij}^{(k_1)} \leq x_{ij}^{(k_2)}, \forall r_{k_1} -> r_{k_2}, r_{k_1}, r_{k_2} \in R_{contain}, r_{k_1}, r_{k_2} \in R_{adjacent}, r_{k_1}, r_{k_2} \in R_{intersect}, \forall i, j,$$

$$R4. \sum_k y_{ij}^{(k)} \geq q\delta_1, \forall k \in R_{contain}, \forall f_i, f_j HasContain(e_i, e_j)$$

$$R5. \sum_k y_{ij}^{(k)} \geq p\delta_2, \forall k \in R_{adjacent}, \forall f_i, f_j HasAdjacent(e_i, e_j)$$

$$R6. \sum_k y_{ij}^{(k)} \geq s\delta_3, \forall k \in R_{intersect}, \forall f_i, f_j HasIntersect(e_i, e_j)$$

$$R7. x_{it}^{(k)} \leq x_{ij}^{(k)}, \forall k \in R_{contain}, \forall f_i, f_t HasContain(e_i, e_t), \forall f_t, f_j HasContain(e_t, e_j),$$

$$R8. x_{ij}^{(k)} = 0, \forall i, k \in O, \forall j \notin O, \forall k \in R_{1-1}$$

其中$x_{ij}^{(k)} \in \{0,1\}, \forall i, j, k$，$O$是正样本集合。通过解上述问题[1]求得最终的得分$x_{ij}^{(k)}$。我们的方法优势如下:1) 在向量化方法的前提下，利用位置和通用规则对含有显性和隐性位置特征的三元组链接预测准确率有明显的提高。2）这是一个通用的框架，向量化方法和规则都可以灵活变化。

4 实验

实验的具体流程如下：1）位置特征和规则挖掘2）基于向量化和规则的链接预测。3）分析位置特征和规则对结果的影响。

4.1 数据集

在实验中我们使用了三个数据集:WikiData-500K,WN-100K,FB-500K分别从Wikidata[15]、WordNet[10]、FreeBase[1] 获取。WikiData是目前较大的一个开放的知识图谱。WikiData包含有human、taxon、administrative territorial、architectural structure、event、chemical compound film thoroughfare astronomical object等类型的实体组成的三元组信息。具我们统计有至少19.8% 的三元

[1] R8.的条件也可以写成$\forall j, k \in O, \forall i \notin O$

组中至少有一个实体含有位置信息（事件、行政区划、地点等）[2],可以直接通过API获取。我们由此构建了WikiData-500K数据集。WN-100K 和FB-500K都是由不同学者发布出的三元组数据集。我们从WN-100K,FB-5000K 筛选出位置相关的三元组来进行训练。此外，我们还利用Yago[3], geoname[4], linkgeodata[5], Wikidata对所有的数据中的实体进行位置信息匹配，以获得实体本身的位置特征。我们过滤了数据集中出现次数少于3次的实体。我们采用了[2]的方法来判断实体的关系是否是一对多还是多对一来制定规则。此外，我们制定了一些同类匹配的规则。数据集如表1 所示。

表 1. 数据集

数据集	实体	关系
WikiData-500K	位置相关实体14,561	231
	所有实体15,321	231
WN-100K	位置相关实体5,325	11
	所有实体38,696	11
FB-500K	位置相关实体5,612	1,345
	所有实体14,951	1,345

4.2 特征和规则挖掘

我们的任务是提取出实体隐含的位置特征。首先，我们对数据集中所有的实体进行位置信息匹配。利用外部数据集拥有的准确地理位置信息匹配数据集中实体。大约40%的实体能匹配到准确的位置特征。然后，我们对数据集中拥有的关系进行分类。我们利用自动分类方法标记了大约63%的关系，剩下的关系采用人工标记的办法。事实上，有大约5%的关系是有歧义的，我们将他们默认归到包含关系类。最后我们利用位置特征和关系的类型挖掘剩下的实体隐藏位置特征。

[2] www.wikidata.org/wiki/Wikidata:Statistics
[3] www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/
[4] www.geonames.org/
[5] www.linkedgeodata.org

4.3 链接预测

我们的任务是补全位置相关的三元组(h,r,t)，也就是说，给定h和t预测r;或者给定h 和r预测t;或者给定r,t预测h。在本节中，我们测试了RESCAL、TRESCAL、TransE。我们把利用基于位置的规则来约束向量化结果的方法命名成l-RESCAL、l-TRESCAL、l-TransE。

对每个数据集，我们把基于位置的三元组按照4:1的比例划分成训练集和测试集。对没一个实体我们都获得其所属类型。对于测试三元组，我们通过计算命中@1(正确命中结果排第一所占的比例)来衡量。在具体实验中，RESCAL，TRESCAL的正则化参数$\lambda = 0.1$，我们迭代训练了十次。在向量化训练过程中，我们将维度分别设置成10,20,50,100来选择最优的参数。然后我们利用利用集成学习的方法获得三种向量化方法获得的最优结果。在规则约束的过程中，$\delta_1 = 0.7, \delta_2 = 0.6, \delta_3 = 0.4$，我们使用lp solve[6]来解整数规划问题。我们对规则约束重复进行了20 次取平均值，以获得最优的结果。

在表格3中，我们展示了不同数据集下不同关系进行关系预测的结果。从结果可以看出，利用基于位置的规则方法对特定的关系有显著的提高。RESCAL和TRESCAL的提升幅度比TransE要高。

表 2. 位置相关关系命中@10(%)结果

关系	RESCAL	l-RESCAL	TRESCAL	l-TRESCAL	TransE	l-TransE
CityLocatedInState	56.1	**67.1**	57.3	**59.3**	55.9	**58.4**
CityLocatedInCountry	61.3	**66.5**	62.4	**62.8**	63.5	**64.1**
CityCapitalOfCountry	45.3	**46.1**	47.2	**47.5**	46.1	**46.4**
NearBy	34.3	**35.2**	35.2	30.2	34.2	**35.3**
WasBornIn	61.3	**65.5**	63.2	60.2	63.2	**65.3**
HasSameHometown	45.5	**45.6**	44.2	40.2	44.2	**45.3**
总平均值	55.2	**61.2**	56.5	**61.7**	57.5	**59.9**

4.4 位置特征和规则分析

我们还对不同关系类型和不同实体进行了结果的比较如表2。从结果可以看出，对我们的方法，包含关系获得的提升程度较高，其次是相邻关系和相交关系。事实上，包含关系的位置隐含特征区域较为狭小，因此对关系的确定限

[6] http://lpsolve.sourceforge.net/5.5/

制较大，可以获得较好的结果；而相邻关系和相交关系（实体都可以直接获得位置特征除外）获取的隐藏位置区域较大，因此限制较为不准确。对实体而言，两个实体都可以直接获得位置关系的预测结果提升幅度最大，其次是单一实体的结果。有趣的是，对于两个都不能直接获得位置信息的实体，本方法仍能获得少量的提升。事实上，例如判断三元组(徐志摩，HasSameHometown，金庸)时候，实体“徐志摩“和“金庸”的隐藏位置特征是可以获得的, 利用人工标记关系”HasSameHometown”为相交关系，使用我们的方法会发现可以获得准确度的提升。

表 3. 不同类型关系命中@10(%)结果

关系和实体	RESCAL	l-RESCAL	TRESCAL	l-TRESCAL	TransE	l-TransE
包含关系均值	55.3	**60.1**	56.1	**57.2**	55.2	**56.8**
相邻关系均值	35.3	**37.9**	35.5	**37.8**	34.5	**35.4**
相交关系均值	41.2	**42.2**	40.2	39.2	39.8	**40.5**
两个实体含位置	57.2	**78.2**	55.6	**70.3**	60.2	**70.0**
单个实体含位置	55.8	**60.2**	50.2	49.2	48.5	**50.3**
实体都不含位置	48.4	**50.2**	49.6	**49.9**	51.6	49.3

5 结论

在本文中, 我们提出了一种针对位置关系的基于向量化和规则的链接预测方法。实体位置特征和规则的使用降低了计算空间和提高了基于位置链接预测的准确度。我们还对位置特征和规则进行了实验分析。实验结果证明，对于特定类型的关系，位置特征和规则的利用可以使链接预测的准确度得到一定程度的提高。将来，我们计划1）分布式我们的方法使得它能够适用于更大的数据集2）加入更加复杂的空间规则3）尝试着在向量化训练的同时直接利用规则，可能它会提高准确度。

参考文献

1. K. Bollacker, R. Cook, and P. Tufts. Freebase: A shared database of structured general human knowledge. In *AAAI*, volume 7, pages 1962–1963, 2007.
2. A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pages 2787–2795, 2013.

3. A. Bordes, J. Weston, R. Collobert, and Y. Bengio. Learning structured embeddings of knowledge bases. In *Conference on Artificial Intelligence*, number EPFL-CONF-192344, 2011.
4. K.-W. Chang, W.-t. Yih, B. Yang, and C. Meek. Typed tensor decomposition of knowledge bases for relation extraction. In *EMNLP*, pages 1568–1579, 2014.
5. X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610. ACM, 2014.
6. S. Jiang, D. Lowd, and D. Dou. Learning to refine an automatically extracted knowledge base using markov logic. In *2012 IEEE 12th International Conference on Data Mining*, pages 912–917. IEEE, 2012.
7. N. Lao and W. W. Cohen. Relational retrieval using a combination of path-constrained random walks. *Machine learning*, 81(1):53–67, 2010.
8. N. Lao, T. Mitchell, and W. W. Cohen. Random walk inference and learning in a large scale knowledge base. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 529–539. Association for Computational Linguistics, 2011.
9. L. Lü and T. Zhou. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, 390(6):1150–1170, 2011.
10. G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.
11. M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich. A review of relational machine learning for knowledge graphs. *arXiv preprint arXiv:1503.00759*, 2015.
12. M. Nickel, V. Tresp, and H.-P. Kriegel. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 809–816, 2011.
13. J. Pujara, H. Miao, L. Getoor, and W. Cohen. Knowledge graph identification. In *International Semantic Web Conference*, pages 542–557. Springer, 2013.
14. T. Rocktäschel, M. Bosnjak, S. Singh, and S. Riedel. Low-dimensional embeddings of logic. In *Proceedings of the ACL 2014 Workshop on Semantic Parsing*, pages 45–49, 2014.
15. D. Vrandečić and M. Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.
16. Q. Wang, B. Wang, and L. Guo. Knowledge base completion using embeddings and rules. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pages 1859–1865, 2015.

基于空间投影和关系路径的地理知识图谱表示学习

段鹏飞[a, b], 王远[c], 熊盛武[a, b], 毛晶晶[a]

a. 武汉理工大学 计算机科学与技术学院，武汉 430070

b. 交通物联网湖北省重点实验室，武汉 430070

c. 南京大学 计算机科学与技术系，南京 210023

duanpf@whut.edu.cn

摘要：近年来，类人智能技术和相关产品飞速发展，这在很大程度上得益于完备知识图谱的构建，特别是以地理为代表的基础教育知识图谱。传统的知识图谱采用网络知识组织形式进行表示，计算复杂度较高，而且三元组的知识表示形式不能有效地度量和利用实体间语义关联关系。本文构建了基于深度学习的知识表示学习算法—PTransW（Path-based TransE and Considering Relation Type by Weight）模型，该模型结合空间投影和关系路径来对翻译模型进行扩展，并加入关系类型的语义信息进行改进。最后，在FB15K数据集和GEOGRAPHY数据集上训练并做链接预测实验。实验结果表明，PTransW模型对复杂关系的建模能力取得了较大地提升；对于规模较小的数据集，复杂度低的TransE和TransR模型将会训练的更充分；但是PTransE和PTransW模型由于利用了关系路径和反向关系中的语义信息，在关系预测方面有很大的优势。

关键词：翻译模型，地理知识图谱，知识表示学习

随着大数据技术的发展，并得益于Linking Open Data等公共数据集项目的展开，互联网也从文档万维网向数据万维网发展。在此背景下，Google为了改善搜索结果，于2012年重新提出了知识图谱（Knowledge Graph）[1]。随后，其它搜索引擎公司也开始构建知识图谱。例如，国内搜狗提出的“知立方”和百度的“知心”。知识图谱除了应用在搜索引擎中，还是自动问答等智能应用的基础，例如，IBM公司开发的Watson系统和日本的高考机器人Todai Robot。

传统的知识图谱一般是采用<实体1，关系，实体2>三元组的方式来表示知识，该种方法可以较好的表示事实性知识，但对很多模糊知识和复杂形式知识，三元组表现出了能力不足。以地理知识图谱为代表的特定领域知识图谱，实体间往往有很强的语义关联，以网络形式来组织知识图谱中的知识，当进行知识推理、知识融合的时候需要设计特定的图算法，计算效率低；而且三元组的知识表示形式无法有效地度量和利用实体间的语义关联关系。

基金项目：国家863项目(2015AA015403),湖北省科技支撑计划(2014BAA146),交通物联网技术湖北省重点实验室(2015III015-B03)

以深度学习[2]为代表的表示学习[3]最近在自然语言处理、图像分析和语音识别等领域取得极大进展。在自然语言处理方面，基于深度学习的词向量表示模型—word2vec模型[4]的提出，掀起了对知识进行表示学习的研究热潮。其中，最引人注目的要属（Bordes A, et al. 2013）受到word2vec模型中的词向量在语义空间的平移不变现象的启发而提出的TransE模型[5]。TransE模型由于其在构建大规模知识图谱时表现出了简单、高效等特点，自提出以来，许多研究者都尝试在TransE模型模型的基础上做进一步的扩展和应用。例如，(Lin, et al. 2015)等人提出的TransR模型[6]，基于空间投影来对TransE进行扩展，提高复杂关系建模能力；以及 (Lin, et al. 2015)提出了PTransE模型[7]，基于关系路径对TransE进行扩展，用来对知识图谱中的关系进行推理。

1 翻译模型及其扩展

1.1 TransE模型

以TransE模型为代表的知识表示学习模型已经在实体链接、关系抽取和知识推理等知识图谱应用中，取得了瞩目的效果[8]。TransE模型将知识图谱中的关系看作是在语义空间中实体间的平移向量[4]。

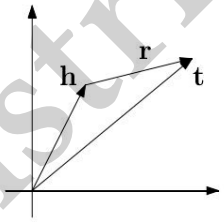


图1 TransE模型简单原理图

对于三元组 (h,r,t)，TransE模型将关系 r 定义为一个平移向量 $\boldsymbol{r}\in R^k$（k 为语义空间维度），嵌入到语义空间的实体向量 $\boldsymbol{h},\boldsymbol{t}$ 可以通过关系向量 $\boldsymbol{r}$ 连接。TransE模型的损失函数定义为：

$$f_r(\boldsymbol{h},\boldsymbol{t})=\|\boldsymbol{h}+\boldsymbol{r}-\boldsymbol{t}\|_{L_1/L_2} \quad (1)$$

TransE模型参数少、复杂度低并且在构建大规模知识图谱中表现出了简单、高效的特点。但是，也正是由于TransE模型的简单，导致了它在复杂关系建模、多源信息融合、关系路径建模等方面的局限性。

1.2 基于空间投影的翻译模型

针对TransE模型在处理知识图谱中复杂关系能力缺失的问题，（Lin et al. 2015）等人提出了TransR模型[6]，将实体看成多种属性的综合，不同关系将专注不同的属性，而将实体和关系嵌入到同一语义空间无法表示该种联系。TransR模型提出将实体、关系分别嵌入实体空间 R^m 、关系空间 R^n（ m,n 均表示空间的维度，并且在TransR模型中 m 和 n 可以相同）。

对于每一个三元组 (h,r,t)，TransR模型设置实体向量 $\boldsymbol{h},\boldsymbol{t}\in R^m$，关系向量 $\boldsymbol{r}\in R^n$。先将位于实体空间 R^m 的头、尾实体，通过投射矩阵 $\boldsymbol{M}_r(\boldsymbol{M}_r\in R^{m\times n})$ 投射到关系空间 R^n，得到位于关系空间的头实体 $\boldsymbol{h}_r$ 、尾实体 $\boldsymbol{t}_r$。然后在关系空间中平移，使 $\boldsymbol{h}_r+\boldsymbol{r}\approx\boldsymbol{t}_r$。

其中，被投射到关系空间中得到的实体向量被定义为：

$$\boldsymbol{h}_r=\boldsymbol{h}\boldsymbol{M}_r,\ \boldsymbol{t}_r=\boldsymbol{t}\boldsymbol{M}_r \tag{2}$$

相应的损失函数被定义为：

$$f_r(h,t)=\|\boldsymbol{h}_r+\boldsymbol{r}-\boldsymbol{t}_r\|_{L_1/L_2} \tag{3}$$

TransR模型采用空间投影在TransE模型的基础上进行扩展，使模型处理复杂关系的能力得到显著提高。

1.3 基于关系路径的翻译模型

TransE和TransR等传统的知识表示学习算法只考虑头、尾实体对之间直接的关系，但是大规模知识图谱中的实体间往往存在关系路径。例如：假设数据集存在三元组<华盛顿，出生地，弗吉尼亚州>、<弗吉尼亚州，所属国，美国>，如果还存在另一个三元组<华盛顿，国籍，美国>，则我们可以学习到存在“出生地+所属国=国籍”，也就是头实体“华盛顿”和尾实体“美国”除了直接关系“国籍”，还存在一个关系路径“华盛顿 $\xrightarrow{\text{出生地}}$ 弗吉尼亚州 $\xrightarrow{\text{所属国}}$ 美国”。

针对这种情况，（Lin, et al. 2015）对TransE模型进行扩展，提出了基于关系路径的翻译模型—PTransE模型[7]。

$$G(h,r,t)=\|\boldsymbol{h}+\boldsymbol{r}-\boldsymbol{t}\|_{L_1/L_2}+\frac{1}{Z}\sum_{p\in P(h,t)}R(p\mid h,t)\|\boldsymbol{p}-\boldsymbol{r}\|_{L_1/L_2} \tag{4}$$

其中，$Z=\sum_{p\in P(h,t)}R(p\mid h,t)$，$P(h,t)$ 是实体对 (h,t) 之间关系路径 p 的集合，$R(p\mid h,t)$ 是关系路径 p 的可信度；$\boldsymbol{p}$ 是关系路径的嵌入向量表示；$\boldsymbol{h}$ 和 $\boldsymbol{t}$ 是头、尾实体嵌入向量表示，$\boldsymbol{r}$ 是关系嵌入向量。

PTransE模型通过寻找实体对间的关系路径，并通过计算关系路径的可信度和对关系路径进行表示，来利用蕴含在关系路径的语义信息，在关系路径方面对TransE模型进行了扩展，为知识表示学习的研究打开了新的方向。

2 基于空间投影和关系路径的翻译模型

TransR模型将实体看作属性的综合体，不同关系专注实体的不同属性。通过采用空间投影的方式，使模型处理复杂关系的能力得到显著提高。PTransE模型试图解决TransE和TransR等模型只局限于学习三元组结构信息的缺陷。通过寻找实体对间的关系路径，并且将关系路径也嵌入到语义空间中来利用关系路径中存在的语义信息，在学习三元组中直接关系的同时，也对关系路径进行学习。TransR模型和PTransE模型是在两个不同的方面对TransE模型进行扩展。

因此，本文考虑结合TransR模型在处理复杂关系时的能力和PTransE模型充分利用了关系路径中语义信息的优势，建立一个新的模型PTransW（Path-based TransE and Considering Relation Type by Weight），提升知识图谱中知识表示的区分能力。并且在TransR模型中，三元组的关系嵌入到同一语义空间中，对于关系路径的寻找、可信度计算和关系路径的表示提供了条件。

对于一个三元组 (h,r,t)，可以使用公式（3）对三元组 h、t 和实体间的直接关系 r 进行学习；如果存在 (h,P,t)，P 为实体对 (h,t) 间的多步关系路径，$P(h,t)=\{p_1,...,p_N\}$，其中 P 定义为 $p=(r_1,...,r_l)$。则再采用公式（4）中的 $\frac{1}{Z}\sum_{p\in P(h,t)} R(p\mid h,t)\,\|\boldsymbol{p}-\boldsymbol{r}\|_{L_1/L_2}$ 对存在关系路径的三元组进行学习。

结合TransR模型和PTransE模型两者优势的新模型损失函数定义为：

$$G(h,r,t)=\|\boldsymbol{h}\boldsymbol{M}_r+\boldsymbol{r}-\boldsymbol{t}\boldsymbol{M}_r\|_{L_1/L_2}+\frac{1}{Z}\sum_{p\in P(h,t)} R(p\mid h,t)\,\|\boldsymbol{p}-\boldsymbol{r}\|_{L_1/L_2} \qquad (5)$$

h，t 在从实体空间 R^m 经过投射矩阵 $\boldsymbol{M}_r\in R^{m\times n}$ 投射到关系空间 R^n 时，投射矩阵 $\boldsymbol{M}_r$ 依赖于关系 r，同一个实体在不同的关系上时，因为所表现的属性不同，将会被投射到关系空间中的不同位置。关系 r 分为四种类型，为了让 h，t 在投射时考虑到所属关系的关系类型，在同一种关系类型上的实体更可能被投射到同一区域，将引入一个与关系类型相关的权重 ω_r。权重 ω_r 与变量 $h_r pt_r$（对于关系 r，数据集中每个尾实体对应的头实体平均个数）和 $t_r ph_r$（对于关系 r，每个头实体对应的尾实体平均个数）相关，参考（Fan, et.al，2014）在TransM模型[9]中的做法，将权重 ω_r 定义为：

$$\omega_r=\frac{1}{\log(h_r pt_r+t_r ph_r)} \qquad (6)$$

则 h， t 在从实体空间 R^m 经过投射矩阵 $\boldsymbol{M}_r$ 投射到关系空间 R^n 时变为 $\boldsymbol{h}_r = \omega_r \boldsymbol{h}\boldsymbol{M}_r$， $\boldsymbol{t}_r = \omega_r \boldsymbol{t}\boldsymbol{M}_r$。

再结合公式（5），PTransW模型的损失函数则定义为：

$$G(h,r,t) = \| \omega_r \boldsymbol{h}\boldsymbol{M}_r + \boldsymbol{r} - \omega_r \boldsymbol{t}\boldsymbol{M}_r \|_{L_1/L_2} + \frac{1}{Z}\sum_{p\in P(h,t)} R(p\,|\,h,t) \| \boldsymbol{p} - \boldsymbol{r} \|_{L_1/L_2} \quad (7)$$

其中， $Z = \sum_{p\in P(h,t)} R(p\,|\,h,t)$， $P(h,t)$ 是实体对 (h,t) 之间关系路径 p 的集合， $R(p\,|\,h,t)$ 是实体对 (h,t) 间的关系路径 p 的可信度； $\boldsymbol{M}_r$ 是将实体从实体空间 R^m 投射到关系空间 R^n 的投射矩阵， $\boldsymbol{M}_r \in R^{m\times n}$； $\boldsymbol{p}$ 是关系路径的嵌入向量表示； $\boldsymbol{h}$ 和 $\boldsymbol{t}$ 是头、尾实体嵌入向量表示， $\boldsymbol{r}$ 是关系嵌入向量（ m,n 分别代表实体空间和关系空间的维度，并且 m,n 可以相同），头、尾实体向量 $\boldsymbol{h}$、 $\boldsymbol{t}$ 和关系向量 $\boldsymbol{r}$ 以及投射矩阵 $\boldsymbol{M}_r$ 即是PTransW模型需要学习的参数。

在训练过程中，对 h 、 t 和 r 对应的嵌入向量 $\boldsymbol{h}$ 、 $\boldsymbol{r}$ 和 $\boldsymbol{t}$ 进行约束。 $\forall h,r,t$ ，有 $\| \boldsymbol{h} \|_2 \le 1$， $\| \boldsymbol{r} \|_2 \le 1$， $\| \boldsymbol{t} \|_2 \le 1$， $\| \omega_r \boldsymbol{h}\boldsymbol{M}_r \|_2 \le 1$ 以及 $\| \omega_r \boldsymbol{t}\boldsymbol{M}_r \|_2 \le 1$。

PTransW模型同样需要计算关系路径的可信度和对关系路径进行表示。关系路径的可信度计算可以采用PTransE提出的PCRA算法[7]。PTransE的数据实验结果也已经表明采用相加的语义组合方式来表示关系路径取得的效果比按位相乘和循环神经网络要好，所以PTransW模型将采用相加方式来表示关系路径。并且在训练时，因为运算时间，本文只考虑2步关系路径。

训练时，将采用随机梯度下降来最小化目标函数。根据公式（7），将PTransW模型的优化目标形式化表示为：

$$L(S) = \sum_{(h,r,t)\in \boldsymbol{S}} [L(h,r,t) + L(h,P,t)] \quad (8)$$

与TransE一样，在实际训练过程中，采用最大间隔法来对知识表示的区分能力进行提升。 $L(h,r,t)$， $L(h,P,t)$ 表示为：

$$L(h,r,t) = \sum_{(h',r',t')\in \boldsymbol{S}^-} [\gamma + E(h,r,t) - E(h',r',t')]_+ \quad (9)$$

$$L(h,P,t) = \sum_{(h,r',t)\in \boldsymbol{S}^-} [\gamma + E(h,P,t) - E(h,r',t)]_+ \quad (10)$$

其中， $[x]_+ = \max(0,x)$ 表示返回0和 x 之间较大的值； γ 为正确三元组损失函数值与错误三元组损失函数值之间的间隔距离； $\boldsymbol{S}$ 是正确三元组所属集合， $\boldsymbol{S}^-$ 为错误三元组所属集合（负样本）。错误三元组是通过替换正确三元组的头实体、尾实体或关系得到， $\boldsymbol{S}^- = \{h',r,t\} \cup \{h,r',t\} \cup \{h,r,t'\}$。

3实验对比分析

3.1 数据集

FB15K :Freebase[15]是一个由元数据组成的大型合作知识图谱，整合了网上大量的资源，目前包含了12亿个三元组和超过8千万的实体。文献[5]从Freebase中抽取了一个稠密子图FB15K（所有实体都包含在维基百科中）用于TransE模型的实验，该数据集包含有592,213个三元组、14,951个实体和1,345条关系。

GEOGRAPHY:地理数据集是本课题组从基础教育地理学科的网络文本资源中，通过信息抽取等知识图谱构建技术得到的三元组集合。地理数据集包含有99,063个三元组、69,123个实体和6,961条关系。

表1 数据集的统计

	训练集	测试集	验证集	#1-1	#1-N	#N-1	#N-N
FB15K	483,142	59,071	50,000	26.2%	22.7%	28.3%	22.8%
GEOGRAPHY	80,815	9,881	8,367	92.61%	0.28%	7.07%	0.04%

3.2 基于FB15K数据集的链接预测实验

3.2.1 参数调节

我们根据前人的经验，将PTransW模型在数据集FB15K上的步长 α 范围设定为 $\{0.1,0.01,0.001\}$；间隔 γ 设定为 $\{1,2,4\}$；为了便于计算，实体空间的维度 m 和关系空间的维度 n 相同，设定范围为 $\{20,50,100\}$，模型运用随机梯度下降优化时总共迭代500次。通过在验证集上作实体预测实验来确定参数。

表2 控制变量调参在验证集上的实体预测结果表

（a）间隔 $\gamma=1$，维度 $m=n=100$，L_1 范式

α	Mean Rank		Hits@10(%)	
	Raw	Filter	Raw	Filter
0.1	1250.88	1131.44	48.11	70.37
0.01	983.45	863.59	49.38	72.16
0.001	**310.59**	**133.87**	**51.54**	**75.35**

(b) 步长 $\alpha=0.001$，间隔 $\gamma=1$，L_1 范式

m,n	Mean Rank		Hits@10(%)	
	Raw	Filter	Raw	Filter
20	**230.48**	**110.99**	**52.71**	**76.63**
50	269.97	149.98	52.17	76.07
100	310.59	133.87	51.54	75.35

(c) 步长 $\alpha=0.001$，维度 $m=n=20$，L_1 范式

γ	Mean Rank	Hits@10(%)

(d) 步长 $\alpha=0.001$，间隔 $\gamma=1$，维度 $m=n=20$

L_1 / L_2	Mean Rank	Hits@10(%)

	Raw	Filter	Raw	Filter
1	**230.48**	**110.99**	**52.71**	**76.63**
2	270.98	151.35	51.64	74.99
4	268.43	148.45	51.27	74.68

	Raw	Filter	Raw	Filter
L_1	230.48	110.99	52.71	76.63
L_2	**229.01**	**110.37**	**53.60**	**77.61**

即使将参数设定了范围，对每一组训练/验证集也有 3×3×3×2＝54 种情况需要考虑。由于数据集规模较大和受限于模型本身的复杂度，将54种情况都训练、验证一遍需要极大的计算花销。因此，我们采用控制变量的思想来确定参数，再在验证集上进行验证。但是有可能出现两个或多个参数相互作用影响结果的情况，为了避免该种情况，再对参数进行随机替换并在验证集上验证。最终，确定了PTransW模型在数据集FB15K上的参数组合为：$\alpha = 0.001$，$\gamma = 1$，$m = n = 20$，采用 L_2 范式。

3.2.2 实体预测

为了便于比较，我们采用文献[5]和文献[7]中所有的方法作为基准线，由于都是基于数据集FB15K进行实验，并且采用相同的评估指标。所以直接参考论文数据，结果如表3所示。

表3 FB15K数据集实体预测计算结果

Metric	Mean Rank		Hits@10(%)	
	Raw	Filter	Raw	Filter
RESCAL [10]	828	683	28.4	44.1
SE [11]	273	162	28.8	39.8
SME(linear) [12]	274	154	30.7	40.8
SME(bilinear) [12]	284	158	31.3	41.3
LFM[13]	283	164	26.0	33.1
TransE[5]	243	125	34.9	47.1
TransH[14]	212	87	45.7	64.4
TransR[6]	198	77	48.2	68.7
PTransE(ADD,2-step)[7]	200	54	51.8	83.4
PTransE(MUL,2-step)[7]	216	67	47.4	77.7
PTransE(RNN,2-step)[7]	242	92	50.6	82.2
PTransE(ADD,3-step)[7]	207	58	51.4	84.6
PTransW	**164.09**	**41.10**	**55.46**	**92.85**
PTransW(only-path)	135.55	18.39	56.69	95.24

从表中可以看出PTransW模型相比其它模型，Mean Rank指标和Hits@10指标的效果远优于其它模型（包括TransR 和PTransE），说明我们将根据关系类型进行空间投影和利用关系路径语义信息相结合是成功的。

在实验过程中，我们发现测试集的59,071个三元组中，存在有2,230个三元组的头、尾实体对间不存在关系路径，那些不包含关系路径的三元组的预测得到的排名都很靠后，从而将测试集中所有三元组的平均排名拉高。因此，我们剔除了那2,230个不存在关系路径的三元组，对剩余的56,841个三元组的排名重新进行了统计，统计结果为表3中PTransW(only-path)所在行。从结果可以得知，剔除了2,230个不存在关系路径的三元组后，Mean Rank的值降低很明显。对于有关系路径的三元组，PTransW模型预测的结果更准确。

为了进一步观察PTransW模型在复杂关系建模时的能力，我们按关系类型作了统计，结果如表4所示：

表4 FB15K数据集上基于关系类型的计算结果

任务	预测头实体(Hits@10)				预测尾实体(Hits@10)			
关系类型	1-1	1-N	N-1	N-N	1-1	1-N	N-1	N-N
SE[11]	35.6	626	17.2	37.5	34.9	14.6	68.3	41.3
SME(linear) [12]	35.1	53.7	19.0	40.3	32.7	14.9	61.6	43.3
SME(bilinear) [12]	30.9	69.6	19.9	38.6	28.2	13.1	76.0	41.8
TransE[5]	43.7	65.7	18.2	47.2	43.7	19.7	66.7	50.0
TransH[14]	66.8	87.6	28.7	64.5	65.5	39.8	83.3	67.2
TransR[6]	78.8	89.2	34.1	69.2	79.2	37.4	90.4	72.1
PTransE(ADD,2-step)[7]	**91.0**	92.8	60.9	83.8	**91.2**	74.0	88.9	86.4
PTransE(MUL,2-step)[7]	89.0	86.8	57.6	79.8	87.8	71.4	72.2	80.4
PTransE(ADD,2-step)[7]	88.9	84.0	56.3	84.5	88.8	68.4	81.5	86.7
PTransE(ADD,3-step)[7]	90.1	92.0	58.7	86.1	90.7	70.7	87.5	88.7
PTransW	90.03	**97.38**	**83.06**	**92.31**	90.14	**90.17**	**96.86**	**94.54**

从表中可以看出PTransW模型在1-N、N-1和N-N复杂关系建模方面，Hits@10指标明显优于其它模型；在1-1关系上，也与表现最好的模型PTransE(ADD,2-step)的结果接近。PTransW模型对比TransE、TransR和PTransE等模型，在复杂关系建模的能力上得到了显著地提高。

3.2.3 关系预测

关系预测，是通过给定 (h,t) 来预测关系 tM 。我们采用文献[7]中所用的方法作为基准线，与PTransW模型作比较。由于都是基于FB15K数据集进行实验，并且采用相同的评估指标。所以直接参考它们的数据结果，整理为表5所示。

表中的Hits@1是指测试集中排名在第一的三元组占整个测试集的比例。从表中可以看出，PTransW模型和其它模型相比，在Mean Rank这项指标要比PTransE（ADD,2-step）差，在测试集中有小部分三元组的排名极靠后，所以导致平均排名偏高。在Hits@1这项指标上比其它模型稍高。我们同样将2,230个不存在关系路径的三元组剔除，得到PTransW(only-path)，发现与不剔除的结果相比，差别并不明显。

在算法复杂度方面，PTransW相较于PTransE增加了投射矩阵 M ，运行时间略有增加，但增加的时间相较于PTransE小很多，所以该方法不会增加过高的时间开销。

表5 FB15K数据集关系预测计算结果

Metric	Mean Rank		Hits@1(%)	
	Raw	Filter	Raw	Filter
TransE(Lin,et.al.2015)[7]	2.8	2.5	65.1	84.3
PTransE(ADD,2-step)[7]	**1.7**	**1.2**	69.5	93.6
PTransE(MUL,2-step)[7]	2.5	2.0	66.3	89.0
PTransE(RNN,2-step)[7]	1.9	1.4	68.3	93.2
PTransE(ADD,3-step)[7]	1.8	1.4	68.5	94.0
PTransW	2.92	2.50	**70.65**	**94.23**
PTransW(only-path)	2.77	2.33	70.14	94.63

3.3 基于GEOGRAPHY数据集的链接预测实验

3.3.1 参数调节

在GEOGRAPHY数据集上，不仅仅需要对PTransW模型进行训练并做链接预测实验，还需要用TransE模型、TransR模型和PTransE模型在GEOGRAPHY数据集上进行训练，并将链接预测实验的结果与PTransW模型做对比分析。

因此，设置TransE在GEOGRAPHY数据集的参数范围为随机步长 α 设定在范围 $\{1,0.1,0.01\}$ ；间隔 γ 设定为 $\{1,2,4\}$ ；语义空间维度 k 的范围为 $\{20,50,100\}$ ，正则化方式为 L_1 / L_2 。经过在验证集上采用与前面3.2.1相同方法进行参数调节，确定参数组合为： $\alpha=0.01$ 、 $\gamma=1$ 、 $k=100$ 以及采用 L_1 正则化方法，并且随机梯度下降时迭代1000次。对于TransR模型，参数范围为 α 范围

为 $\{0.1,0.01,0.001\}$；间隔 γ 设定为 $\{1,2,4\}$；实体空间的维度 m 和关系空间的维度 n 相同，范围为 $\{20,50,100\}$；正则化方式为 L_1 / L_2。最后确定为 $\alpha=0.001$、$\gamma=1$、$m=n=100$ 以及采用 L_1 正则化方法，迭代1000次。对于PTransE模型，参数范围为 α 范围为 $\{0.1,0.01,0.001\}$；间隔 γ 设定为 $\{1,2,4\}$；语义空间维度 k 的范围为 $\{20,50,100\}$，正则化方式为 L_1 / L_2。最后确定为 $\alpha=0.001$、$\gamma=1$、$k=100$ 以及采用 L_1 正则化方法，迭代1000次。对于PTransW模型，参数范围为 α 范围为 $\{0.1,0.01,0.001\}$；间隔 γ 设定为 $\{1,2,4\}$；实体空间的维度 m 和关系空间的维度 n 相同，范围为 $\{20,50,100\}$，正则化方式为 L_1 / L_2。最后确定为 $\alpha=0.001$、$\gamma=1$、$m=n=100$ 以及采用 L_1 正则化方法，迭代500次。

3.3.2 实体预测

实体预测实验中，与上文一致通过给定 (h,r) 来预测 t 以及给定 (r,t) 来预测 h。将TransE、TransR、PTransE模型的结果进行比较，如表6所示。

从表中可以看出，之前在FB15K数据集上表现较好的PTransE模型和PTransW模型在GEOGRAPHY数据集上，实体预测结果反而不如TransE模型和TransR模型。经过分析得知是由于GEOGRAPHY数据集训练规模较小，相对复杂的PTransE模型和PTransW模型在数据集GEOGRAPHY上训练不够充分，并不能发挥它们的优势。

表6 GEOGRAPHY数据集实体预测计算结果

Metric	Mean Rank		Hits@10(%)	
	Raw	Filter	Raw	Filter
TransE	**11,772.24**	**11,765.49**	25.96	26.34
TransR	11,882.67	11,875.83	**26.26**	**26.70**
PTransE(ADD,2-step)	28,281.86	28,273.18	16.37	17.05
PTransW	27,322.89	27,314.23	23.42	24.29

3.3.3 关系预测

在关系预测子实验中，也是通过给定 (h,t) 来预测关系 t。将TransE、TransR、PTransE模型在数据集GEOGRAPHY上做关系预测实验，并将所求结果进行对比分析，如表7所示。

表7 GEOGRAPHY数据集关系预测计算结果

Metric	Mean Rank		Hit@10(%)	
	Raw	Filter	Raw	Filter

TransE	3872.39	3972.37	17.63	17.63
TransR	3,623.20	3,623.18	18.26	18.26
PTransE(ADD,2-step)	1795.50	1795.48	18.67	18.68
PTransW	**720.68**	**720.66**	**45.05**	**45.06**

从表中可以看出，考虑了关系路径和反向关系的PTransE模型和PTransW模型取得的效果明显比TransE和TransR要好。其中，PTransW的效果尤为突出。

4 总结

针对TransE模型在处理知识图谱中复杂关系能力缺失和只局限的使用三元组结构信息的问题。我们将TransR模型和PTransE模型进行结合，并对结合后的模型做了进一步地改进，在空间投影时考虑关系类型，通过加入关系类型的权重来使实体在投射时在不同关系类型上有所区别。未来需要对知识图谱中的知识类型进行更具体的划分，并对不同类型的知识表示进行研究。以及除了链接预测，将知识表示学习应用到关系抽取、实体消歧、实体识别等更多任务中，来进一步的探究以及验证知识表示学习的有效性。

参考文献

1. Singhal A: Introducing the knowledge graph: things, not strings. Google-Blog.2012,http://googleblog.blogspot.co.uk/2012/05/introducing-knowledge-graph-things-not.html.
2. Bengio Y. Learning deep architectures for AI[J]. Foundations and Trends in Machine Learning, 2009, 2(1):1-127.
3. Bengio Y, Courville A, and Vincent P.Representation learning: A review and new perspectives [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2013.35(8):1798-1828.
4. Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositionality[J]. Advances in Neural Information Processing Systems, 2013, 26:3111-3119.
5. Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O. Translating embeddings for modeling multi-relational data[C]//In Advances in Neural Information Processing Systems 26. Curran Associates, Inc. 2787-2795.
6. Lin Y, Liu Z, Sun M, Liu Y, Zhu X. Learning Entity and Relation Embeddings for Knowledge Graph Completion[C]//The 29th AAAI Conference on Artificial Intelligence.
7. Lin Y, Liu Z, Luan H, Sun M, Rao S, Liu S. Modeling Relation Paths for Representation Learning of Knowledge Bases[C]//The Conference on Empirical Methods in Natural Language Processing (EMNLP 2015).
8. 刘知远, 孙茂松, 林衍凯, 谢若冰. 知识表示学习研究进展[J]. 计算机研究与发展, 2016, 53(2): 1-2.

9. Fan M, Zhou Q, Chang E, et al. Transition-based knowledge graph embedding with relational mapping properties[C]//In Proceedings of the 28th Pacific Asia Conference on Language, Information, and Computation. 2014:328-337.
10. Nickel M, Tresp V, Kriegel H. A three-way model for collective learning on multi-relational data[C]//Proc of ICML. New York: ACM, 2011: 809-816.
11. Bordes A, Weston J, Collobert R, et al. Learning structured embeddings of knowledge bases[C]//Proc of AAAI. Menlo Park, CA: AAAI, 2011: 301-306.
12. Bordes A, Glorot X, Weston J, et al. Joint learning of words and meaning representations for open-text semantic parsing[C]//Proc of AISTATS. Cadiz, Spain: JMLR, 2012:127-135.
13. Jenatton R, Roux N L, Bordes A, et al. A latent factor model for highly multi-relational data[C]//Proc of NIPS. Cambridge, MA: MIT Press, 2012:3167-3175.
14. Wang Z, Zhang J, Feng J, Chen Z. Knowledge graph embedding by translating on hyperplanes[C]//In Proceedings of AAAI, 2014:1112-1119.
15. Bollacker K, Evans C, Paritosh P, et al. Freebase: A collaboratively created graph database for structuring human knowledge[C]//Proc of KDD, New York: ACM, 2008: 1247-1250.

Space Projection and Relation Path based Representation Learning for Construction of Geography Knowledge Graph

Abstract. Human-like intelligence has developed rapidly and it benefited from the complete knowledge graph especially primary education knowledge graph represented by geography. The traditional knowledge graph is represented by network knowledge and it is high computational complexity and can't measure or make use of semantic association between entities effectively. This paper puts forward a new algorithm based on deep learning of knowledge representation--PTransW (Path-based TransE and Considering Relation Type by Weight). It combines the space projection with the semantic information of relation path, and consider the semantic information of relation type for further improvement. The experiment results on the FB15K and GEOGRAPHY data sets show that the ability of dealing with complex relation in knowledge graph is improved greatly for PTransW model. For small data sets, the training of TransE and TransR which are low complexity will be more enough. However, PTransE model and PTransW model utilize the semantic information of relation path and reverse relation and perform more outstanding in relation prediction than TransE model and TransR model.

Keywords: TransE, Knowledge Representation Learning, Geography Knowledge Graph

DRTE：面向基础教育的术语抽取方法

李思良[1]，许斌[1]

（1. 清华大学计算机系知识工程研究室，北京市 100084）

摘要：术语抽取从非结构化文本中自动提取专业术语。该工作在中文分词、信息抽取、知识库构建中发挥着重要的作用。当前术语抽取方法很大程度上依赖于词的统计信息，由于基础教育学科中术语具有极强的长尾特性，导致基于统计的术语抽取方法很难提取出处于尾端的术语。本文结合基础教育的学科特点，提出了 DRTE：一种利用术语定义与术语关系挖掘，综合构词规则与边界检测的术语抽取方法。本文以初高中的数学课本为数据源进行术语抽取，实验结果表明我们的术语抽取方案 F1 值达到 83.7%，相比目前的方法提高了 41.8%，能够有效地在中文基础教育领域进行自动化地术语抽取。

关键词：术语抽取；术语定义；术语关系

DRTE: A term extraction method for elementary education

Siliang Li[1], Bin Xu[2]

(1.Tsinghua University knowledge Engineering Group ,Beijing 100084, China)

Abstract: Term extraction is an essential task where we extract terms automatically from unstructured text based on a specific domain. The task plays an important part in the work of Chinese segmentation, Information Extraction and Knowledgebase construction. Previous methods largely rely on terms' statistic information. However, terms in elementary education area have serious Long Tail Effect, which makes it hard to extract terms at the tail part in methods based on statistics. In light of the characteristics of elementary education, we propose DRTE, a method which focus on extracting terms from their definitions and relations. Our method also utilizes term-formation rules and boundary detection strategies. We experiment on math textbooks for middle school and high school. Our method gets 83.7% on F1 performance which significantly improve the current method by 41.8%. Experiments show that our method is fit for term extraction in the area of Chinese elementary education.

Key words: term extraction; term definition; term relation

1 引言

术语作为在特定领域内表达专业概念的约定性符号，在中文分词、句法分析等自然语言领域都发挥着重要的作用。在构建领域知识库的过程中，术语作为领域内知识的主要体现，在知识实例的扩充工作中有着重要的地位。从非结构化文本中手工进行术语标注耗费大量人力与时间，且会存在因标注遗漏而导致召回率降低的情况。因此自动地术语抽取工作受到了越来越多研究者的重视。

目前的术语抽取方法主要包含两个步骤。第一步是通过对字符串的单元性计算来获取候补术语；第二步则通过术语性这一衡量指标来抽取出真正的术语。其中单元性是用来刻画特定字符串组合的稳定性，术语性是用来描述一个语言单位在该领域内的相关程度（Kageura and Umino， 1996）。术语抽取工作已经在多个领域中进行了尝试，例如数学（Stoykova and Petkova， 2012）、生态学（Conrado et al. 2013）、生物医学（Lossio-Ventura et al. 2014）、信息科学（Lossio-Ventura et al. 2014）和自然科学（Dobrov et al. 2011），这些方法大都是基于统计的方法。但当我们为基础教育知识库构建进行术语抽取时，术语的统计特征和专业领域中的术语有较大的不同。我们以数学学科为例，术语“三角形”在初高中课本中共出现 1779 次，而术语“切点圆”则仅仅出现 3 次。数学教材中仅有少部分重要术语被反复使用，这种长尾特性会造成低频词的遗漏。此外，一些基础性术语如“面”、“线”也被广泛地使用在其他领域，这种现象会导致通用性高的术语会因为逆向文件频率而被认为是领

域无关的词语。

基础教育的相关书籍以教授知识为主，内容蕴含了大量术语的定义与术语关系的描述。我们结合基础教育资源的这种学科特性，提出了 DRTE：以挖掘术语定义与术语关系为主，综合构词规则和边界检测的术语抽取方法。我们首先对书籍进行定义抽取，从定义中生成初始的术语候补。之后会进行数次迭代操作，每一轮迭代中，我们会进行如下的操作：在全文和术语候补中寻找带有术语关系指示的内容并挖掘出新的术语候补;从术语候补中综合构词特点与边界检查的方法提取出新的术语；最后将新发现的术语添加到分词的识别中，并开始下一次迭代。当不再有新术语发现时，我们停止迭代操作。

我们的实验针对基础教育的数学学科，选用了初高中数学课本的电子化书本作为数据源。我们的抽取方法的 F1 值达到达到 83.7%，相比目前方法提高了 41.8%。本文的创新点主要包括：（1）提出了一种利用术语定义与术语关系的非监督术语抽取方法；（2）我们通过利用术语的定义与关系的背景信息，避免了基础教育中大量低频术语带来的术语遗漏现象。（3）针对因中文分词误差导致的长术语抽取困难现象，我们提出了迭代式的术语抽取方案。余下内容组织形式如下：第二部分介绍术语抽取的相关工作；第三部分介绍我们的术语抽取方法：DRTE；第四部分介绍我们的实验；第五部分展示我们的实验结果与分析；第六部分给出结论。

2 相关工作概述

术语抽取关注于简单术语（仅由一个词构成的术语）和复合术语（由多个词复合的新术语）的抽取。目前的术语抽取的方法可以分为三种类型：基于语法规则型、基于统计型以及基于机器学习型。

基于语法规则型

术语作为一个领域内独立存在的语言单位，其构词的结构应该是稳定且具有规律的。基于这种假设，我们可以通过挖掘这种语言上的规律来进行术语抽取。一个进行生物学术语提取的方法（Gaizauskas et al. 2000），通过分析生物学词汇的构词方式来构建出一套通用的术语命名规则。另一方面，一些特殊的构词部件（如前缀和特定的缩写）也被用来进行术语的抽取（Krauthammer and Nenadic 2004）。除了构词规则之外，一些词汇在句子中上下文的信息也可以用来进行定义抽取规则的生成（Golik et al. 2013）。基于语法规则的术语抽取方法普遍具有较高的准确率。但由于术语构词规则多变，这些方法通常召回率都不高。

基于统计型

与领域相关的一篇文档通常会针对一个或几个术语展开描述，因而术语在这些文档中的分布应当具有一定的统计特性。利用术语的不同统计特征，可以对术语的术语性进行衡量。例如有利用 TF 信息的方法（Zhang et al. 2005），基于 TF－IDF 的方法（Zhou et al. 2010）。为了解决复合术语的识别问题，C－value 方法（Frantzi et al. 2000）在原有的统计信息中加入了术语长度和嵌套术语的考量。结合中文的特点，一些如互信息（张锋 et al. 2005）、改进 C－value 的方法（胡阿沛 et al. 2013）也相继被提出。基于统计的术语抽取方法对于领域的背景知识要求较低，拥有较高的召回率。但在面对基础教育领域时，由于相关的文档通常以系统教授概念为主，术语的统计规律与其他领域有很大的区别，导致现有的统计量并不能很好地筛选出该领域下的术语。为了应对这种情况，LiTeWi 方法（Conde et al. 2016）提出了利用外部 wikipedia 资源，通过实体链接的办法来进行术语筛选。但该方法受限于外部资源的术语覆盖度与实体链接的准确程度，F1 值仅为 36.8%。

基于机器学习型

基于机器学习的术语抽取方案通常将术语抽取与术语分类结合在一起。这些方法利用训练数据基于机器学习的方法来学习术语抽取的特征（Zhang et al. 2010）。一个机器学习的术语抽取方法（Conrado et al. 2013）使用了 8 个术语的语言学特征（如词性、词根），7

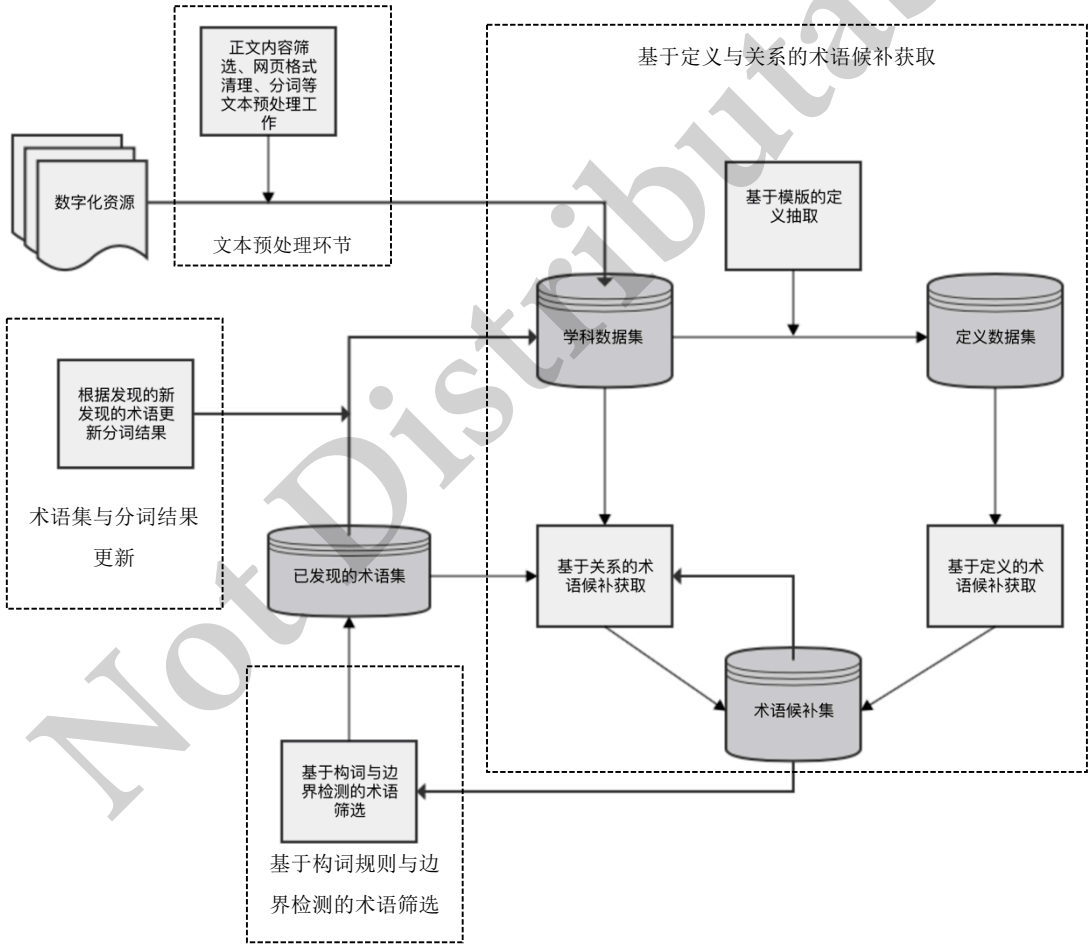
个术语的统计特征（如 TF-IDF 值、词的长度）以及 4 个混合特征（如 C－value）。对于这类通常为有监督学习的方法，如何获取到优质的训练数据是基于机器学习方法的一个困难之处。此外，如何选择适合进行术语抽取工作的特征也是该类方法的难点之一。

实际的术语抽取工作通常不是单独使用上述三种方式的某一种，而是将它们选择性地组合在一起。例如为了利用术语的语言学特征与统计上的趋势，采用了规则与统计相结合的方法。但是，上述方法在直接应用到基础教育术语抽取时，还存在着低频词难以抽取的问题。

3 方法

该小节阐述了我们面向基础教育的术语抽取方法：DRTE。与基础教育相关的书籍以向学生讲授相关知识为主要目标，其内容包含大量的术语定义与对术语关系的描述。为了利用好这些信息，我们提出了以术语定义与术语关系挖掘为主的术语抽取方法。该方法是一个迭代的过程，每一步根据已有的术语集从术语的定义和术语间的关系当中，综合构词规则和边界检测的方法发现新的术语，并更新术语集。DRTE 的流程如图 1 所示，包括如下四个关键环节：（1）文本预处理；（2）基于定义与关系的术语候补获取；（3）基于构词规则与边界检测的术语筛选；（4）术语集与分词结果更新。下面具体描述着四个环节。

图 1 DRTE 方法的流程



3．1 文本预处理

我们的数据来源是基础教育课本的数字化 epub 资源。epub 资源的内容并非纯文本，而是以类似 html 网页的形式进行组织。故在利用这些数据之前，我们需要对其进行数据清洗。

首先筛选出书籍的正文部分（即不包括标题、前言、习题与单元总结的内容），因为正文部分是这些书籍的知识主要来源。图片与表格中的内容会从正文中剔除不被处理。为了避免公式对分词效果产生影响，我们用正则表达式过滤掉书中的数学符号与数学公式。之后我

们会去除所有的网页标签，并根据句号、逗号、分号与问号对文本进行重新分段。我们利用ansj分词工具[1]对文本进行中文分词，并计算每个词的词频。

3．2 基于定义与关系的术语候补获取

3．2．1 通过定义获取术语候补

我们首先从清理后的数据集中抽取定义。在我们的方法中，定义并不是获取术语候补的唯一途径，对于定义抽取的召回率要求不高，故我们采用模版来进行定义的抽取。表1展示了我们使用的模版：

表1 用于定义抽取的模版

<定义部分>(叫|称)(做|为)<被定义部分>
<定义部分>是指<被定义部分>
<被定义部分>的定义(是|为)<定义部分>
称<定义部分>(做|为)<被定义部分>

通过模版匹配抽取出的定义会被分解为两个部分：被定义部分与定义部分。被定义部分揭示了该定义是在描述谁的定义，而定义部分则表明了用来进行下定义的内容。

我们利用定义来获取术语候补基于如下两个假设：(1) 课本中的定义都是用来讲授该学科知识的，故一定都是用来描述该学科中的术语的；(2) 基础教育学科中的术语应当呈现较强的自包含特性，即用来定义某一个术语的词语很可能本身也是术语。故对于一个定义，我们能够将其定义部分和非定义部分各作为一个术语候补。

我们以垂线的定义为例展示基于定义的术语候补获取。垂线的定义：“互相垂直的两条直线中的一条直线叫做另一条直线的垂线”。根据模版匹配，我们能确定被定义部分为“另一条直线的垂线”，定义部分为“互相垂直的两条直线中的一条直线”。根据上述的假设，这两个部分都能作为术语候补。

从上面的例子中，我们可以看出定义部分和被定义部分的句子复杂程度是不同的。通常情况下，定义部分的句子更为复杂。此外，由于定义部分中经常混有公式，这会造成预处理后定义部分的结构并不完整。

针对上面的情况，尽管一条定义当中能够产生两个术语候补，我们设置定义部分产生的术语候补为低置信度，被定义部分产生的术语候补为高置信度。在术语筛选的环节中，会根据不同的置信度等级采取不同的筛选策略。

此外，我们认为在定义部分与被定义部分产生的术语候补中，术语都应当处于靠右侧的部分，故它们均会被标记为右型候补（Rc）。左型候补（Lc）与右型候补（Rc）是用来指出术语更容易出现在术语候补的左侧部分还是右侧部分。在术语筛选阶段会根据术语候补方向的不同采取不同的策略分析。

3．2．2 通过关系获取术语候补

在该步骤中，我们根据已经获取到的术语集，结合术语之间的逻辑关系进行进一步的术语候补的获取。用于术语抽取的逻辑关系包括三种关系：上下位关系、整体部分关系与并列关系。

3．2．2．1 上下位关系

上下位关系指两个词之间体现出的语义包含关系。例如“正方形是一种特殊的长方形”中，“正方形”是下位词，“长方形”是上位词。关系依靠模版：“<下位部分>是<上位部分>”来进行抽取。我们抽取所有的含有上下位关系的句子，并且匹配到的下位部分或上位部分中恰有一个部分是已发现的术语。我们将其中不是术语的部分作为术语候补。例如在上例中，若我们“正方形”已经在我们的已发现术语集中出现，则我们可以根据上面的规则，将“一种特殊的长方形”作为术语候补。

[1] https://github.com/NLPchina/ansj_seg

匹配到的下位部分会被标记为 Rc，上位部分会被标记为 Lc。由于能够反映上下位特征的句式并不一定都具有这种关系，例如“解三角形是一个重要的数学问题”中，匹配到的上位部分并不是一个真正的术语。故我们将上下位关系产生的术语候补设置为低置信度。

3．2．2．2 整体与部分关系

整体与部分关系通过“的”字短语来进行抽取。整体与部分关系中既存在“三角形的边”这样仅涉及术语的关系，也存在如“函数的难点”这样的有非术语参与的关系。为了在保证术语抽取准确度的前提下，更好地利用整体与部分关系进行术语抽取，我们根据抽取到关系的来源的不同，分别针对高置信度术语候补、低置信度术语候补与普通文本采取了不同的关系分析方法。

从高置信度术语候补中发现的整体部分关系很有可能是在描述仅涉及术语的关系，故我们认为是最为可靠的，所以直接将“的”字短语中“的”左右两侧的内容均设置为高置信度的术语候补。“的”字左侧的内容标记为 Rc，“的”字右侧的标记为 Lc。

由于低置信度的术语候补通常句式会比较复杂，我们需要选择“的”字短语中句式较为简单、更可能存在术语的一部分作为术语候补。这里我们根据左右型候补来进行判断。若术语候补是 Lc，则我们选择“的”字短语左侧部分作为术语候补并标记其为 Rc，否则选用右侧部分并设置其为 Lc。最后我们设置这个新发现的术语候补为低置信度。

从普通文本中发现的整体部分关系往往处于句子中的中段部分，关系的上下文较为复杂，很容易引入诸如“三角形的难点”这种类型的噪声结构。故对于这种类型的关系，我们采取了更严格地筛选措施。

由于从普通文本中获得的整体部分关系中很可能并不存在术语，我们首先取出“的”字短语的左右两侧的词。这两个词中必须恰有一个是已经发现的术语。由于我们每次更新已经发现的术语集时都会重新更新一遍分词的结果，所以只要是已发现的术语，它一定会在分词时处理为一个词，而不会被切分成多个词语。我们会将“的”字短语两侧中不是术语的词作为初选的术语候补。

之后，为了避免发生类似从 “三角形的难点”中提取出噪声术语候补“难点”的现象，我们会对上一步中得到的术语候补进行进一步候补可靠性检查。如果一个词是术语，那么与它之间有整体部分关系的词中，术语应当占多数。基于这一假设，我们会检查所有有该术语候补参与的整体部分关系，并根据已经发现的术语集统计其中非术语与术语的比值。若该比值大于我们指定的阈值 Tr，则判断该术语候补是应当剔除的。最终确定的术语候补将被设置为低置信度术语候补。

3．2．2．3 并列关系

我们通过模版：“<并列部分>（<并列部分>、）*[和|或|与]<并列部分>等?”来识别并列关系。若并列部分中有一个为已发现的术语，则其他的并列部分皆很有可能为术语。我们基于上面的假设将满足条件的并列部分作为术语候补。例如“棱锥与棱柱都是常见的几何体”中，若“棱柱”在已发现的术语集中，则我们会将“棱锥”添加到术语候补中。由于并列关系中并列部分的句式结构通常较为简单，且一旦有一个并列部分为术语，其他并列部分为术语的可能性很高，故我们设置抽取出的术语候补为高置信度术语候补，并标记为 Rc。

在并列关系中，经常会出现术语词根省略的情况。例如“锐角、直角和钝角三角形”中，词根“三角形”就在前两个并列内容中被省略了。我们采取词根检验的方法来处理这种情况。我们取出并列关系中最后一个并列部分，依次将其倒数最后一个字、倒数两个字，直至全并列部分作为词根。例如上面的例子中，我们检验的词根有“形”、“角形”、“三角形”、“角三角形”和“钝角三角形”。我们依次检查所有的词根，将该词根置于其余并列关系的尾部构成新的词语，并统计这些词语的出现次数之和。若和的最大值超过了给定的阈值 Ts，则我们认定该并列关系中出现了词根省略现象。在上例中，当词根为“三角形”时，“锐角三角形”

和“直角三角形”的出现次数之和最高，故我们最终产生的术语候补为“锐角三角形”、“直角三角形”和“钝角三角形”。需要说明的是，我们没有统计不带词根的情况中的出现次数之和，即只要我们认定了并列部分可以是“锐角三角形”，我们就不会考虑并列部分为“锐角”的情况。因为，虽然“锐角”和“锐角三角形”从语法上讲都可以看作是处于并列部分的术语，但在人的理解方式中，更倾向于用“锐角三角形”来进行理解。

3．3 基于构词规则与边界检测的术语筛选

基于术语的定义与关系抽取到的术语候补是从句式特征出发获取到的，并不能体现出术语作为词语本身的特点，因此还需要从构词规则与边界检测的角度对术语候补进行进一步的筛选，以确定最终的术语。

3．3．1 构词规则

在平衡词性搭配规则的准确性与普适性的问题上，之前的研究工作主要采取了两种应对措施。一种方法是限制抽取的术语长度，如限制在 2-6 字之间。这种方法可以有效地减少可能地词性搭配情况，但会造成长术语的缺漏。另一种方法是适当宽松词性搭配规则的限制，但这种方法容易造成术语的误判。

我们称一个术语分词后的组成词语个数为该术语的元数。例如“三角形”是一元术语，而“直角三角形”就因为分词结果为“直角”和“三角形”而是二元术语。术语的元数会随着分词结果的变化而变化。我们在每一轮迭代中只考虑元数小于 4 的术语。在每一轮迭代结束后，我们会用已发现的术语更新分词结果。例如“单位正交基底”的初始分词结果是：“单位 正 交 基 底”，该术语是一个五元术语。但在第一次迭代结束之后，其分词结果为：“单位 正交 基底”，是一个三元术语，故在第二次迭代中该术语候补就会被确认为术语。

我们参考的词性表是 ansj_seg 提供的词性表[2]。词性表包括 22 个大类，每个大类下有若干小类。后文提到的词性均指该词性对应的大类以及其包含的小类，在词性标注的过程中，我们发现很多领域术语的词性与分词工具标出的词性有很大区别。例如“边”通常会被标注为副词，但在领域中却应当作为名词。这种现象在基础教育领域的理科中尤为严重。因此在词性搭配规则的选取上，我们去除了常用的必须含有名词成分的限制，根据置信度的不同采用了更宽松的规则，如表 2 所示。

表 2 词性搭配规则

元数	置信度	词性搭配规则
一元	高	无限制
	低	非代词类r、语气词类y、助词类u、连词类c、叹词类e、拟声词类o、处所词类s、状态词类z、方位词类f 、时间词类t 及英文词类en
二元	高	无限制
	低	第二个词不为英文 两个词不是代词类 r、语气词类 y、助词类 u、连词类 c、叹词类 e、拟声词类 o、处所词类 s、状态词类 z、方位词类 f 、时间词类 t
三元	高	至少有一个词性为名词类 n、形容词类 a、动词类 v
	低	最后一个词性为名词类 n、形容词类 a、动词类 v 其余词不是代词类 r、语气词类 y、助词类 u、连词类 c、叹词类 e、拟声词类 o、处所词类 s、状态词类 z、方位词类 f 、时间词类 t
四元及以上	高	全部拒绝
	低	

低置信度的术语候补本身并不可靠，宽松的词性搭配规则容易降低术语抽取的准确性。

[2] https://github.com/NLPchina/ansj_seg/wiki/词性标注规范

故我们对置信度低的术语候补增加了术语命名规则。复杂的术语一般通过简单术语复合而成，故复杂术语的构词核心应当是一个术语。例如术语“离散型随机变量”的核心“变量”就是一个术语。通常情况下，术语的构词核心都在术语的后部，故我们会在已发现的术语集中寻找是否存在一个术语是该术语候补的后缀。如果不存在这样的术语，则在该轮迭代中不再考虑该术语候补。最后，我们会对低置信度的术语候补再进行一次词频的检测。我们会统计它们的出现次数，并选取出现次数高于给定阈值 Tc 的术语候补。

3．3．2 边界检测

学科的语言表达通常具有一定的固定表述方式，这会导致一些领域无关的词语因为经常与特定术语搭配而被误认为是术语的一部分。例如“一条直线”就因为“一条”经常与直线搭配而被误认为是术语。与其结构完全一致的“一元方程”却是一个术语，这导致统计词首词尾中特定字出现概率的方法并不能解决这个问题。

我们选择手工建立边界词表。选取常见的副词（如“时”、“都”、“于”、“各”等等）以及常用的代词和量词搭配（如“这个”、“一组”、“一对”、“一条”等等）。算法 1 展示了我们的边界检测过程。

算法 1 边界检测

```
输入：术语候补集 TCS，满足词性搭配的术语候补集 FTCS，边界词表 BWL，术语集 TS
flag ← false;
newcandidate ← ""
for each 术语候补 tc in TCS do
    if tc.tag = Rc then
         tc.words ← Reverse(tc.words);
    end
    for each word in tc.words do
        if flag and word in BWL then
           break;
        end
        if word not in BWL then
           flag ← true;
           newcandidate ← newcandidate + word;
        end
    end
    if tc.tag = Rc then
         newcandidate ← Reverse(newcandidate);
    end
    if tc in FTCS then
        TS add newcandidate;
    else
        TCS add newcandidate;
    end
    TCS remove tc;
end
return;
```

通过边界检测的步骤，我们达成了两个目标：（1）对通过词性搭配检查的术语候补进行进一步分析，确定最终术语；（2）过滤掉一些四元及以上的术语候补中的一些边界信息，使其元数够降到四元以下。

3．4 术语集与分词结果更新

前三步结束后，这一轮迭代的术语发现工作已经结束。若术语集较上一步相比没有发生变化，则终止迭代并输出最终的术语集。若术语集有更新，则利用这一轮中新发现的术语更新学科数据集和术语候补集中的分词结果。

我们会对分词结果中被分为几个词的术语进行修正，将其合并为一个词。新合并的词的词性根据合并前的最后一个词来判断。例如“异面直线”在合并前被分为了“异面”和“直线”两个词。我们根据最后一个词“直线”来判断“异面直线”的词性。若最后一个词是名词类 n、形容词类 a 或动词类 v，则新词与其词性相同。否则新词的词性为名词 n。如上例中，“直线”的词性是名词类 n，所以“异面直线”的词性与它相同也是名词类 n。

更新分词结果之后，我们会重新计算所有词的词频，并进行下一轮的迭代。

4 实验

4．1 实验数据

我们选择基础教育的数学学科为研究对象，选择了人民教育出版社的初中数学课本 6 本，高中数学必修与理科选修课本 12 本以及初高中教辅书 2 本，共计 20 本书的电子版。数字化的资源以 epub 格式（类似网页形式）组织。经过文本预处理后，共得到 7 万余个短句，共计 45 万余词。

4．2 实验设置

对于从普通文本中发现的整体与部分关系，在术语候补的可靠性检查中，我们设置的阈值 Tr 为 0，即我们采取了最严格的术语检查。只有当与该术语候补之间有整体部分关系的词均为术语时，我们才认为该候补是可靠的。这是由于在实验中，我们发现从普通文本中发现的整体部分关系远没有从定义和术语候补中发现的整体部分关系可靠。

在并列关系中，我们为术语词根省略的处理过程设定的阈值 Ts 为(并列内容数-1)＊3。例如在“锐角、直角和钝角三角形”中，我们会检查“锐角三角形”和“直角三角形”的出现次数之和是否大于 6。这里我们设置的阈值比较低，是由于并列内容的句式较为简单，可靠性较高。将阈值设低一些能够有效地涵盖低频术语。

构词规则筛选中，对于低置信度术语候补的词频环节，我们设置的阈值 Tc 为 10，即只有当该候补出现的总次数超过 10 次时，我们才接受其为术语。

4．3 评价方式

我们首先请基础教育数学老师对全部的课本进行一次标注，从中共标注出 862 个术语。之后我们请专家对由 DRTE 抽取出的术语进行审核，从中挑选出是数学基础教育领域需要涉及的术语。我们将人工标注的结果与 DRTE 抽取出的正确结果进行合并，作为书本中总的术语抽取结果。

由于基础教育领域中术语呈现显著的长尾特性，且如“点”、“线”、“面”这样的术语在很多领域中都有涉及。这导致目前大多数基于统计信息的算法都无法正常工作。我们选择了两个针对大量低频术语存在情况的术语抽取方法进行对比。LiTeWi 方法（Conde et al. 2016）通过与维基百科实体链接来提高低频术语的识别。基于信息熵和词频的方法（李丽双 et al. 2015）是一个针对中文术语的抽取方法。

5 实验结果与分析

表 3 实验结果对比

	准确率（%）	召回率（%）	F1 值（%）
LiTeWi	27.1	57.3	36.8
基于信息熵和词频	47.0	37.8	41.9
DRTE	**90.4**	**78.0**	**83.7**

表 3 展现了 DRTE 的实际效果。DRTE 共抽取出 1087 个正确的术语，F1 值达到了 83.7%，效果相比之前的方法有了巨大的提升。根本原因在于我们改进了术语候补的获取方法。之前的方法为了照顾低频术语而引入了术语候补噪声，为此不得不采取了如与维基百科词条比对和信息熵的方法来提高术语的筛选能力。而我们的方法则从术语候补获取出发，通过定义来获取术语，并利用术语关系借助已发现的术语来识别未发现的术语，大大提高了术语候补的质量，进而提升了整个术语抽取的效果。

为了展示出术语构词长度的分布情况，我们对抽取出的每个术语进行分词，统计构成该术语使用的词语数量。结果如表 4 所示。

表 4 术语构词长度分布情况

	术语个数	出现总次数	总词频（%）
一元词	362	60408	13.23
二元词	496	12657	2.772
三元词	200	2406	0.5269
四元词	28	98	0.02146
五元词	1	4	0.0008759

我们可以看出术语多数是以 3 个以内的词构成的，最复杂的术语是由 5 个词构成的。我

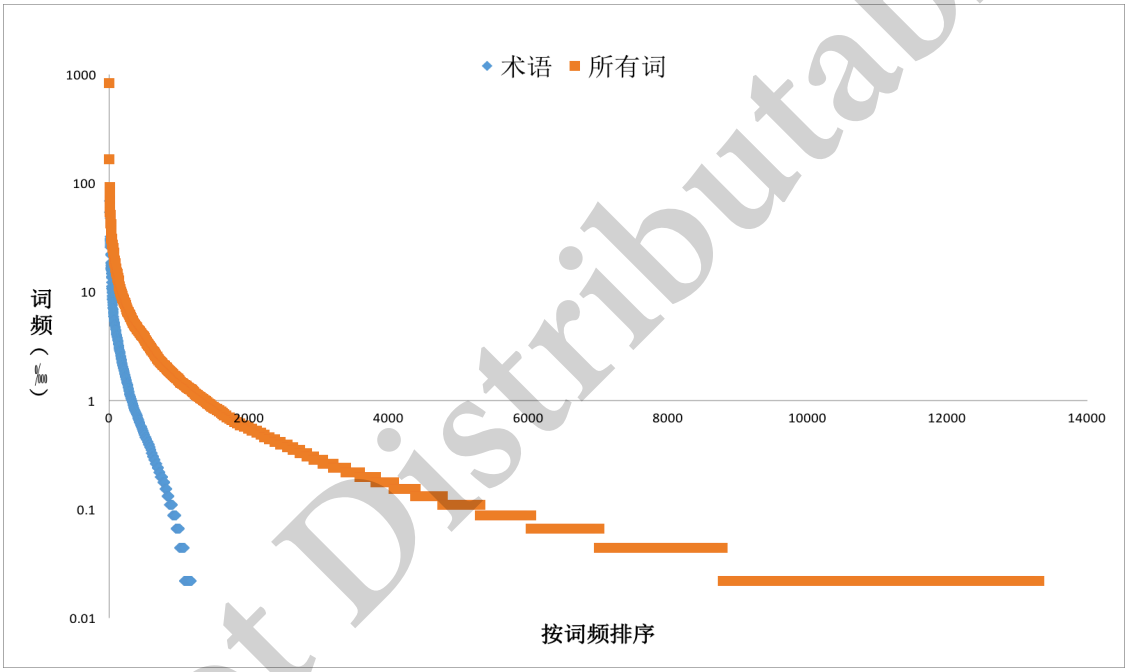
们可以发现一个有趣的现象，尽管二元术语个数最多，但其总词频却远远低于一元术语。这是由于基础教育中词频较高的术语（如三角形）本身就具有很强的通用性，在中文分词系统的训练语料中出现几率较高，故本身就容易被视为一个词。而词频低的术语，由于在通用语料中出现几率低，故容易被分词为多个词。

为了更好地说明为什么基于统计的方法不适用于基础教育中的术语抽取，我们对课本中的术语和所有词按照词频排序后绘制了词频的分布图，如图 2 所示。尽管基于统计的术语抽取方法并不直接使用词频作为唯一的筛选，但该统计量在其他的复合统计量（如 c-value、TF-IDF 等）中有着重要体现，故我们选择词频进行分析。

我们可以看出术语词频在对数坐标轴下呈现近乎直线的分布，这说明术语词频有着指数级的下降速度，呈现明显的长尾效应。故我们在提高方法的召回率时必须以低频术语的词频作为筛选标准，这会导致大量非术语词汇的引入。

此外我们可以看出术语词频的分布区间的下界与所有词词频分布区间一致，这说明处于尾端的术语的词频非常低。术语中词频排位在 2/3 的词语，在所有词的词频排位为 3500 左右。而处排位在 3500 之后的词语本身也非常见词。TF-IDF 统计量将很难区分这两类词，故基于统计的方法很难有效地筛选出术语候补。

图 2 词频分布图



我们从准确率和召回率两个方面来进行 DRTE 方法的误差分析。

DRTE 方法抽取错误的术语共有 115 个，经过分析可归为如下四种情况：

（1） 课本中存在领域无关的定义，如学习指数函数时，给出了“半衰期”的定义。这种情况仅出现 9 次，故我们对课本中的定义绝大部分都是术语定义的假设是比较可靠的。

（2） 固定搭配带来的误差。例如“函数的重点”中，“重点”一词的词频很高，而且与其构成整体部分关系的词均为术语。

（3） 边界检测的误判。我们发现基础教育领域中的一些术语具有多义现象，即在该领域中有特殊含义，都在通常情况下却有不同的含义。例如“一次函数”中的“一次”指“最高项次数”，而“一次独立重复试验”中的“一次”又有不同的含义。故边界检测无法判断这种类型的边界。

（4） 因分词造成的错误。一些句子在一开始就出现了无法纠正的分词错误。例如“其中大圆和小圆”就会被分词为“其 中大圆 和 小圆”，导致误认为“中大圆”是一个术语。

在召回率方面，DRTE 没有抽取出的术语可以分为三种情况：

（1） 一些术语的词频太低。如术语“周期数列”在课本中仅出现过一次。

（2） 一些术语虽然词频较高，但却未在定义与关系中多次出现。如“随机数”。

（3） 一些术语命名方式独特，与其他术语之间没有构词上的联系。这种类型的术语如果由多于 3 个词组成，则无法被识别。如“更相减损术”。

从整体的实验结果来看，我们的方法通过术语定义与术语关系抽取术语候补，充分利用已发现术语挖掘新的术语，能够解决大量低频术语存在的问题。实验结果证明了 DRTE 方法可以有效地应用于基础教育领域中的术语抽取工作。

5 总结

本文针对基础教育领域，提出了 DRTE：一种利用术语定义与术语关系，综合构词规则与边界检测的术语抽取方法。为了解决基础教育领域中术语显著的长尾效应带来的对于低频术语召回困难的现象，我们结合基础教育以知识教授为主的特点，选择从课本中术语的定义与关系来获取术语。我们分别介绍了从术语定义与术语关系中获取术语候补的方法，并阐述了基于构词规则和边界检测的筛选方案。随后我们介绍了实验的数据集与具体设置，并展现了最终的实验结果和相关分析。

实验数据显示我们的方法在数据集上有着良好的表现，能够有效地进行面向基础教育的术语抽取工作。我们的方法对术语的词频依赖很低，能够有效地应对低频术语的情况。此外，我们的方法采取了循环进行术语发现的措施，不断修正分词的结果，能够避免因为分词的误差带来的术语遗漏。

6 致谢

本研究工作得到国家 863 课题（2015AA015401），清华大学自主科研计划（20131089190）的资助。感谢网络多媒体北京市重点实验室对本研究工作的支持。

参考文献：

[1] Kageura K, Umino B. Methods of automatic term recognition[C]//Papers of the National Center for Science Information Systems. 1996: 1-22.

[2] Stoykova V, Petkova E. Automatic extraction of mathematical terms for precalculus[J]. Procedia Technology, 2012, 1(10):464-468.

[3] Conrado M S, Pardo T A S, Rezende S O. Exploration of a Rich Feature Set for Automatic Term Extraction[M]// Advances in Artificial Intelligence and Its Applications. Springer Berlin Heidelberg, 2013:342-354.

[4] Lossio-Ventura J A, Jonquet C, Roche M, et al. Yet Another Ranking Function for Automatic Multiword Term Extraction[J]. Lecture Notes in Computer Science, 2014, 8686(8686):52-64.

[5] Dobrov B V, Loukachevitch N V. Multiple Evidence for Term Extraction in Broad Domains[C]//RANLP. 2011: 710-715.

[6] Gaizauskas R, Demetriou G, Humphreys K. Term Recognition and Classification in Biological Science Journal Articles[C]// Computional Terminology for Medical & Biological Applications Workshop of the 2 Nd International Conference on Nlp. 2000:37--44.

[7] Krauthammer M, Nenadic G. Term identification in the biomedical literature[J]. Journal of Biomedical Informatics, 2004, 37(6):512-526.

[8] Golik W, Bossy R, Ratkovic Z, et al. Improving term extraction with linguistic analysis in the biomedical domain[M]// The rarest of the rare :. :312-313.

[9] Zhang F, Yun X U, Hou Y, et al. Chinese Term Extraction System Based on Mutual Information[J]. Application Research of Computers, 2005.

[10] Zhou L, Shi S, Feng C, et al. A Chinese Term Extraction System Based on Multi - Strategies Integration[J]. Journal of the China Society for Scientific & Technical Information, 2010.

[11] Frantzi K, Ananiadou S, Mima H. Automatic recognition of multi-word terms: the C-value/NC-value method[J]. International Journal on Digital Libraries, 2000, 3(2):115-130.

[12] 胡阿沛, 张静, 刘俊丽. 基于改进 C-value 方法的中文术语抽取[J]. 现代图书情报技术, 2013(2):24-29.

[13] Zhang X, Song Y, Fang A C. Term recognition using Conditional Random fields[C]// Natural Language Processing and Knowledge Engineering (NLP-KE), 2010 International Conference on. IEEE, 2010:1-6.

[14] Conde A, Larrañaga M, Arruarte A, et al. litewi: A combined term extraction and entity linking method for eliciting educational ontologies from textbooks[J]. Journal of the Association for Information Science & Technology, 2015, 67(2):380-399.

[15] 张锋, 许云, 侯艳,等. 基于互信息的中文术语抽取系统[J]. 计算机应用研究, 2005,

22(5):72-73.
[16] 李丽双，王意文，黄德根．基于信息熵和词频分布变化的术语抽取研究[J]．中文信息学报，2015，29(1):82-87.

基于表示学习的开放域中文知识推理

姜天文[1]　秦 兵[2]　刘 挺[3]

(哈尔滨工业大学 计算机科学与技术学院，哈尔滨 150001)

摘要. 知识库通常以网络的形式被组织起来，网络中每个节点代表实体，而每条连边则代表实体间的关系。为了利用这种网状知识库中的知识，往往需要设计专门的复杂度较高的图算法，但这些算法并不能很好适用于知识推理，尤其是随着知识库的知识规模不断扩大，基于网状结构知识库的推理很难较好地满足实时计算的需求。本文的主要研究内容是，使用基于TransE模型的知识表示学习进行知识推理，包括对实体关系三元组中关系指示词以及尾实体的推理，其中关系指示词推理的实验取得了较好的结果，且推理过程无需设计复杂的算法，仅涉及向量的简单运算。另外，本文对原始TransE模型的代价函数进行改进，以更好地适用于开放域中文知识库表示学习。

关键词: 知识库表示学习；知识推理；开放域；中文；知识库

1　介绍

在过去的十几年里，大规模的知识库的构建已经有了很好的进展。由普林斯顿大学设计的覆盖范围宽广的语言知识库WordNet[1]；知识条目由用户添加并共享的世界知识库FreeBase[2]；以及国内哈尔滨工业大学社会计算与信息检索研究中心设计构建的开放域中文知识图谱《大词林》等。这些知识库通常以网络的形式被组织起来，网络中每个节点代表实体，而每条连边则代表实体间的关系。因此大部分知识往往可以用三元组来表示（头实体，关系，尾实体），其中最具代表性的就是万维网联盟发布的资源描述框架技术标准[3]。

但随着知识库的知识规模不断扩大这种网状的表示形式目前却住主要存在以下问题：计算效率问题[4]、无法很好应对数据稀疏问题[4]。以符号为基础的网状形式的知识库无法应对连续空间里的数值计算，单纯的符号和逻辑的表示使得知识库中的知识越来越离散化，知识之间无法很好整合在一起，无法有效应对长尾问题，这也就使得智能系统无法更加灵活地使用知识库，比如进行知识推理。

姜天文(1994-)，男，吉林辽源人，本科生，主要研究方向：自然语言处理、信息抽取；
秦　兵(1968-)，女，陕西华阴人，博士，教授，博士生导师，主要研究方向：中文信息处理、情感分析、信息抽取、篇章分析；
刘　挺(1972-)，男，黑龙江哈尔滨人，博士，教授，博士生导师，主要研究方向：自然语言处理，文本挖掘，文本检索

表示学习[5]旨在将网状的语义信息表示为稠密低维的实值向量，在低维空间中两个对象距离越近语义相似度越高，正是这一点有望解决别名问题。在这种低维空间中有望高效计算实体和关系的语义联系；另外，由于每个对象的向量均为稠密有值的，因此可以度量任意对象之间的语义相似度，并且将大量对象投影到统一空间的过程能够将高频对象的语义信息用于帮助低频对象的语义表示，提高低频对象的精确性，由此可知这种知识表示学习可以有效解决数据稀疏问题。基于以上叙述的特点，这种知识的分布式表示最终可以使得知识的获取、推理的性能显著提升。

本文使用Bordes等人于2013年提出的TransE模型[10]，同时对模型的代价函数进行改进以用于开放域中文知识库的表示学习，相比于传统知识库，开放域知识库使用关系指示词代替关系类型，而且实体更为丰富，粒度更加细腻。本文主要研究对开放域中文知识库基于表示学习的知识推理方法，包括对实体关系三元组中关系指示词以及尾实体的推理。

2 基于翻译模型的知识库表示学习方法

目前国内外的知识表示工作主要针对的是传统的非开放域的英文知识库。主要的思路是把知识库嵌入到一个连续的向量空间中，并保留了原始知识库的某些特性。这些知识表示的方法通过最小化全局损失函数来获得实体和关系的表示，而且这个全局损失函数涉及到了所有知识图谱中的实体和关系，这也就意味着实体或关系的表示是编码了全局的信息所得到的。

早期在知识表示方面主要有以下模型：距离模型[6]、能量模型[7][8]、张量模型[9]。早期的这种知识表示的方法中，大多数关注于提高表现力和模型的普遍性，而越来越高的表现力随之而来的是模型的复杂度增加、参数增加，以及训练的花销巨大，不仅如此，由于高能力的模型正则项很难设计，所以有潜在的过拟合的情况发生；另外，由于非凸最优化问题有很多局部的极小值，这使得训练难度增加，导致模型无法拟合数据[10]。

近年来提出的翻译模型[10]简单有效，在大规模知识图谱上效果明显，自提出以来大量研究工作[11][12][13]都对其进行扩展和展开，可以说翻译模型已经成为知识表示的代表模型，其中Bordes等人于2013年提出的TransE模型[10]简单可行，完全适合大规模知识库的表示学习，近年来提出的一系列模型都是以TransE模型为蓝本，所以本文的研究主要基于TransE模型，同时对模型的训练方法进行改进以用于开放域中文知识库的表示学习。

2.1 表示学习概念以及理论基础

表示学习概念 表示学习是指，通过使用机器学习的方法将研究对象的语义信息表示为低维稠密的实值向量。在该低维稠密的向量空间中，我们可以通过余弦距离或欧氏距离等方式计算任意两个对象之间的语义相似度。

除了表示学习之外，实际上还有更简单的数据表示方案，称其为“one-hot” 表示[14]。这种方案也是将对象表示为实值向量，只不过向量中只有某一

维度为非零，其余维度的值均为0，这也正是“one-hot”一词的由来。“one-hot”无需学习过程，正是由于其简单而高效，在信息检索和自然语言处理中得到广泛应用。但“one-hot”的缺点在于，它认为所有表示对象时相互独立的，也就是说，在这个表示空间中所有对象的向量都是正交的，如此一来通过余弦距离或是欧式距离计算的语义相似度均为0，而这一点是不符合实际情况的，会丢失大量的信息。例如，*“哈尔滨”*和*“长春”*虽然是两个不同的词汇，但由于他们都是省会城市，因此应当具有较高的语义相似度。然而“one-hot”无法有效利用这些对象间的语义相似性用于表示对象。与“one-hot”不同，表示学习维度较低，从而有助于提高计算效率，同时也能够充分利用对象间的语义信息。

表示学习理论基础 我们所处的世界是离散的，每个物体具有明确的界限。当人们观察这个世界时，大脑中相应的大量神经元会产生抑制或者激活的信号，这些信号的状态构成大脑中的内部世界，在这个内部世界中，外界事物对于它变成了众多神经元共同产生的一系列抑制或激活信号。单纯看一个神经元的状态，并没有明确的含义，无法通过它来区分不同的事物，但是众多神经元产生的状态集合在一起却可以表示世间的万物。

通过表示学习得到的低维稠密向量表示是一种分布式表示，向量的每一维并没有明确的含义，但是综合各维形成的向量却能够表示对象的语义信息。分布式表示的向量可看作大脑中众多的神经元，每一维对应于单独的一个神经元，而每一维度值代表该神经元抑制或激活状态。

2.2 TransE模型的改进

TransE模型的表示学习对象是知识库中的实体关系三元组。TransE模型将实体间的关系看作一种两个实体间的翻译操作，关联着两个实体。在本文中，我们使用h代表头实体[4]、$\boldsymbol{h}$表示头实体的向量表示、r代表关系、$\boldsymbol{r}$表示关系的向量表示、t代表尾实体、$\boldsymbol{t}$表示尾实体的向量表示，TransE模型的核心思想是：如果(h, r, t)成立，那么，认为尾实体t的向量表示应该和头实体h的向量表示加上某个由关系r决定的向量表示结果相接近。基于这个核心思想，TransE 优化的目标是对于满足关系的(h, r, t)，有:

$$\boldsymbol{h} + \boldsymbol{r} \approx \boldsymbol{t}$$

如图 1所示。也就是说，当(h, r, t)成立时，在向量空间中$\boldsymbol{t}$应该是向量$\boldsymbol{h} + \boldsymbol{r}$最近的邻居；当$(h, r, t)$不成立时，在向量空间中$\boldsymbol{t}$应远离向量$\boldsymbol{h} + \boldsymbol{r}$。

使用$d(\boldsymbol{h} + \boldsymbol{r}, \boldsymbol{t})$表示向量$\boldsymbol{h} + \boldsymbol{r}$到$\boldsymbol{t}$的距离，可以使用L1或L2范式计算距离。模型的代价函数为：

[4] 在本文中，我们考虑实体关系三元组的方向性。如，对于知识“黑龙江的省会城市是哈尔滨”，那么三元组（哈尔滨，省会，黑龙江省）是不正确的表述，而（黑龙江省，省会，哈尔滨）才是正确的，所以对于关系“省会”：“黑龙江省”就是头实体，“哈尔滨”是尾实体，反过来是不正确的。

$$\mathcal{L} = \sum_{(h,r,t)\in S} \sum_{(h',r,t')\in S'_{(h,r,t)}} [\gamma + d(h + r, t) - d(h' + r, t')]_+ \qquad (1)$$

其中$[x]_+$代表x的正数部分，$\gamma > 0$是一个边界值，另外，

$$S'_{(h,r,t)} = \{(h', r, t) | h' \in E\} \cup \{(h, r, t') | t' \in E\} \qquad (2)$$

其中E代表实体集合。模型训练过程中所需的三元组负例是通过公式(2)构造的，即替换正确三元组的头尾实体。实体和关系的向量表示都是随机初始化的，训练的过程就是不断减小正例三元组的距离$d(h + r, t)$，并使它尽可能的小于所有它对应的三元组负例的距离$d(h' + r, t')$。

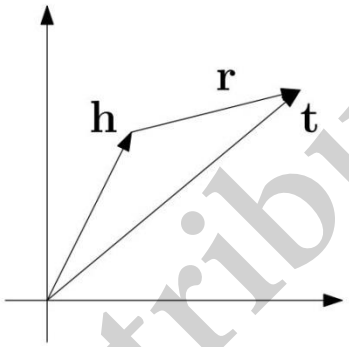


图 1. TransE模型的核心思想

通过观察公式(1)，可以发现TransE方法在构建三元组负例的时候只对头尾实体进行替换，其原因在于传统的知识库中的关系是由关系类型代替，而关系类型的数量较少且相互的区分性较大，所以构造三元组负例时替换关系的意义不大。但对于开放域实体关系三元组，其关系用关系指示词表示，关系指示词的数量较大且相比关系类型区分性并不大，如，关系指示词“董事长”和“校长”在传统三元组中都会使用“雇佣关系”代替，但在开放域三元组中使用不同的关系指示词代替，所以在面向开放域知识库的研究中，关系指示词对于训练的过程不容忽视。

基于以上的原因，我们对原始TransE模型的代价函数进行改进以更好的适用于开放域中文知识库的研究工作。为了进行区别以便后续比较，将改进后的TransE模型命名为TransE_ipv（ipv为imporove简写），TransE_ipv的训练过程中的代价函数为：

$$\mathcal{L} = \sum_{(h,r,t)\in S} \sum_{(h',r',t')\in S'_{(h,r,t)}} [\gamma + d(h + r, t) - d(h' + r', t')]_+ \qquad (3)$$

其中$[x]_+$代表x的正数部分，$\gamma > 0$是一个边界值，另外，

$$S'_{(h,r,t)} = \{(h', r, t) | h' \in E\} \cup \{(h, r, t') | t' \in E\} \cup \{(h, r', t) | r' \in R \qquad (4)$$

其中E代表实体的集合，R代表关系指示词的集合。主要的改进在于在构造三元组负例的时候不仅替换头尾实体，而且替换关系指示词，使得训练出来的关系指示词更具有区分性。

3 实验

由于国内没有适合本文研究并且公开数据的开放域中文知识库，我们从结构化的百度百科结构化数据“infobox”中抽取获得大量开放域实体关系数据进行实验。在本节中，我们提出了应用知识表示学习的关系指示词推理方法，以及尾实体推理方法。实验结果显示，应用知识分布式表示的关系指示词推理准确率可以达到80%以上。在进行应用知识分布表示的尾实体推理测试中，准确率在20%左右，和关系指示词推理相比效果较差，我们对其原因进行分析并验证，使用增加训练过程中三元组负例的方法可以将准确率提升7个百分点。

3.1 实验数据的获取

由于国内没有适合本文研究并且公开数据的开放域中文知识库，我们决定从互联网中抽取开放域实体关系三元组作为实验数据。通过观察，我们发现百度百科有一部分被称为“infobox”描述词条属性的结构化内容，该部分包含大量潜在的实体关系信息，我们希望从中获取实体关系三元组作为实验使用的实体关系三元组数据。

“infobox”[5]一词源于维基百科，是一种包含属性-值对结构化文档。作为全球最大的中文百科网站，百度百科也借鉴了这一设计，在大部分词条页面中都设有“infobox”，用于记录该词条的重要属性-值对信息，如图 2所示。

中文名	哈尔滨工业大学计算机科学与技术学院	现任校长	周玉院士
英文名	哈工大计算机学院	知名校友	陈光熙 王天然 怀进鹏
创办时间	2000年	校　训	规格严格，功夫到家
类　别	国家示范性软件学院	专职院士	方滨兴 人
学校类型	工科	主要院系	计算机科学与技术系 信息安全 生物信息
属　性	211工程 985工程 C9	主要奖项	国家科技进步一等奖（2002年）

图 2. 百度百科中“哈尔滨工业大学计算机科学与技术学院”一词的infobox

“infobox”中包含的是与词条相关的众多*“属性-值”*对，这些*“属性-值”*对与词条可以组成三元组，但这种三元组并不都是我们要找的实体关系三元组，因为*“属性-值”*对中的值并不一定是实体，如*“规格严格功夫到家”*，而*“周玉院士”*就是一个实体，所构成的即为实体关系三元组。

[5] https://en.wikipedia.org/wiki/Infobox

通过观察发现，词条的百科页面中存在很多的具有链接的词汇，这部分文本一般称为锚文本，而百科页面中的这些锚文本是指向另一个百科词条页面的，如果我们假设百度百科中收录的词汇全部为实体词（百科中记录的一般是现实世界中的概念，可以认为其大部分是实体），那么百科页面中的锚文本也即是实体词汇，如图 3所示。


哈尔滨工业大学（*Harbin Institute of Technology*），简称"哈工大（HIT）"，坐落于中国北方冰城哈尔滨市，中华人民共和国工业和信息化部直属重点大学，首批"211工程"、"985工程"重点建设院校，"九校联盟(C9)"、"中俄工科大学联盟"、"中国-西班牙大学联盟"主要成员，国家首批"111计划"、"2011计划"、"千人计划"、"卓越计划"入选高校，中管副部级建制，由工业和信息化部、教育部、黑龙江省人民政府三方重点共建。

哈尔滨工业大学溯源于1920年创办的哈尔滨中俄工业学校，建校初衷为培养铁路工程技术人才；而后历经"中俄工业大学校"、"哈尔滨工业大学校"、"哈尔滨高等工业学校"等多个阶段，学校在1938年1月正式定名为哈尔滨工业大学，沿用至今。[1]

截止2015年7月，哈工大已有材料科学、工程学、物理学、化学、计算机科学、环境与生态学、数学、生物学与生物化学等8个学科进入ESI全球前1%的研究机构行列，其中材料科学、工程学已进入全球前1‰的研究机构行列。该校拥有哈尔滨本部及哈尔滨工业大学（威海）、哈尔滨工业大学深圳研究生院三个校区，共有全日制学生31903人，其中本科生16718人、研究生13263人（含硕士生7585人、博士生5678人），留学生1922人。

图 3. 百度百科*“哈尔滨工业大学”*一词的百科页面中部分文本

我们可以认为在“infobox”中含有锚文本的*“属性-值”*对为实体关系。如图 2中属性*“知名校友”*以及*“专职院士”*，这两个属性值都是锚文本，由此我们可以获取三个实体关系三元组：（哈尔滨工业大学计算机科学与技术学院，知名校友，王天然）、（哈尔滨工业大学计算机科学与技术学院，知名校友，怀进鹏）、（哈尔滨工业大学计算机科学与技术学院，专职院士，方滨兴）。

据此方法，我们共从百度百科的“infobox”中共获取2,438,145条开放域实体关系三元组[6]，虽然可能存在一些噪声数据，但就像知识库允许存在少量噪声数据，这些噪声数据对实验结果并无太大影响。

将获得的三元组数据集作为规模最大的“all数据集”，另从其中抽取50余万的三元组组成“small数据集”，设置不同规模的数据集原因在于使用小规模数据集进行课题研究前期的快速实验测试，以快速改进模型，设置合适的测试实验并记录结果。

将三元组数据集划分为两个集合：训练集、测试集，并需要使得两个集合满足独立同分布条件，以用于模型的训练和测试。除独立同分布外，两个集合需满足以下三个条件：

1) 测试集中的实体集合为训练集中实体集合的子集，即测试集中所有三元组涉及到的实体在训练集中都有出现，其目的在于防止测试时实体词存在未登录，从而找不到对应的实体向量；
2) 测试集中的关系指示词集合为训练集中关系指示词集合的子集，即测试集中所有三元组涉及到的关系指示词在训练集中都有出现，其目的在于防止测试时关系指示词存在未登录，从而找不到对应的关系指示词向量；
3) 训练集和测试集的三元组交集为空，即不存在既在训练集中出现又在测试集中出现的三元组。

[6] Code: https://github.com/twjiang/baike_crawler

获得的两个不同规模的实验数据集如表 1所示：

表 1. 实验所用到的两个不同规模的数据集

–	Small 数据集	All 数据集
实体数量	333,007	1,551,231
关系指示词数量	21,649	57,235
关系三元组数量	524,676	2,438,145
训练数据三元组数量	519,676	2,428,145
测试数据三元组数量	5,000	10,000

3.2 关系指示词推理

为何要进行关系指示词推理？在这之前，我们需要引出一个概念——“知识库关系补全”。知识库关系补全是指：对于现有知识库中有潜在关系但未在知识库中标明的两个实体进行关系推理。如知识库中有以下两个实体关系三元组：

（泰坦尼克号，主要角色，杰克），（莱昂纳多，饰演，杰克）

那么，我们希望推理出如下关系以补全到现有知识图谱中：

（泰坦尼克号，主演，莱昂纳多）

总结下来，知识库关系补全需要两个阶段：存在潜在关系实体对的发现、对潜在关系进行推理。本实验假设已经识别出存在潜在关系的实体对，主要任务是测试通过表示学习得到的向量空间中的知识库是否可以对这个潜在关系进行推理，并给出较为准确的答案抑或包含答案的候选集合。

我们将测试数据中的三元组的关系指示词“挖空”，基于已训练好的实体和关系指示词的向量表示对关系指示词进行推理，并和标准答案进行对比，以计算准确率。

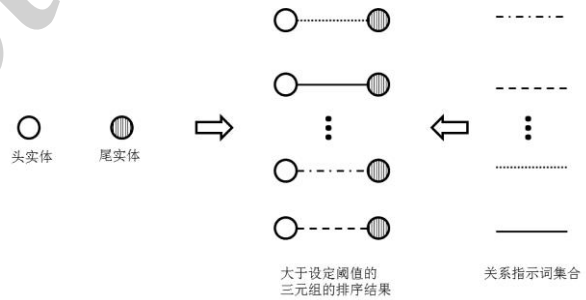


图 4. 关系指示词推理方法简图

具体的测试方法：对于每一对实体，遍历所有的关系指示词组合成一个三元组，对每个这样的三元组计算头实体与关系指示词相加得到的向量到尾实体向量在空间中的距离*d*，距离*d*越小说明三元组成立的可能性越大。设定一个距离阈值，对距离*d*小于阈值的三元组按照距离*d*升序排列（过程简图见图 4）。对每一对实体记录排名前十的三元组的关系指示词，记录正确关系指示词的排名在前十名的比例，以及排名为第一的比例分别作为准确率，并分别记录召回率，计算F值。

这里我们需要对阈值进行确定。在确定阈值的实验中，不同阈值的结果如表2、表 3所示。

表 2. 测试不同阈值对关系指示词推理实验结果（small数据集）

model	threshold	@hit_10	recall_hit_10	@hit_1	recall_hit_1
TransE	0.7	48.05%	9.34%	39.81%	7.74%
TransE_ipv	0.7	94.04%	17.36%	88.73%	16.38%
TransE_ipv	1	92.41%	36.06%	83.03%	32.40%
TransE_ipv	1.3	91.59%	53.80%	79.03%	46.42%
TransE_ipv	--	81.82%	81.82%	63.48%	63.48%

表 3. 测试不同阈值对关系指示词推理实验结果（all数据集）

model	threshold	@hit_10	recall_hit_10	@hit_1	recall_hit_1
TransE_ipv	0.7	93.58%	30.18%	77.36%	24.95%
TransE_ipv	1.0	90.65%	48.27%	72.29%	41.16%

其中`threhold`表示阈值的取值，“--”表示未设定阈值，`@hit_10`和`@hit_1`分别表示正确关系指示词的排名在前十名的比例和排名为第一的实体对数目占所有存在*d*小于阈值的关系指示词的实体对数目的比例，`recall_hit_10`和`recall_hit_1`表示正确关系指示词的排名在前十名的比例和排名为第一的实体对数目占所有测试集中实体对数目的比例。

表 2记录在small数据集中测试不同阈值对关系指示词推理实验结果。通过表2中的数据，首先可以发现TransE_ipv的效果明显优于原始TransE的训练方法，无论是准确率还是召回率都有大幅度的提升，究其原因在于TransE_ipv在构造三元组负例的时候考虑到了关系指示词，不仅仅是替换头尾实体，这对于开放域知识库中关系指示词数量较大的特点极为重要。另外，通过表 2可以发现，在TransE_ipv中随着阈值的增加召回率随之增加，但准确率却在下降。由于在本实验中我们更关注于准确率，所以最佳阈值锁定在0.7和1.0，观察发现在阈值

从0.7过渡到1.0时，虽然准确率有所下降，但召回率却翻倍增长，所以将最佳阈值定为1.0。

表 3记录在all数据集中测试不同阈值对关系指示词推理实验结果。同样，我们将阈值定为1.0，另外，很容易发现在all数据集中的各项数据相比small数据集中有所下降，其原因在于由于硬件条件限制导致两者的训练方式不同造成的。

综合上述实验结果并选取最佳的阈值，得到所示的本实验在small数据集合all数据集的最终结果。

表 4. 关系指示词推理测试的实验结果

data set	@hit_10	recall_hit_10	F1_hit_10	@hit_1	recall_hit_1	F1_hit_1
small(TransE)	48.05%	9.34%	15.64%	39.81%	7.74%	12.96%
samll(TransE_ipv)	92.41%	36.06%	51.88%	83.03%	32.40%	46.61%
all(TransE_ipv)	90.65%	48.27%	63.00%	72.29%	41.16%	52.45%

其中F1_hit_10和F1_hit_1表示对应的F1值。

相比于符号化的网状知识库表示，使用表示学习得到的实体分布式表示可以通过计算高效地推理出实体对中潜在的关系，召回率可以达到40%左右，准确率高达80%左右。

3.3 尾实体推理

有些情况下，我们希望获取某个实体具有特定关系的实体，比如给定实体A和关系B，我们希望找到和实体A具有关系B的实体，我们称这个实体为C。当三元组（A，B，C）不存在于知识库中时，我们希望通过简单的计算即可较为准确的得到C，抑或得到一个候选序列并C存在于这个候选序列中。

本实验的目的就是当（A，B，C）不存在于知识库中时，测试通过表示学习得到的向量空间中的知识库是否可以推理出尾实体，给出较为准确的答案抑或包含答案的候选集合。

我们将三元组的尾实体“挖空”，基于已训练好的实体和关系指示词的向量表示对测试集三元组中的尾实体进行推理，并和标准答案进行对比，以计算准确率。

具体的测试方法：和关系推理相似，对于每一对头实体、关系指示词组合，遍历所有的实体作为尾实体组合成一个三元组，对每个这样的三元组计算头实体与关系指示词相加得到的向量到尾实体向量在空间中的距离d，距离d越小说明三元组成立的可能性越大。之后的步骤设置了两种方法：

方法一：设定一个距离阈值，对距离d小于阈值的三元组按照距离升序排列，对每一对实体记录排名前十的三元组的尾实体（方法一简图见图 5）。记录正确尾实体的排名在前十名的比例，以及排名为第一的比例作为准确率，并分别记录召回率。

方法二：设定一个距离阈值，对距离*d*小于阈值的三元组取出其头实体以及尾实体，其中头实体即为A，尾实体即为要推理的目标实体（记为C’），然后利用A和C’对关系进行推理，记录正确关系B的排名，使排名和距离*d*相乘作为对C’的打分，认为分数越少越有可能是正确实体。方法二是将方法一和关系指示词推理相结合，利用关系指示词推理的结果反馈指导实体推理。

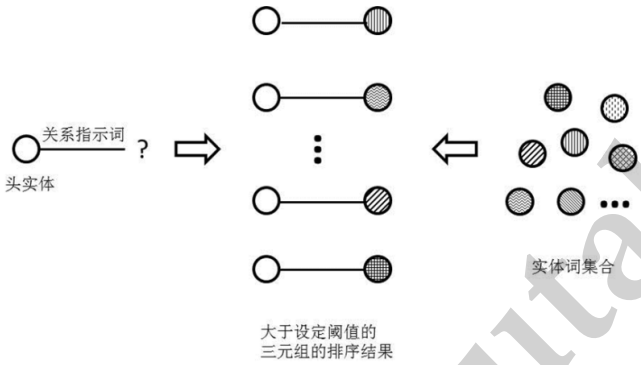


图 5. 尾实体推理方法一简图

对方法一、二的测试结果如表 5所示：

表 5. 利用方法一、二做尾实体推理测试的实验结果（small数据集）

model	method	threshold	@hit_10	@hit_1	recall_hit_1
TransE_ipv	first	1.0	30.45%	15.46%	14.44%
TransE_ipv	second	1.0	38.49%	20.83%	15.22%

实验结果显示方法二的效果更好。本实验也存在阈值的选择问题，由于1.0是关系推理时的最佳阈值，这里只增加了一组阈值为1.3的对比实验，实验结果如表 6所示：

表 6. 阈值为1.0、1.3的尾实体推理测试的实验结果（small数据集）

model	threshold	@hit_10	recall_hit_10	@hit_1	recall_hit_1
TransE_ipv	1.0	38.49%	28.12%	20.83%	15.22%
TransE_ipv	1.3	36.97%	32.84%	18.15%	16.12%

综合来看阈值为1.0时的F值较高，选择1.0为最佳阈值取值。综合上述实验结果并选取最佳的阈值，得到所示的本实验在small数据集合all数据集的最终结果。

通过观察发现尾实体推理的准确率远不如关系推理，通过分析可能是实体具有长尾分布的特点造成的，这是很多大规模数据具有的，这些长尾部分的实体和其他实体有极少的关系联系在一起，从而导致这部分实体涉及的三元组较少，进而导致无法充分对其进行训练。

表 7. 尾实体推理测试的实验结果（TransE_ipv）

data set	@hit_10	recall_hit_10	F1_hit_10	@hit_1	recall_hit_1	F1_hit_1
samll	38.49%	28.12%	32.50%	20.83%	15.22%	17.59%
all	26.69%	21.40%	23.75%	11.15%	8.94%	9.92%

为了验证可能是实体的数据的长尾无法充分训练，进而影响准确率，我们设计实验进行研究。在之前的训练中每次迭代为每个训练三元组构造一个三元组负例进行训练，为了缓解训练不充分的问题，改进算法在每次迭代中对每个训练三元组构造50个三元组负例进行训练（标记为TransE_1.1），使用相应的测试集进行尾实体推理测试，最后在small数据集上得到的实验结果如所示。

Table 8. 尾实体推理测试的实验结果（small数据集）

model	@hit_10	recall_hit_10	F1_hit_10	@hit_1	recall_hit_1	F1_hit_1
TransE_ipv	38.49%	28.12%	32.50%	20.83%	15.22%	17.59%
TransE_1.1	41.47%	27.34%	32.95%	27.76	18.30%	22.06%

实验结果显示`@hit_1`和`recall_hit_1`都有显著提升，可见尝试增加三元组负例的数量对尾实体推理有较好的影响。当大量增加三元组负例时，尾实体推理效果可能会得到大幅度提升，但限制于训练时间原因，本实验未继续增加三元组负例数量进行测试。

4 结束语

基于传统网状结构的知识库无法有效地进行知识推理，尤其当知识库的知识规模不断扩大，基于网状结构知识库的推理很难较好地满足实时计算的需求。因此，本文使用TransE模型对开放域中文知识库进行表示学习，并对模型的代价函数进行改进，主要研究基于知识库表示学习的知识推理，包括对实体关系三元组中关系指示词以及尾实体的推理。实验结果显示，基于知识库表示学习的关系指示词推理准确率可以达到80%以上，且无需设计复杂的算法。在进行

应用知识分布表示的尾实体推理测试中，准确率和关系指示词推理相比效果较差，我们对其原因进行分析并验证，使用增加训练过程中三元组负例的方法可以将准确率提升7个百分点，同样无需设计复杂算法即可实现对尾实体的推理。

参考文献

[1]. Miller G A. WordNet: a lexical database for English[J]. Communications of the ACM, 1995, 38(11): 39-41.

[2]. Bollacker K, Evans C, Paritosh P, et al. Freebase: a collaboratively created graph database for structuring human knowledge[C]//Proceedings of the 2008 ACM SIGMOD international conference on Management of data. ACM, 2008: 1247-1250.

[3]. Miller E. An introduction to the resource description framework[J]. Bulletin of the American Society for Information Science and Technology, 1998, 25(1): 15-19.

[4]. 刘知远, 孙茂松, 林衍凯, 等. 知识表示学习研究进展[J]. 计算机研究与发展, 53(2): 247-261.

[5]. Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2013, 35(8): 1798-1828.

[6]. Bordes A, Weston J, Collobert R, et al. Learning structured embeddings of knowledge bases[C]//Conference on Artificial Intelligence. 2011(EPFL-CONF-192344).

[7]. Bordes A, Glorot X, Weston J, et al. A semantic matching energy function for learning with multi-relational data[J]. Machine Learning, 2014, 94(2): 233-259.

[8]. Bordes A, Glorot X, Weston J, et al. Joint learning of words and meaning representations for open-text semantic parsing[C]//International Conference on Artificial Intelligence and Statistics. 2012: 127-135.

[9]. Socher R, Chen D, Manning C D, et al. Reasoning with neural tensor networks for knowledge base completion[C]//Advances in Neural Information Processing Systems. 2013: 926-934.

[10]. Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data[C]//Advances in Neural Information Processing Systems. 2013: 2787-2795.

[11]. Wang Z, Zhang J, Feng J, et al. Knowledge Graph Embedding by Translating on Hyperplanes[C]//AAAI. 2014: 1112-1119.

[12]. Lin Y, Liu Z, Sun M, et al. Learning Entity and Relation Embeddings for Knowledge Graph Completion[C]//AAAI. 2015: 2181-2187.

[13]. Ji G, He S, Xu L, et al. Knowledge Graph Embedding via Dynamic Mapping Matrix[C]//Proceedings of ACL. 2015: 687-696.

[14]. Turian J, Ratinov L, Bengio Y. Word representations: a simple and general method for semi-supervised learning[C]//Proceedings of the 48th annual meeting of the association for computational linguistics. Association for Computational Linguistics, 2010: 384-394.

基于字信息学习词汇分布的实体上位关系识别

刘 燊[1] 姜天文[2] 秦 兵[3] 刘 挺[4]

(哈尔滨工业大学 计算机科学与技术学院，哈尔滨 150001)

摘要. 本文在实体上位关系识别任务上，使用基于字信息的词向量学习模型学习词向量表示，并以此学习上位关系向量表示，在实体上位关系识别实验结果上效果较好，并且很大程度上缓解了未登录词的问题。首先基于字信息的词向量模型可以学习出几乎任意词语的词向量，然后根据语料中的上下位词对学习上位关系向量并聚类，学习每个簇的上位关系映射矩阵。最后利用上位关系映射矩阵来判别上位关系是否成立。实验结果表明，在未登录词多的数据集中，上位关系判别依然有着近80%的准确率，达到了可以应用的结果。

关键词: 类别层次化；开放域；上位关系；词汇分布

Learning Type Hierarchies for Open-domain Named Entities via Word Embeddings based on Chararter Information

Shen Liu [1], Tianwen Jiang[2], Bing Qin[3], Ting Liu[4]

(School of Computer Science and Technology, Harbin Institute of Technology, Harbin, 150001, China)

Abstract. We learn word embeddings based on character information to discover entity type hierarchies. Experiment results show that it is good to learn type hierarchies for open-domain named entities and mostly alleviates the unlisted words problem. We firstly use a model to learn word embeddings, which can almost learn word embedding of all words, and cluster the hypernym rela-

刘燊(1992-)，男，江西南康人，硕士研究生，主要研究方向：自然语言处理、关系抽取；
姜天文(1994-)，男，吉林辽源人，本科生，主要研究方向：自然语言处理、信息抽取；
秦兵(1968-)，女，陕西华阴人，博士，教授，博士生导师，主要研究方向：中文信息处理、信息抽取、篇章分析；
刘挺(1972-)，男，黑龙江哈尔滨人，博士，教授，博士生导师，主要研究方向：自然语言处理，文本挖掘，文本检索

tion vectors based on the hypernym-hyponym word pairs in training data. Secondly we train the mapping matrices of each cluster. Finally, we recognize the hypernym by using the hypernym mapping matrices. The experimental results show that hypernym recognition obtains almost 80% of the precision in the dataset which has lots of unlisted words, and the achievement can be applied.

Keywords: Type Hierarchy, Open-domain, Hypernym, Word Embedding

1 引言

传统领域命名实体主要分为3种：人名、地名、机构名。而在实际的自然语言处理应用中，传统意义上的命名实体是无法满足实际需求的。因此引入了开放域命名实体。相对于传统命名实体，开放域命名实体的类型更多也更细，很难通过人工定义类别体系。一种方法就是使用实体的上位词作为实体的类别。

上位词是一个语言学概念，它指语义范畴相对较广的词语[1]。例如“*美洲豹*”是一种“*动物*”，则“*动物*”就被称为“*美洲豹*”的上位词，因为在表达的含义上，“*动物*”的语义范畴更广，还包括了“*熊猫*”、“*狮子*”等。对于语义范畴更广的“*生物*”，“*动物*”则成为了“*生物*”的下位词。因此，命名实体的类别就可以认为是它的类别，并且类别往往也是有层级关系的。Suchanek等人[2]借助维基百科内容进行扩充和细化人工词典WordNet[3]的语义结构，但其只能覆盖维基百科本身的内容范围。Hearst[4]和Snow[5]等基于模式匹配的方法抽取上下位关系，但人工构建的模式仅能处理小部分语言现象，且费时费力，同时Snow等人[5]自动抽取模式的方法对句法分析和语料质量的要求很高，不容易应用到互联网等开放域语料中。随着深度学习的发展，大量研究基于词汇分布表示开始进行。

词汇分布（word embedding）表示，通常将词语表示成稠密且低维的实数向量，从而使得词语之间可以进行数学运算，如向量的加减等。实验表明，使用这样的实数向量表示的词语可以保留语言的规律性，可以用于计算词语之间的关系[6]。例如在Mikolov等人[7]的实验中，观察到了$v(\texttt{king})-v(\texttt{queen})\approx v(\texttt{man})-v(\texttt{women})$的现象，其中$v(\mathrm{w})$表示词语w的词向量。Fu等人[8]受到启发得到$v(\text{上位关系})\approx v(\text{上位词})-v(\text{下位词})$。

在词汇分布表示时，一个重要问题就是未登录词是没有词向量的，通常情况下所学习出的词向量表示都是基于训练数据中出现过的词构成的词表进行学习的。因此在使用词向量进行上层应用的时候，对于没有词向量的未登录词是无法处理的。Wang等人[9]使用bi-LSTM[10]（bidirectional LSTM，双向LSTM）基于字信息学习词向量表示，在形态丰富的语言上有着更好的学习效果，在语言模型与词性标注任务上获得了不错的结果。由于其是基于字信息学习词向量表示，因此只要词语中的字是在训练数据中出现了的，就可以学习词语的词向量表示。

本文使用基于字信息学习的词向量进行实体上位关系识别，首先基于字信息的词向量模型学习出几乎任意词语的词向量，然后根据语料中的上下位词对学习上位关系向量并聚类，学习每个簇的上位关系映射矩阵。最后利用上位关系

映射矩阵来判别上位关系是否成立。在未登录词多的数据集中，上位关系判别依然有着近80%的准确率。对于常规词向量学习模型中未登录词的词汇分布表示问题有着较好的解决方法，并达到了可以应用的性能。

2 基于字信息的词向量学习模型

词向量表示，最普通的方法就是使用一个词表V来表示所有的词语，那么具体地，一个词w的词向量表示可以使用独热向量（one-hot vector）来表示，即词向量维度为|V|，每个维度代表一个词语，除了w所在的维度数值为1，其余每一维都为0。例如词表V={我,爱,吃,苹果}，|V|=4，则“我”的词向量v(我)=[1,0,0,0]，“吃”的词向量v(吃)=[0,0,1,0]。这种方法所表示的词向量没有词语之间的语义信息，并且无法比较词语之间的关系。

Mikolov等人[7]提出的CBOW和Skip-gram模型的主要思路为通过设置一个固定大小的窗口，通过词语的上下文窗口信息来学习词语的词向量表示。这一类方法都是将词语作为最小单位进行学习的。对于“cats”和“cat”、“kings”和“king”都是分别对待的，即没有利用词语本身的字信息。

Wang等人[9]提出的C2W（character to word）模型基于双向LSTM学习词向量，通过学习字之间的信息来组合成词向量的表示。双向LSTM可以学习出序列模型中的非局部依赖信息。

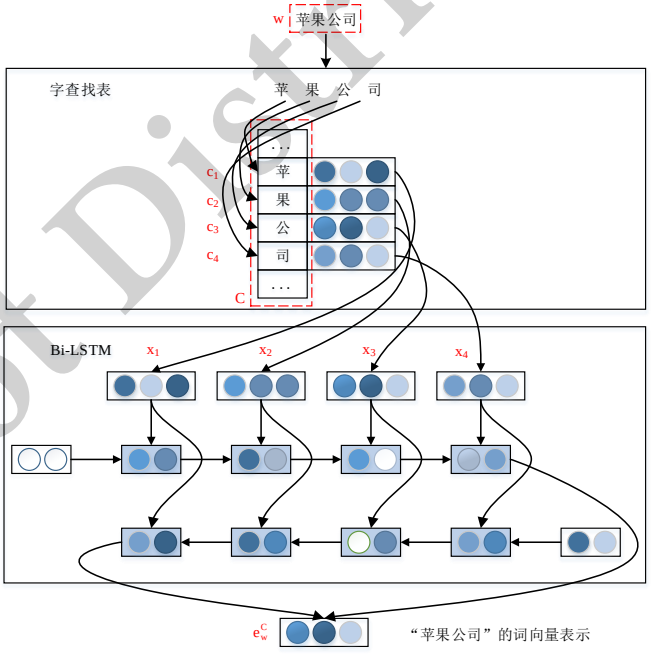


图 1. C2W模型框架图

C2W模型框架如图 1所示，以*“苹果公司”*作为输入词为例。C2W模型的输入为一个词w，我们所希望获得的就是d维的词向量用于表示w。

作为模型的输入，我们定义一个字表C。输入的词w使用其字序列$c_1,\dots,c_m$表示，其中m为词w的字长度。每一个c_i定义为一个独热向量$\mathbf{1}_{c_i}$，字c_i在字表C中对应下标位置为1。我们定义投影层$\mathbf{P}_C \in \mathrm{R}^{d_c \times |C|}$，其中$d_C$为每个字在字集合C中的参数个数。因此，对于每个输入的字c_i的投影有$\mathbf{e}_{c_i} = \mathbf{P}_C \cdot \mathbf{1}_{c_i}$。我们给“*苹果公司*”的字序列获取其4个字的独热向量，并使用投影层$\mathbf{P}_C$获得4个输入向量作为LSTM的输入。

给定输入向量$\mathbf{x}_1,\dots,\mathbf{x}_m$，LSTM 迭代计算状态序列$\mathbf{h}_1,\dots,\mathbf{h}_{m+1}$如下：

$$
\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_{ix}\mathbf{x}_t + \mathbf{W}_{ih}\mathbf{h}_{t-1} + \mathbf{W}_{ic}\mathbf{c}_{t-1} + \mathbf{b}_i)\\
\mathbf{f}_t &= \sigma\left(\mathbf{W}_{fx}\mathbf{x}_t + \mathbf{W}_{fh}\mathbf{h}_{t-1} + \mathbf{W}_{fc}\mathbf{c}_{t-1} + \mathbf{b}_f\right)\\
\mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tanh(\mathbf{W}_{cx}\mathbf{x}_t + \mathbf{W}_{ch}\mathbf{h}_{t-1} + \mathbf{b}_c)\\
\mathbf{o}_t &= \sigma(\mathbf{W}_{ox}\mathbf{x}_t + \mathbf{W}_{oh}\mathbf{h}_{t-1} + \mathbf{W}_{oc}\mathbf{c}_t + \mathbf{b}_o)\\
\boldsymbol{h}_t &= \mathbf{o}_t \odot \tanh(\mathbf{c}_t)
\end{aligned}
$$

其中，σ为分量 sigmod 函数，⊙为分量阿达马（Hadamard）积。LSTM 定义额外的存储单元$\mathbf{c}_t$用于线性组合每个时间点 t 的结果。从$\mathbf{c}_{t-1}$到$\mathbf{c}_t$的信息传输由 3 个门控制：$\mathbf{i}_t$，$\mathbf{f}_t$和$\mathbf{o}_t$。$\mathbf{i}_t$为输入门，决定从输入$\mathbf{x}_t$所包含的信息；$\mathbf{f}_t$为遗忘门，决定遗忘从$\mathbf{c}_{t-1}$来的信息；$\mathbf{o}_t$为输出门，决定对于当前状态$\boldsymbol{h}_t$相关的信息。我们使用$\mathbf{P}$表示 LSTM 中的所有参数，如$\mathbf{W}_{ix}, \mathbf{W}_{fx}, \mathbf{b}_f$等。对于输入的字表示序列$\mathbf{e}^C_{c_1},\dots,\mathbf{e}^C_{c_m}$，前向 LSTM 输出状态序列$\mathbf{s}^f_0,\dots,\mathbf{s}^f_m$，反向 LSTM 则将前向 LSTM 输入的字表示序列反向作为输入，然后输出状态序列$\mathbf{s}^b_0,\dots,\mathbf{s}^b_m$。2 种 LSTM 使用不同的参数集合，其中前向 LSTM 使用$\mathbf{P}^f$，反向 LSTM 使用$\mathbf{P}^b$。词 w 的向量表示通过整合前向和后向的状态来获得：

$$\mathbf{e}^C_w = \mathbf{D}^f \mathbf{s}^f_m + \mathbf{D}^b \mathbf{s}^b_0 + \mathbf{b}_d$$

其中，$\mathbf{D}^f, \mathbf{D}^b$和$\mathbf{b}_d$为决定状态组合方式的参数。

最后，我们使用 C2W 模型获得了“苹果公司”的词向量。我们使用 C2W 模型来学习字信息，然后基于字信息重组词向量，从而达到基于字信息学习词向量的效果。

2.1 上位关系向量表示

我们分别使用word2vec[5]和C2W模型[6]训练词向量。

在Mikolov等人[7]的实验中，观察到了v(king)−v(queen)≈v(man)−v(women)的现象，其中v(w)表示词语w的词向量。从这个例子可以看出，两个向量之间的向量差值可以表达出词对之间一定的语义信息。

[5] https://code.google.com/archive/p/word2vec/
[6] https://github.com/wlin12/JNN

在上下位关系中，也观察到了类似了的性质。Fu等人随机选取了一些上下位词对，同样使用向量差值表达语义关系，结果如表 1所示[8]。

表 1. 上下位词对的词分布向量偏移

序号	实例
1	*v*(虾)−*v*(对虾)≈*v*(鱼)−*v*(金鱼)
2	*v*(工人)−*v*(木匠)≈*v*(演员)−*v*(小丑)
3	*v*(工人)−*v*(木匠)≠*v*(鱼)−*v*(金鱼)

可见，前两个实例表明上下位关系也是可以通过词向量的向量差值近似表达的。而第三个实例则说明了上下位关系更加复杂，无法简单地使用一个上下位关系向量来表达。

我们假设通过v(上位词)−v(下位词)近似可以得到上位关系向量v(上位关系)。假设所有的词都能通过一个矩阵映射到其上位词。给定一个词的词向量表示x和它的上位词向量y，存在一个矩阵Φ使得$\mathbf{y} = \Phi\mathbf{x}$。

通过最小化均方误差求解下位词到上位词的映射矩阵：

$$\Phi^* = \arg\min_{\Phi} \frac{1}{N} \sum_{(\mathbf{x},\mathbf{y})} \|\Phi\mathbf{x} - \mathbf{y}\|^2$$

其中N为训练数据中上下位词对$(\mathbf{x}, \mathbf{y})$的数量。这是一个线性回归问题，优化算法使用随机梯度下降法。

在进一步的数据观察发现，上位关系仍然可继续细分，因为上下位关系是一个多对多的关系。一个具体的下位词往往有多个上位词。因此无法使用单一的映射矩阵来刻画上位关系，需要对每一个上位关系向量簇学习一个矩阵映射：

$$\Phi_k^* = \arg\min_{\Phi_k} \frac{1}{N_k} \sum_{(\mathbf{x},\mathbf{y}) \in C_k} \|\Phi_k\mathbf{x} - \mathbf{y}\|^2$$

其中N_k表示第k个簇C_k中上下位词对的个数。我们使用k-均值（k-means）算法对上位关系进行聚类获得上下位关系簇。

2.2 上位关系识别

我们在对上下位关系进行聚类后，对训练数据中的每一个上下位关系簇C_k学习一个向量矩阵Φ_k。对于下位词向量x和上位词向量y，我们先找出距离y-x向量最近的上下位关系簇Φ_k。既然已经聚类了上下位关系，对于上位关系的识别，可以使用距离度量来计算所得的上下位关系向量是否属于聚类后的上下位关系中的一类。同时，上位关系显然是存在传递性的。那么对于上位关系识别，如果y是x的上位词，则需要满足以下两个条件之一。

条件1：通过映射矩阵Φ_k使得$\Phi_k\mathbf{x}$尽可能接近y。设$d(\Phi_k\mathbf{x},\mathbf{y})$表示$\Phi_k\mathbf{x}$与y之间的欧氏距离，则应满足：

$$d(\Phi_k\mathbf{x},\mathbf{y}) = \|\Phi_k\mathbf{x}-\mathbf{y}\|^2 < \delta$$

其中δ为距离阈值。

条件2：上位关系的传递性。存在一个词z，满足$x \overset{H}{\rightarrow} z$且$z \overset{H}{\rightarrow} y$。

其中$x \overset{H}{\rightarrow} z$表示词x为词z的下位词，词z为词x的上位词。

正常的上下位关系是一个有向无环图。而通过映射矩阵所得的上下位关系是可能存在环的。因此，在上下位关系中出现环时，我们删除置信度较低的那一条边，即如果$d(\Phi_j\mathbf{y},\mathbf{x}) > d(\Phi_k\mathbf{x},\mathbf{y})$，则删除从y指向x的边。

3 实验结果与结论分析

通常情况下，我们需要大量的语料来训练词向量，这样才能够较好的学习词向量的表示，充分利用词语的上下文信息。

我们使用百度百科[7]中文语料训练词汇分布。百度百科中文语料共包含100多万百科词条，共约3000万句，文件大小4GB左右。我们先将语料进行中文分词，使用哈尔滨工业大学社会计算与信息检索研究中心发布的语言技术平台[11]（LTP，Language Technology Platform）进行分词。分别使用word2vec和C2W模型将百度百科语料中的正文分词后文本作为训练语料获得词向量，词向量维度设置为300。

其中word2vec使用Skip-gram模型进行训练，获得了约56万中文词汇的词向量。在C2W模型中，设置字信息维度为300维，LSTM状态向量维度为150维，字表大小根据字出现频率从高到低限制为1万，学习结果的词向量维度为300维。

使用C2W模型训练所得的词向量，其中部分词的词向量最近5个词结果如表 2所示。

表 2. C2W模型训练所得词向量部分词语最近5个词结果

词语	相似度	词语	相似度	词语	相似度
中国	1.0000	**北京**	1.0000	**清华大学出版社**	-
德国	0.8379	南京	0.9569	出版社	0.7924
美国	0.8144	东京	0.9371	高等学校	0.7742
泰国	0.8134	南北	0.7959	清华大学	0.7664
大国	0.7935	东北	0.7832	师范学院	0.7626
爱国	0.7886	南海	0.7830	理工大学	0.7564

[7] 百度百科（http://baike.baidu.com）是最大的中文在线百科知识库之一。

其中加粗部分词语为查询词，其余词为在训练语料中出现过并使用cosine距离计算得到的最相似的5个词；相似度即为cosine相似度。

从表 2中可以看出，C2W模型所得到的词向量基于字信息学习出了词向量表示并且带有语义信息，例如*“中国”*和*“北京”*的最近5个词在字面上都与查询词有着紧密的联系，其中*“中国”*的最近几个词都与国家相关，并且在字面表达形式上都以*“国”*结尾，*“北京”*的最近几个词都与城市或方位相关。查询词“清华大学出版社”在训练语料中是没有出现过的，即为未登录词，但其最近的5个词也表达出了语义上的相似性，即*“出版社”*作为核心词相似度最高，同时*“清华大学”*的相似度也很高。表 2说明了C2W模型是可以基于字信息学习出带有一定语义性的词汇分布表示的，并且对于未登录词仍然可以学习出带有语义性的词向量表示。

3.1 上位关系簇聚类

上下位关系簇聚类使用《同义词词林》抽取所得的上下位关系词对数据进行，并随机选取聚类后的每个簇的1/10作为该簇的映射矩阵学习的开发集，数据结果统计如表 3所示。

表 3. 上位关系簇训练数据结果统计

关系类型	训练集	开发集	总计
上位-下位关系对词对数	13,718	1,524	15,242

3.2 上位关系识别

数据集使用2个数据集：①Fu等人从《同义词词林（扩展版）》[8]（Tongyi Cilin (Extented)）中处理所得上下位词对[8]；②从《大词林》中已有的数据中分别随机抽取了500个实体及其上位词并进行人工标注确认所得上下位词对。数据集统计结果如表 4所示。这两个数据集为上位关系识别的测试数据。

本文中将《同义词词林（扩展版）》简称为《同义词词林》，英文简称CilinE。

表 4. 上位关系识别数据统计

关系类型	《同义词词林》数据集	《大词林》数据集
上位-下位关系词对数	2,158	752
无关系词对数	3,250	1,864
总计词对数	5,408	2,590

[8] http://www.ltp-cloud.com/download/

其中，《大词林》数据集主要分两部分数据，一部分为开放域命名实体与其上位词之间的上下位关系，另一部分为类别词之间的上下位关系。

统计发现，在《大词林》数据集中，使用word2vec所获得的词向量在计算开放域命名实体与上位词之间的上下位关系时，其中77.39%的上下位关系包含未登录词[9]；在计算类别词之间的上下位关系时，其中15.83%的上下位关系包含未登录词。即便对从《大词林》中随机抽取的开放域命名实体和类别词进行分词后重新组合构成词向量，分别仍然有33.51%和11.15%的上下位关系包含未登录词。如果抛弃无法判断的上下位关系不参与最后结果统计，实验结果如表 5所示。

表 5. 使用word2vec在《大词林》数据集进行上位关系识别实验结果

数据	词向量处理方式	未登录词比例	P	R	F1
实体与类别词	无	77.39%	1.0000	0.1607	0.2769
	Avg	33.51%	0.8909	0.3952	0.5475
	Min		0.9787	0.3710	0.5380
	Max		0.9778	0.3548	0.5207
类别词之间	无	15.83%	0.9683	0.3836	0.5496
	Avg	11.15%	0.8289	0.3851	0.5250
	Min		0.9688	0.3780	0.5439
	Max		0.9683	0.3720	0.5374

其中，avg表示在对上位词或下位词进行分词后，将分词后的每个词对应的词向量求和取平均作为原始词的词向量表示，例如“哈尔滨工业大学”分词后为*“哈尔滨”*、*“工业”*和*“大学”*，则*“哈尔滨工业大学”*的词向量表示即为*“哈尔滨”*、*“工业”*和*“大学”*3个词的词向量求和取平均而得。同理，min和max分别表示将分词后的每个词对应的词向量求和取最小值或者最大值作为原始词的词向量表示。

结合未登录词比例统计结果数据，从表 5可以看出：

- 未登录词所占比例较大，特别是开放域命名实体与类别词上下位关系部分，大部分的开放域命名实体都是没有对应的词向量的，如*“伍氏锯鳞鱼”*和*“镰苞鹅耳枥”*等。对于未登录词的情况，使用C2W模型则可以学习出对应的词向量表示；
- 对于原始词语进行分词处理后也还是存在一定量的未登录词，主要因为开放域命名实体即使在分词后，仍有不少的词语是较少见的词语，即分词后仍然会出现未登录词；
- 对于原始词语进行分词处理前后的上位关系识别准确率都较高，基本大于80%，对于部分结果甚至高于95%，即如果判断一条上位关系成立，那么这

[9] 上位关系中若上位词或下位词其中之一为未登录词（没有词向量表示）时，则无法判断此关系。

条上位关系很可能确实成立。因此从准确率的角度上看，已经可以达到应用的要求。

- 上位关系识别的召回率普遍较低，这可能是因为开放域命名实体与上位词之间的上位关系较复杂，导致相当一部分的上位关系没有通过聚类学习出来，这也与当前使用的训练语料较小有关，没有足够数据表达出对应的上位关系。

由于未登录词的问题在实际应用情况中频繁出现，因此使用C2W模型重新训练了基于字信息的词向量。使用C2W模型学习所得词向量作为获得上位关系向量的来源，尝试调整上位关系向量聚类数目*K*对结果产生的影响如图 2所示。

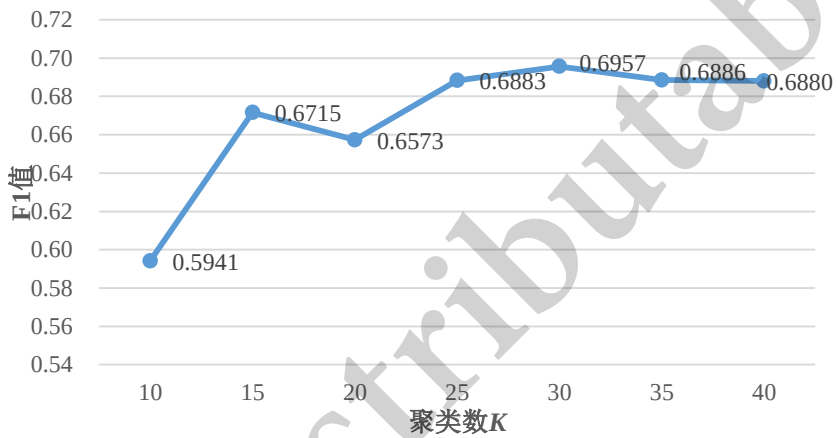


图 2. 聚类数*K*对上位关系识别影响

从图 2中可以看出，聚类数目在30附近时，上位关系识别的结果获得了最好的结果，因此在聚类数30附近进行了微调，得到聚类数*K*=31时得到最好结果。

实际上，我们所设置的聚类数较小时，会导致相当一部分并不是一类上下位关系的结果聚类到了一起。以*K*=20为例，其中的上下位关系：*木匠* $\xrightarrow{H}$ *工人*和*金鱼* $\xrightarrow{H}$ *鱼*，并不是一类上下位关系，但是却被聚类到了一起。而在设置的聚类数较大时，部分类的上下位关系数会很少，导致映射矩阵的学习结果较差，并且同样会导致部分上下位关系类被聚成了两类或多类。一定程度上而言，这个与我们所使用的语料是相关的。

同时，也在聚类数目为31时，对距离阈值δ进行了调整，如图 3所示。

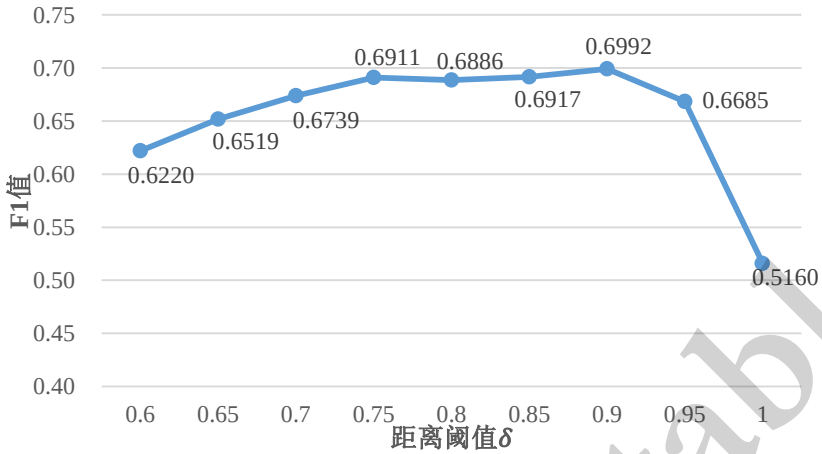


图 3. 距离阈值δ对上位关系识别结果影响

其中，距离阈值$\delta = 0.9$时，上位关系识别获得了最好结果。

因此，在本实验中，使用C2W模型学习所得词向量进行上位关系识别时，最佳参数聚类数目K=31，距离阈值$\delta = 0.9$。

使用C2W模型所得的词向量进行上位关系识别结果与word2vec所得词向量结果如表 6所示。

表 6. 不同词向量和方法的上位关系识别结果

测试数据集	词向量来源	方法	P	R	F1
《同义词词林》数据集	word2vec	M_{Emb}	0.8054	0.6799	0.7374
		$M_{Emb+CilinE}$	0.8059	0.7242	0.7629
		$M_{Emb+CilinE+Wiki}$	0.7978	0.8081	0.8029
	C2W	M_{Emb}	0.7882	0.6282	0.6992
		$M_{Emb+CilinE}$	0.8015	0.6891	0.7411
		$M_{Emb+CilinE+Wiki}$	0.7839	0.7565	0.7700
《大词林》数据集	word2vec	M_{Emb}	0.7609	0.2369	0.3613
		$M_{Emb+CilinE}$	0.7500	0.4772	0.5832
		$M_{Emb+CilinE+Wiki}$	0.7717	0.4805	0.5923
	C2W	M_{Emb}	0.9449	0.3191	0.4771
		$M_{Emb+CilinE}$	0.7927	0.5798	0.6697
		$M_{Emb+CilinE+Wiki}$	0.7935	0.5824	0.6718

其中M_{Emb}的方法为仅使用词向量进行上位关系识别的结果。$M_{Emb+CilinE}$的方法为在词向量基础上，融合《同义词词林》的结果，即将两种方法所获得的上位关系的正例简单合并，使用合并后的结果作为融合后的方法的融合结果。同理，

$M_{Emb+CilinE+Wiki}$方法的结果为融合了词向量、《同义词词林》和中文维基百科[10]所得正例合并的结果。

从表 6中可以看出，在《同义词词林》数据集中，使用word2vec所得结果优于C2W模型所得结果，分析原因为在语义关系的学习上，word2vec比C2W模型更好，因为word2vec在学习的过程中更注重上下文信息，对于一个词的词向量的学习会与其上下文相关，而C2W模型则更注重词语的字结构信息来学习词向量表示（从“*中国*”这个例子就可以发现，与“*中国*”相近的词语大多以“*国*”字结尾）。在《大词林》数据集中，使用C2W模型所得结果更优，原因则为C2W对于未登录词仍然可以学习出较好的词向量，很大程度上解决了未登录词的问题，例如“加拉帕戈斯群岛”在训练词向量的语料中并没有出现，即便分词后仍然存在未登录词，而C2W模型则学习出了其对应的词向量，并且在上位关系识别中正确识别出了上位关系：加*拉帕戈斯群岛* $\xrightarrow{H}$ *群岛*。并且使用C2W模型所得词向量，在不同数据集上的准确率变化不大，都在80%左右，较稳定。

4 结束语

针对词向量应用中的未登录词问题，本文使用C2W模型在百度百科语料上学习了一个基于字信息的词向量学习模型。使用C2W模型所学习得到的词向量在上位关系识别任务上使用《同义词词林》所得上下位词对数据集上与word2vec所得效果相当，略低于word2vec。未来可以将word2vec与C2W相结合，既缓解未登录词的问题，在词向量的学习上也能够更好地学习词语的语义信息。在《大词林》中所得上下位关系对数据中，由于包含较多的开放域命名实体，因此未登录词较多。C2W模型在《大词林》所得数据中，对于未登录词仍然可以较好地学习出词向量，上位关系识别结果优于使用word2vec所得结果，很大程度上缓解了未登录词的词向量学习问题。

参考文献

[1]. 付瑞吉. 开放域命名实体识别及其层次化类别获取[D]. 哈尔滨工业大学, 2014.

[2]. Suchanek F M, Kasneci G, Weikum G. Yago: A large ontology from wikipedia and wordnet[J]. Web Semantics: Science, Services and Agents on the World Wide Web, 2008, 6(3): 203-217.

[3]. Miller G A. WordNet: a lexical database for English[J]. Communications of the ACM, 1995, 38(11): 39-41.

[4]. Hearst M A. Automatic acquisition of hyponyms from large text corpora[C]//Proceedings of the 14th conference on Computational linguistics-Volume 2. Association for Computational Linguistics, 1992: 539-545.

[5]. Snow R, Jurafsky D, Ng A Y. Learning syntactic patterns for automatic hypernym discovery[J]. Advances in Neural Information Processing Systems 17, 2004.

[10] https://dumps.wikimedia.org/zhwiki/20131205/，主要使用其中的开放分类信息。

[6]. Mikolov T, Yih W, Zweig G. Linguistic Regularities in Continuous Space Word Representations[C]//HLT-NAACL. 2013: 746-751.
[7]. Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[C]. In Proceedings of Workshop at ICLR, 2013.
[8]. Fu R, Guo J, Qin B, et al. Learning Semantic Hierarchies via Word Embeddings[C]//ACL (1). 2014: 1199-1209.
[9]. Ling W, Luś T, Marujo L, et al. Finding function in form: Compositional character models for open vocabulary word representation[C]. EMNLP, 2015.
[10]. Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures[J]. Neural Networks, 2005, 18(5): 602-610.
[11]. Che W, Li Z, Liu T. Ltp: A chinese language technology platform[C]//Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations. Association for Computational Linguistics, 2010: 13-16.

基于混合模型的电子产品属性值识别

邵元新，白宇，张桂平

（沈阳航空航天大学，人机智能研究中心，辽宁，沈阳 110136）

摘要. 针对电子产品种类繁多，属性值多样化的特点，提出了一种基于混合模型的电子产品属性值识别方法。该方法根据属性的特点，将其分为通用属性和专用属性两类，对于前者，因其具有良好的规律，故采用基于规则的方法，对于后者，由于不同产品之间的差异性较大，采用了一种两阶段的方法，即在边界检测阶段采用条件随机场模型；在类别判定阶段采用支持向量机模型。实验表明，对于通用属性，基于规则的方法不仅可以减少人工标注的任务量，而且能提升识别结果；对于专用属性，本文在边界检测基础上又进行了边界后处理工作，使边界检测的结果得到了进一步的优化。最后，本文采用的混合模型融合了规则、边界后处理以及 CRF 与 SVM 的优势，在 F 值达到 0.9417 的同时模型的训练效率也得到了很大的提升。

关键词: 属性值识别，边界检测，条件随机场，支持向量机

Electronic Products Attribute Value Recognition Based on Hybrid Model

SHAO Yuanxin, BAI Yu, ZHANG Guiping

(Research Center for Human-Computer Intelligence，Shenyang Aerospace University，Shenyang，Liaoning 110136，China)

Abstract. According to the characteristics of electronic products of wide ranges and diversified attribute value,proposes an attribute value recognition method based on hybrid model for electronic products.This method according to the characteristics of the attribute,divided it into two categories,namely general and special attribute.For the former one,due to its great regularity,the rule-based method is adopted,For the later one,because of the differences among various products,we adopt a two-stage method,at the stage of boundary detection,we use the CRF model,while at the stage of category determination,we use the SVM model.The experimental results indicate that to the general attribute the rule-based method can not only reduce the quantity of manually annotation but also improve the recognition result.And to the special attribute we conduct the post-processing work based on the boundary detection result,so that the boundary detection result optimized further.At last,our hybrid model fusion the advantage of rule,post-processing of boundary and CRF,SVM,F-Measure achieves 94.17%,meanwhile the training efficiency of the model got a great improvement.

Keywords:attribute value recognition, boundary detection, conditional random field, support vector machine

1 引言

近年来，伴随着网络的快速发展与普及，互联网已成为人们获取知识和信息的重要途径，特别是随着互联网+时代的到来，无论是企业还是个人对于网络资源的依赖都将显得尤为凸出。目前，网络上关于电子产品的资源很多，涉及到在线百科，垂直网站和电商网站等，如何将这些不同的资源整合到一起，并且将产品与产品之间，产品与企业之间以及产品及其属性之间的关系进行梳理，绘制出一个电子产品的知识图谱，这无论是对于消费者进行产品的横向和纵向对比，还是对于企业把握产品趋势，进行商业决策都有重要的意义。

知识图谱（Knowledge Graph）于 2012 年 5 月由 Google 提出，随后国内也掀起了研究知识图谱的热潮并在产业界得以应用，如百度的“知心”，搜狗的“知立方”等。知识图谱本质上是一种语义网络，其结点代表实体（entity）或者概念（concept），边代表实体与概念之间的各种语义关系。知识图谱的构建主要包括知识单元的构建、知识单元间关系的构建和知识的可视化三个部分，其中前两个部分是构建知识图谱的最基本任务，可以分别映射为实体识别和实体关系的抽取两个子任务[1]。针对电子产品更新速度快，网络上结构化数据不够全面的特点，本文着手从非结构化文本中进行电子产品属性值的识别，并以手机领域为例。

2 相关工作

属性值识别任务与实体识别任务相似，因此可以借鉴实体识别研究的方法。目前，实体识别方法主要有三种：（1）基于规则的识别方法（rule-based），在实体识别研究的开始阶段，基于规则的方法占主导地位，一个成功的基于规则的命名实体识别框架是 AutoSlog 信息抽取系统[2]。基于规则的方法有相对精确的识别效果，但是其覆盖面窄，只能应用于较小的领域内且系统的移植性差。（2）基于统计的识别方法（statistic-based），基于统计的方法利用人工标注的语料进行训练，而语料不需要特定领域的专家参与就可以标注完成。更重要的是，统计的方法系统移植性强，只需要用新领域的语料进行训练即可完成。目前，常用到的基于统计的模型有：隐马尔科夫模型（Hidden Markov Model，HMM）[3] 、支持向量机（Support Vector Machine，SVM）[4]、决策树（Decision Tree）[5]、最大熵模型（ Maximum Entropy Model，ME）[6]、条件随机场模型（ Conditional Random Fields Model，CRF）[7，8]。（3）规则和统计相结合的方法，规则与统计相结合的方法能够综合规则和统计的优点，一度受到研究者们的青睐，如闫萍[9]针对中文命名实体中人名的自动识别问题，使用统计与规则相结合的算法，克服了规则或统计单一方法的缺点，同时引入了互信息的算法对人名产生的交集歧义进行识别，实验结果表明此方法对人名有较好的识别能力，识别效率较平均水平也有较大的提高。

在特定领域方面，毛存礼等针对有色金属领域产品名、组织机构名、矿产名、地名这 4 类实体识别任务面临分词准确率不高、缺乏大量已标注的训练样本等问题，提出一种基于深度神经网络(DNN)架构的有色金属领域实体识别方法[10]，实验结果表明，提出的方法对于专业领域的实体识别具有较好的效果；邹涛根据电子产品领域语料的特点，提出了一种层叠模型将基于规则和基于统计方法结合起来的一种电子产品领域命名实体识别方法[11]，将基于规则识别后的结果作用于基于统计识别模块，在一定程度上避免了分词和训练语料稀疏等问题，提高了识别的准确率和召回率。

两阶段方法方面，何楠等针对中文命名实体识别任务提出了一种两阶段的方法[12]，第一阶段应用条件随机场（CRF）模型检测实体边界，第二阶段应用最大熵模型（ME）识别实体类别。与一阶段的方法相比在仅损失 1%的性能下，将计算复杂性降低了 80%以上；李芳提出了一种基于条件随机场的两阶段中文微博命名实体识别方法[13]，在不同阶段的条件随机场模型中，设置不同的特征模板，在提高命名实体识别效果的同时，有效减少了系统训练的时间。

通过对相关工作方法的研究并结合自己任务的特点，本文对电子产品专用属性的属性值识别，采用了两阶段的方法，对于通用属性的属性值识别，采用了基于规则的方法。

3 基于混合模型的电子产品属性值识别

3.1 数据预处理

3.1.1 文本处理

中文文本不像英文那样每个单词之间有空格分隔，所以对中文文本进行处理的第一步就是分词，其中分词是基于领域词典进行，另外在大多数的实体识别任务中都需要用到词性特征，因此需要对分词后的文本进行词性标注，分词和词性标注均采用中科院的 ICTCLAS 系统完成，此外文本中经常出现的形如“的”、“了”、“吗”等之类的词还有一些标点符号，它们在文本中出现的频率非常高，但对于我们的识别任务却是无关紧要的，所以我们要将其从文本中剔除。这不仅节省了存储空间，而且减少了后期训练模型的时间，文本处理的示例如下表 1 所示。

表 1 文本处理

原句	华硕在巴西发布低端新机。
分词结果	华硕 在 巴西 发布 低端 新机 。
词性标注结果	华硕/n 在/p 巴西/nsf 发布/v 低端/n 新机/n 。/wj
去除停用词后	华硕/n 巴西/nsf 发布/v 低端/n 新机/n

3.1.2 定义属性

针对电子产品领域的特点，将其属性分为通用属性和专用属性两类，通用属性是任何电子产品都具有的，并且它们在写法或是后缀单位上没有区别，如无论对于哪个电子产品，价格的表示方法总是 xx 元等。专用属性不是所有电子产品共有的属性，如手机的属性通常有内存和像素，而笔记本电脑的属性通常有硬盘存储容量，显卡的类型等。由于实验是在手机语料上进行，最终本文定义的电子产品通用属性有价格（PRI）、重量（WEI）、颜色（COL）和产品尺寸（SIZ）4 种。专用属性有品牌（BRA）、型号（TYP）、电池容量（BAT）、屏幕尺寸（SCA）、运行内存（RUV）、操作系统（OPS）、像素（PIX）、核心数（COR）、屏幕分辨率（SCR）、版本（VER）10 种。

3.1.3 标记设置

在对专用属性进行属性值边界检测时，需要先对语料进行人工标注，本文采用 BIESO 标注准则人工标注语料，其中 B 代表当前词是属性值的开头，I 代表当前词是属性值的中间，E 代表当前词是属性值的结尾，O 代表当前词是非属性值，S 代表单独的一个词就是属性值。如在“华为/S Mate/B 8/E 确定/O 将/O 于/O 11 月/O 26 日/O 在/O 上海/O 发布/O”这句话中共有两个属性值一个是品牌值“华为”，一个是型号值“Mate 8”。最后将语料处理为 CRF 所需的格式如下表 2 所示：

表 2 标注样例

当前词	当前词词性	标注集
雷军	nr	O
在	p	O
红米	n	B
Note	x	I
3	m	E
发布会	n	O

3.2 电子产品专用属性的属性值识别

3.2.1 基于 CRF 的属性值边界检测

针对电子产品领域的专用属性，其属性值边界检测可以被视为一个序列标注任务，鉴于条件随机场模型中的特征选择较为灵活变通，并且具有强大的特征融合能力，它没有隐马尔科夫模型那样强的独立性假设，同时也解决了最大熵模型中标记偏置问题，在序列标注任务中取得了很好的效果[14]，故本文采用 CRF 模型来完成边界检测任务。

3.2.1.1 边界检测特征的选取

在进行 CRF 特征选取的时候，随着选择特征数量的增加，数据集的训练时间将会呈现出数量级的增长。基于提高训练效率和减少特征冗余两方面的考虑，本文在特征模板的制定方面，充分研究了前人的经验，并通过实验对比，权衡了各种模板的效率。最终本文选取的特征主要包括词本身特征、词性特征、上下文窗口词特征、上下文窗口词词性特征等属性值内部和外部特征的集合。条件随机场可以引入很多外部特征来增强边界检测的效果如属性值的前后缀单位信息构成的外部词表特征，但是根据前人对电子产品属性值识别的研究经验[11]，由于电子产品领域内文本的特点，更多的外部特征没有增强识别效果，而仅仅增加了训练模型的时间和人力标注成本，因此，本文没有采用更多的外部特征。最终本文选取的边界检测特征如下表 3 所示。

3.2.2 边界检测后处理

属性值识别第二阶段的目标是给已经识别出边界的属性值进行类别的判定，属性值的分类效果依赖于边界检测结果的好坏。因此为了进一步优化边界检测的结果，本文在借鉴前人经验[15]的基础上，提出了基于规则的边界后处理方法。具体的做法就是用验证集在边界检测模型上做测

试，通过对检测错误的结果进行分析，总结规律。如通过测试发现手机版本的属性值边界检测结果比较差，这主要是由于手机的版本多样化造成的，但是通过分析可以发现构成手机的版本具有一定的规律，如：它们中一般都会带有“版”字，而且其构成词也相对集中，大部分为“移动4G 版”、“港版”、“双网通版”、“美版”，以及与手机的品牌和系列这类与运营商网络，地域名称和品牌系列相关的词。因此，本文对于这类错误总结规律，收集版本的前缀词表，手机的品牌系列词表，并利用词性信息制定规则。后处理示例如下表 4 所示：

表 3 边界检测特征

模板内容	模板释义
$C_{[-1,0]}$ $C_{[0,0]}$ $C_{[1,0]}$ $C_{[2,0]}$	$C_{[0,0]}$表示当前词，$C_{[n,0]}$表示当前词的右（n>0）/左（n<0）边第 n 个词
$C_{[-1,0]}/C_{[-1,1]}$ $C_{[0,0]}/C_{[0,1]}$, $C_{[1,0]}/C_{[1,1]}$ $C_{[2,0]}/C_{[2,1]}$	$C_{[0,0]}/C_{[0,1]}$表示当前词与词性的组合，$C_{[n,0]}/C_{[n,1]}$表示当前词的右（n>0）/左（n<0）边第 n 个词与词性的组合
$C_{[-1,0]}/C_{[0,0]}$ $C_{[0,0]}/C_{[1,0]}$, $C_{[0,0]}/C_{[1,0]}/C_{[2,0]}$	相邻词的组合
$C_{[-1,1]}/C_{[0,1]}$ $C_{[0,1]}/C_{[1,1]}$, $C_{[-1,1]}/C_{[0,1]}/C_{[1,1]}$ $C_{[0,1]}/C_{[1,1]}/C_{[2,1]}$	相邻词性的组合

表 4 边界后处理

边界检测结果	边界后处理后
双网通 n B O	双网通 n B B
标准版 n E O	标准版 n E E
双网通 n B O	双网通 n B B
高配版 n E O	高配版 n E E

3.2.3 基于 SVM 的属性值类别判定

在属性值类别判定任务中，待识别的属性值需要判别出它们属于预定义属性中的哪一类。根据预处理部分的定义，属性值应当属于专用属性 10 种类型的其中一种。由于属性值类别判定是对已有的属性值进行分类，属于典型的分类问题，基于 SVM 在分类效果的优良表现，所以本文使用 SVM 模型进行分类模型的构建。

3.2.3.1 合并属性值

在分类时属性值被视为一个整体，而不是单个词，因此要对训练集和测试集中的属性值进行合并，在合并的过程中需要对合并之后的属性值重新定义词性，在此，结合属性值合并后的词性标注结果，定义了两种词性，一种是当合并的词词性均为非语素字“x”时，其合并后词性仍为

“x”，除此之外词性全部为名词“n”，属性值合并示例如下表 5 所示。

表 5 合并属性值

原始语料			合并后		
标记	当前词	当前词性	合并后标记	合并词	合并后词性
S-BRA	三星	nt	S-BRA	三星	nt
B-TYP	Galaxy	x	S-TYP	Galaxy J5	x
E-TYP	J5	x	O	配置	v
O	配置	v	S-SCA	5.2 英寸	n
B-SCA	5.2	m	O	显示屏	n
E-SCA	英寸	q			
O	显示屏	n			

3.2.3.2 分类特征的选定

在产品属性值的边界检测与类别判定任务过程中，属性值的内部特征和外部特征都是重要信息，共同指示着属性值的出现及其类别，所以在选择分类特征时，这部分信息依然是十分重要的，此外，除了这部分特征影响外，属性值合并之后的组成信息也可以给类型判断提供一定的依据，因此在类别判定阶段选定的特征有合并后的属性值及其上下文相关的词和词性特征，组成属性值词的个数，组成属性值的第一个词，最后一个词等，其特征模板如下表所 6 示。

表 6 分类特征

模板内容	模板释义
$C_{[-2,0]}$ $C_{[-1,0]}$ $C_{[0,0]}$ $C_{[1,0]}$ $C_{[2,0]}$	$C_{[0,0]}$表示当前词，$C_{[n,0]}$表示当前词的右（n>0）/左（n<0）边第 n 个词
$C_{[-2,0]}/C_{[-2,1]}$ $C_{[-1,0]}/C_{[-1,1]}$ $C_{[0,0]}/C_{[0,1]}$ $C_{[1,0]}/C_{[1,1]}$ $C_{[2,0]}/C_{[2,1]}$	$C_{[0,0]}/C_{[0,1]}$表示当前词与词性的组合，$C_{[n,0]}/C_{[n,1]}$表示当前词的右（n>0）/左（n<0）边第 n 个词与词性的组合
$C_{[-1,0]}/C_{[0,0]}$ $C_{[0,0]}/C_{[1,0]}$ $C_{[0,0]}/C_{[1,0]}/C_{[2,0]}$	相邻词的组合
$C_{[-1,1]}/C_{[0,1]}$ $C_{[0,1]}/C_{[1,1]}$,	相邻词性的组合
Len(C) C_{st} C_{en}	构成属性值词数，属性值第 1 个词，最后 1 个词

3.2.4 电子产品通用属性的属性值识别

对于电子产品通用属性的属性值识别，由于这类属性具有很好的规律，因此采用基于规则的方法。具体的做法就是收集“价格”、“重量”、“颜色”、“产品尺寸”的单位信息，如价格的单位一般是货币（元、美元、日元、欧元等）；重量的单位一般是克(g)、千克(kg)、吨(t)、磅(lb)等；产品尺寸的单位一般是毫米(mm)、厘米(cm)、分米(dm)、米(m)、英尺(ft)、英寸(in)等，对于颜色则是从垂直网络上获取手机的各种颜色组成颜色词表。此外还收集了这几个属性的前缀词表如价格的前缀词一般是售价、定价、仅售等，重量的前缀词一般是重量、重、重达、仅重等，

尺寸的前缀词一般是尺寸等，同时又结合语料资源及词性信息最终制定出了识别通用属性值的规则集。

3.2.5 电子产品属性值识别系统

本文的系统流程图如下图 1 所示。图中训练集 1 与训练集 2 的区别在于训练集 1 中的数据形式为 CRF 要求的格式，训练集 2 中的数据形式为 SVM 要求的格式且训练集 1 中对属性值的标记形式只标注边界信息，训练集 2 中对属性值的标记同时包含边界和类别信息。

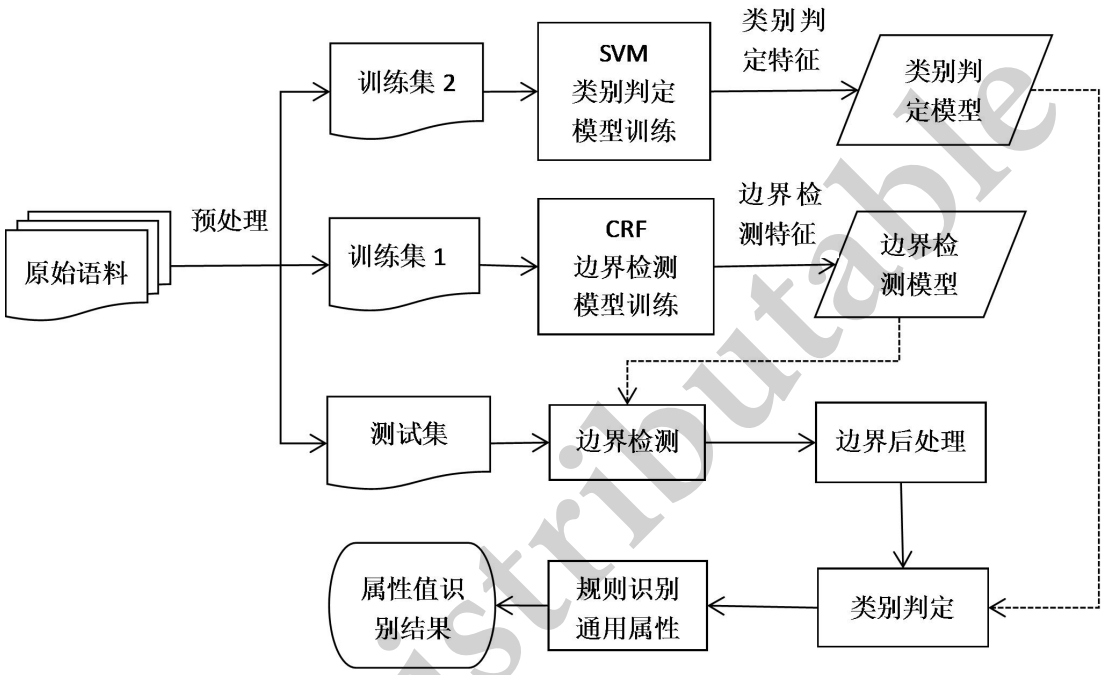


图 1 系统流程图

4 实验

4.1 数据来源

实验采用的数据是从“手机中国”网站上抓取的手机领域相关新闻 1500 篇，经过筛选无关新闻后，选取其中的 1048 篇作为本次实验的语料。在实验语料中随机选取 230 篇作为实验的验证集，剩余的 818 篇中再随机选取 573 篇作为训练集（占剩余语料的 70%），其中包含 10773 个属性值，剩余的 245 篇（占剩余语料的 30%）作为测试集，其中包含 4657 个属性值（通用属性 358 个，专用属性 4299 个），进行实验。

4.2 评价标准

为了综合评价各种方法的性能，本文采用的评价指标主要有 P 准确率、R 召回率以及准确率和召回率的调和平均值 F 值。P 准确率是描述属性值结果准确程度的指标，R 召回率则体现了属性值识别的能力范围，一般情况下，这两者是相互制约的，F 值则综合考虑了准确率和召回率之间的关系，避免了仅仅进行单一的比较 P 准确率和 R 召回率的片面性。三者在属性值识别中的

具体定义如下：

$$P = \frac{\text{正确识别出属性值的个数}}{\text{识别出属性值的个数}} \text{x100\%} \quad (1)$$

$$R = \frac{\text{正确识别出属性值的个数}}{\text{标准结果中属性值的个数}} \text{x100\%} \quad (2)$$

$$F = \frac{2 \times P \times R}{P + R} \times 100\% \quad (3)$$

4.3 实验结果分析

为了比较各种方法的性能和效率，本文做了一系列的实验，所有试验均在同一电脑上完成，电脑的配置为 Intel(R) Core(TM) i5，3.20GHz CPU，Window7 64 位操作系统，4GB 安装内存。实验结果如下表 7 所示，表中 C 表示 CRF 模型，S 表示 SVM 模型，R 表示规则方法，后处理表示边界后处理操作，实验结果的柱状图及耗时柱状图如下图 2，3 所示：

表 7 实验结果

模型	方法	准确率	召回率	F 值	训练耗时/秒
C	一阶段	0.9586	0.8557	0.9042	2121.09
C+R	一阶段	0.9634	0.8707	0.9147	1037.47
S	一阶段	0.9723	0.8883	0.9284	1880.24
S+R	一阶段	**0.9739**	0.8969	0.9338	1667.88
C+S	两阶段	0.9559	0.8791	0.9159	60.96
C+后处理+S	两阶段	0.9533	0.9171	0.9349	61.48
C+S+R	混合模型	0.9593	0.8853	0.9208	**60.12**
C+后处理+S+R	混合模型	0.9568	**09270**	**0.9417**	60.62

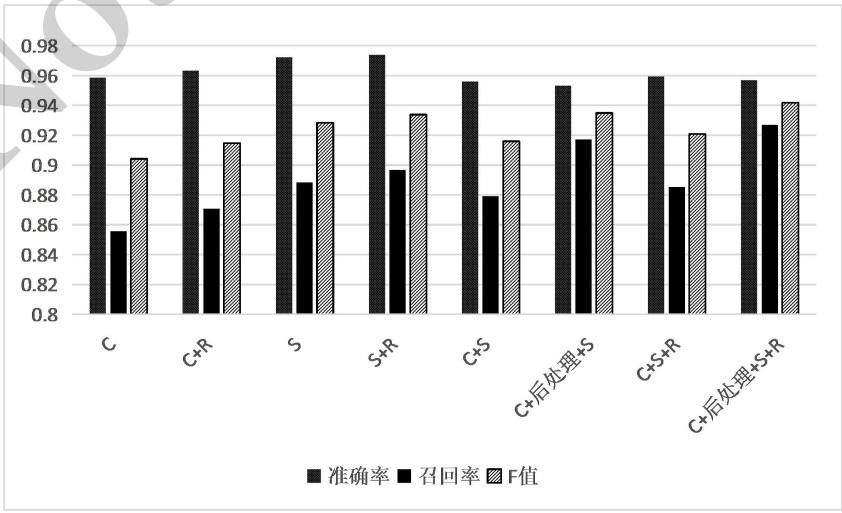


图 2 实验结果图

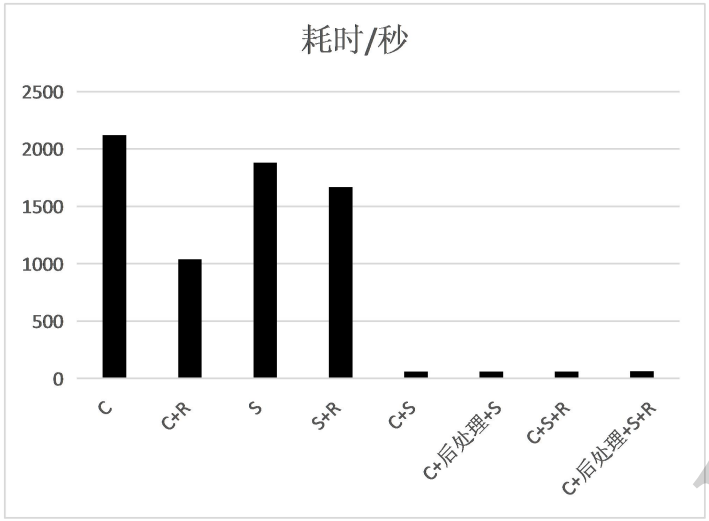


图 3 训练耗时图

规则的有效性：由实验结果可知，无论采用哪一种模型和方法，规则的加入均能使 F 值得到不同程度的提升，这是因为通用属性具有良好的规律，宜于采用规则的方法，而且采用规则的方法还可以有效的减少人工标注的任务量。从训练时间上看，对于一阶段 CRF 的情况，规则的加入可以有效的减少模型训练的时间，这是因为由于 CRF++软件在 CRFs 迭代时使用了 limited memory variable metric (LMVM)，Cohn[16]指出使用 LMVM 的 CRFs 迭代的时间复杂度为：

$$T = O(L^2 NTF) \tag{4}$$

其中 L, N, T, F 分别是标记数量，序列数量、序列平均长度和平均特征数量。当没有加入规则时共有 15 个类别 57 种标记形式（每个类别有以 B、I、E、S 开头的四种标记形式，再加上非属性值标记 O），加入规则后只有 11 个类别 41 种标记形式，同时由于类别的减少也使得平均特征数量 F 减少，综上所述，L 和 F 的减少是效率得到提高的主要原因。对于一阶段 SVM 的情况，是将属性值的识别当做多分类问题来看待，本文采用的是 libsvm 工具包，其所用的是 one- versus-one（一对一）法实现多分类的，所以对于一个 K 分类的问题就需要训练出 K(K-1)/2 个两类分类器。由于规则模块的加入减少了 4 个类别 16 种标记形式，因而减少了训练分类器的数目，所以减少了训练模型的时间。对于两阶段和混合模型的方法，规则的加入对训练的时间影响不大，这是因为混合模型方法与两阶段的方法在边界检测阶段其类别标记数目是一致的，在类别判定阶段，其类别标记数目也变化不大，而且规则的执行是源于字符间的匹配，所以执行只需很少的时间，因此，它们的耗时情况差异不大。

边界后处理的有效性：从实验结果来看，加入边界后处理较不加后处理两种方法 F 值分别提高了 1.9%和 2.09%，其 F 值的提高主要来源于召回率的提升，这是因为采用边界后处理后边界识别的结果得到了进一步的优化，使得识别正确的边界得到了很好的提升。从训练时间上来看，加入边界后处理对时间的影响也十分微弱，这是因为边界后处理也是基于规则执行的，只需很少的时间就可完成。

两阶段方法的有效性：从实验效果上看，两阶段的方法在经过后处理后其 F 值已经优于一阶段的最好方法，这是因为边界后处理工作进一步优化边了边界检测的结果，从而提升了整体的结果。从训练时间来看，两阶段的方法与一阶段的方法相比，效率均有很高的提升，这是因为两阶段的方法在边界检测阶段只需识别出属性值的边界，其类别标记只有 5 种形式（B、I、E、S、O），

类别标记数量较一阶段大大减少；而类别判定阶段是在边界的基础上判别出属性值属于哪一类，经属性值合并后其类别标记数量也只有 14 种，而且在训练模型时清除了训练集中属性值以外的文本，大大减少了训练语料的规模，因此，效率得到了很大的提升。

结论：综合比较各种方法的效果和效率，本文所选取的基于混合模型的电子产品属性值识别方法在效果上优于一阶段和两阶段的方法，在效率上也具有很大的优势，这是由于该混合模型的方法融合了规则、边界后处理以及 CRF 和 SVM 的优点，因此能够在性能和效率上均取得优势。

5 下一步工作

产品属性值识别作为构建产品知识图谱的基础，直接影响着图谱构建的质量。本文提出的基于混合模型的电子产品属性值识别方法，取得了令人满意的效果，但还有值得深入研究的地方，如使用规则进行通用属性值识别的时候，由于属性值所在的上下文环境经常会出现变化，针对这些变化的规则还不够全面，导致有部分属性值无法识别，因此需要进一步研究规则的自动获取方法；在特征选取方面，目前选取的词特征、词性特征等只是对属性值名称或语法成分的一种匹配，只用到了表层的文本信息，因此可以研究更加高效的特征表示方法，如采用词向量的形式，从而提高属性值识别的能力。

6 参考文献

1. 王仁武，王毅，袁旭平.基于深度学习与图数据库构建中文商业知识图谱的探索研究[J].图书与情报，2016(01)：110-117.
2. Riloff E，Phillips W.An introduction to the sundance and autoslog systems[R]. Report，2004.
3. Zhou GuoDong， Su Jian. Named Entity Recognition using an HMM based Chunk Tagger. Proceedings. of the 40th Annual Meeting of the ACL，Philadelphia，July 2002：473-480.
4. Eunji Yi. SVM-based Biological Named Entity Recognition Using Minimum Edit-Distance Feature Boosted by Virtual Examples. IJCNLP，2004：807-814.
5. F.Bechet，A.Nasr，F.Genet. Tagging Unknown Proper Names Using Decision Trees. Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics，Hong Kong，China，2000：77-84.
6. Hal Leong Chieu，Hwee Tou Ng. Named Entity Recognition: A Maximum Entropy Approach Using Global Information[C]. Proceedings of the 19th International Conference on Computational Linguistics(COLING 2002)；Taipei. 2002：46-53.
7. A. Chen，F. Peng，Roy Shan，and Gordon Sun. Chinese named entity recognition with conditional probabilistic models. In 5th SIGHAN Workshop on Chinese Language Processing；Australia，July 2006：173-176
8. Lu P，Yang Y P，Gao Y B et a1．Hierarchical conditional random fields(HCRF)for Chinese named entity tagging．The Third Internat ional Conference on Natural Computation．Haikou，2007：24-28.
9. 闫萍.基于概率统计与规则相结合的命名实体识别研究[J]．计算机与数字工程，2011，39 (9) ：88-91.

10. 毛存礼，余正涛，沈韬等.基于深度神经网络的有色金属领域实体识别[J].计算机研究与发展，2015，52(11)：2451-2459.
11. 邹涛.一种电子产品领域命名实体识别方法研究[D].西安：西安电子科技大学，2010.
12. 何楠.基于统计机器学习的两阶段中文命名实体识别研究[D].北京：北京邮电大学信息工程学院，2008.
13. F Li， Y Du， H Zhao， Z Feng. Two-phase Strategy of Chinese Named Entity Recognition in Micro-blog[J]. Journal of Computational Information Systems. 2014，Vol.10(19)：8421-8428.
14. 孙镇；王惠临.命名实体识别研究进展综述[J].知识组织与知识管理，2010(06)：42-47.
15. Lin Y， Tsai T， Chou W.et al.A Maximum Entropy Approach to Biomedical Named Entity Recognition[J].In proceeding of the 4th Workshop on Data Mining in Bioinformatics (with SIGKDD Conference).2004：56-61.
16. T.Cohn，A.Smith，M.Osborne. Scaling conditional random fields using error-correcting codes[J]. Processings of the 43rd Annual Meeting of the ACL，2005：10-17.

基于概念层次网络的知识表示与本体建模*

文亮，李娟，刘智颖，晋耀红

北京师范大学中文信息处理研究所，北京，100875
wenliang@mail.bnu.edu.cn

摘要：知识表示是自然语言理解的重要基础，知识表示不统一、语义信息无法系统化利用是目前存在的一个亟待解决的问题。要解决这个问题就要解决语义知识表示的问题。本文基于概念层次网络，描述了词语、句子、句群和篇章层面的语义知识表示方法。基于文中描述的词汇层面的表示方法，构建了一个多语言本体知识库。该知识库的知识表示方法不仅可以为知识表示理论提供基础，也可以为自然语言处理相关领域的应用提供资源支持。

关键词：概念层次网络；语义知识表示；本体建模

1 引言

在自然语言处理（NLP）领域，知识表示（Knowledge Representation）的主要目标是把知识数字化、形式化、系统化，便于计算机储存、识别和理解知识。知识表示是自然语言理解的前提和基础，任何语言的理解都要建立在知识表示的基础上。

本体（Ontology、又称为本体论）, 在人工智能领域，本体是指一种“形式化的，对于共享概念体系的明确而又详细的说明”[1]。本体提供的是一种共享词表，也就是特定领域之中那些存在着的对象类型或概念及其属性和相互关系[2]。所以，本体实际上是依据某种类别体系，对实体、概念、事件及其属性和相互关系的形式化表达。

概念层次网络（HNC）理论以概念联想脉络为主线，建立了一种模拟大脑语言感知过程的自然语言理解和处理模式，使计算机获得消解模糊的能力。HNC通过类别符号、层次符号以及组合结构符号的组合，构建了自然语言概念空间的符号化表述体系；同时，HNC以概念基元为基本单位，可以实现概念之间的联想功能。概念基元的联想脉络模拟了人脑的认知机制，可以表述词语、句子、句群和篇章层面的语义知识。

本文基于概念层次网络的知识表示方式，构建了多语言本体词语知识库。具体来说，是以 HNC 概念节点表为纲，对每一个概念进行文字解释，并列出概念所对应的多语言词语，目前为中英双语词语捆绑[3]。

adfa, p. 1, 2011.

本文承国家语委“十二五”科研规划项目“语言资源建设规划研究”(YB125-124)资助。

2 相关工作

目前主要的表示方式可以分为以WordNet[4]、知网（HowNet）[5]等本体知识库为代表的知识表示方式和以Word Embedding为代表的词向量的表示方式。

WordNet 是一个包含了语义信息的机读词典，它能够支持自动的文本分析以及人工智能应用。首先，WordNet 描述了每一个词的基本意义；然后，根据词条的意义，WordNet 将具有相同意义的词条集合为一个Synset（同义词集合）；其次，WordNet 描述了不同Synset之间的语义关系。但是，WordNet只描述了名词、动词、形容词和副词组成的同义词网络，描述的语义信息和关系相对有限，有其不足之处。

知网是一个描述词语（汉语和英语）所代表的概念，揭示概念与概念之间以及概念所具有的属性之间关系的常识知识库。知网定义了事件、万物、属性、属性值、部件、空间和时间七类最顶层的概念。建立了这七类概念之间的关系。知网用“义原”对这些概念进行描述。义原指的是最基本的、不能再分割的表达意义的最小单位。为了描述概念间的关系，知网定义了同义、反义、对义、上下义等语义关系。但知网对概念的定义过于模糊，使用义原解释概念，虽然有利于整合概念之间的关系，但这种描述语言的方式在计算机处理语言时却不能很好的被利用。

词向量的知识表示方式一种是One-hot representation，另一种是Distributed Representation, Tomas Mikolov 等提出的词向量表示工具Word2vec[6]很有代表性，它将词语转化为向量，之后，Tomas Mikolov团队也将其推广到了句子和文档中[7]，将它们转换为一个低维语义空间中的数值向量。其优势在于将自然语言处理过程中的语义鸿沟的现象通过低维空间中向量间数值计算得以一定程度的改善或解决[8]，因此基于深度学习知识表示技术在自然语言处理领域得到了广泛应用。但是，向量表示难以具体描述具体的语义信息，在消解歧义方面还存在着巨大的挑战[9]。

3 基于概念层次网络的知识表示

3.1 概念层次网络基本原理

概念层次网络（Hierarchical Network of Concepts，HNC）是模拟大脑对语言感知的过程建立起的表示概念联想脉络的语义网络[10]。这个理论框架是以语义表达为基础的，它对语义的表达是概念化、层次化、网络化的，所以称它为概念层次网络理论[11]。

HNC理论认为概念无限而概念基元有限、语句无限而句类有限、语境无限而语境单元有限、显记忆无限而隐记忆有限，所以HNC将语言概念空间分为概念基元空间、句类空间、语境单元空间、语境框架空间四个层级。HNC对这四层级的结构体设计了相应的符号体系，建立了语言概念空间体系（包括语义概念基元体系和语句基元体系），通过作用效应链，建立起层次性、网络性的概念表述模式，从而使计算机理解词语、句子、句群及篇章的语义。

3.2 词汇层面的表示模式

词汇层面的表示模式主要通过概念节点来表示，对应于概念基元表示式，即概念基元符号体系，这种表示模式具有语义完备性，能够与自然语言的词语建立起语义映射关系，同时，它高度形式化，每一个符号基元（每个字母或数字）都具有确定的意义，可充当概念联想的激活因子。

HNC把概念分为抽象概念和具体概念。具体概念是指必须确定“所指对象”的概念，基本物概念和挂靠概念属于具体概念，如光和房子；抽象概念是指不必确定“所指对象”的概念，除了基本物概念和挂靠概念的都属于抽象概念。

抽象概念的第一子类即作用效应链，HNC命名为主体基元概念，黄曾阳先生认为“所谓一个事物的知识表示，归根结底就是对作用、过程、转移、效应、关系和状态这6个侧面的表述”[12]，这6个节点是自然语言对万事万物进行描述的六个基本角度，也是一切事物发生、发展和消亡的六个基本环节。在这六个一级节点之下，衍生出许多子节点，共同描述每个概念的各个方面。

抽象概念的第二子类称为扩展基元概念，主要描述人类活动的方方面面，包括生理本能活动、心理活动及精神状态、思维活动、社会活动等一级节点及其衍生的子节点。

具体概念中，基本物概念节点主要包括热、光、声、电磁、微观基本物、宏观基本物和生命体这些一级节点及其衍生子节点，但基本物只是具体物的一小部分，挂靠概念也用来描述具体物。所谓挂靠，就是把一个概念与相关概念的层次符号直接拼接在一起。例如，要用HNC符号表示“交通工具”这个具体物，首先交通工具的主要功能是“转移”，其次交通工具是人造物，所以就将pw（人造物）和 22b（自身转移）这两个概念的层次符号拼接在一起，pw22b就代表交通工具。

HNC使用英语字母、数字和组合结构符作为概念或概念基元的表示符号。描述抽象概念的字母主要有j(表示基本概念)、l(语法逻辑概念)、f(语习逻辑概念)、s(综合逻辑概念)，抽象概念具有五元组特性（字母表示如表1所示）；描述具体概念的字母主要有p(人)、w(物)。数字0～14表示概念的层次性内涵，称为层次符号。组合结构符（#，$，&，|，/等字母）代表符合概念的组合结构。

HNC理论用五元组特性表示抽象概念。现代汉语将词分为动词、名词、形容词、副词等词性。HNC理论用五元组来描述同一概念的不同侧面,分别代表概念的动态（v）、静态（g）、属性（z）、值（u）和效应（r）。

Table 1. 抽象概念的五元组特性

HNC符号	HNC说明	词类命名对应	举例
v	概念的作用（动态）描述	动词	思考 v80
g	概念的作用（静态）描述	名词	思维 g80
z	概念的属性描述	形容词、副词	力度 z00
u	概念的值描述	量词	弱u00c21/强u00c22
r	概念的效应描述	名词	想法 r80

基于HNC的词语表示在计算语义距离时非常方便，如国家表示为pj2，亚洲国家表示为pj2*1，中国表示为pj2*16,从它们的HNC表达式可以看出国家和中国之家是有关联关系的。其中，p表示人，pj表示人化的基本概念，数字表示概念的层次性。

人工生成HNC符号的效率和成本很低，在应用过程中，也产生了HNC符号与词汇的映射工具[13]，这一自动化映射工具大大减轻了词汇与HNC符号的转换成本，为后续的词汇理解、句子理解、句群和篇章理解奠定了基础。

3.3 句子层面的表示模式

句子层面的表示模式指的是用句类表示式描述句子的语义结构特征，HNC用句类(Sentence Category,简称SC)表示式来表征无限的语句。

HNC定义的句类指的是句子的语义类型，而不是指陈述句、疑问句、祈使句和感叹句之分[14]。句类体系主要由广义作用句和广义效应句组成，前者包括作用句、转移句、关系句和一般判断句4个类型，后者包括过程句、效应句、状态句和基础判断句4个类型[15]。这8大类型细分为57种基本句类，57种基本句类理论上可以衍生出3,192组混合句类。以57种基本句类为基元，通过句类的混合和复合就可以实现对自然语言语句的语义结构描述[14]。句类命名和句类符号对应关系如下表：

Table 2. 句类命名和句类符号对应关系

句类命名	作用句	过程句	转移句	效应句	关系句	状态句	判断句
句类符号	X	P	T	Y	R	S	D

句类表示式由语块构成，语块是语句的下一级语义构成单位。HNC定义语块是句类的函数，即句类决定句类表示式中含有哪些语块的表示式。语块存在主块和辅块两种基本类型，语块和主块用同一个字母K表示，辅块用字母fK表示。主块四要素为：特征要素（E）、作用者（A）、对象（B）和内容（C），辅块七要素为：手段（Ms）、工具（In）、途径（Wy）、比照（Re）、条件（Cn）、起因（Pr）、目的（Rt）。

HNC句类一般表示式如下：

SC=JK1+{EK+JKm}(m=2-4)

SCR= SC+fKm

举例如下：

例1： 李四||拒绝了||领导的要求。

X21J = X2A + X2 + XBC

主动反应句 反应者+反应+反应引发者及其表现

例子中X21是句类代码，X表示作用句，等号右边是这个句子的句类表示式。其中，X2A表示反应者，X2表示反应行为，XBC表示反应引发者及其表现。

主动反应句属于广义作用句，还可以有不同的格式代码，例子可以变为李四把领导的要求拒绝了（！11X21J=X2A+XBC+ X2）、领导的要求被李四拒绝了（！12X22 J= XBC+ X2A+ X2）。

通过字母符号及句类衍生，HNC句类表示式可以实现对有限的句类的表示，从而解决无限的语句形式化问题。

3.4 句群、篇章层面的表示模式

在HNC表示体系下，我们把信息抽象成三个侧面：领域、情景、背景，三个侧面构成语境三要素[16]。(在这里，我们把句群、段落、篇章称为信息的载体。)对句群、段落、篇章的表示就是对不同颗粒度大小的语境的描述。通过对表征信息的三个不同侧面（领域、情景、背景）的描述，我们可以形式化表示出语境。

在HNC语境框架理论中，领域描述事件的所属类型，可以看成是对事件范畴的静态描述。情景用来描述事件的作用效应链的具体表现。各参与者以及他们之间的语义关系、事件的内容通常由情景描述指定。背景则用来描述事件发生的条件、叙述者和论述者的背景、叙述者和论述者的特定视野等。情景和事件背景可以理解为是领域的函数。

HNC理论认为任何语段、篇章等构成的语境都是由若干个有限的基本构件组合而成。我们把这些基本构件称之为语境单元。语境单元由领域DOM、情景SIT和背景BAC三要素构成，而背景BAC又区分事件背景BACE和述者背景BACA。语境框架被用来抽象表示语境各要素的构成方式。语境各要素的构成方式可以形式化地表示如下[17]：

SGUN=(DOM;SIT;BACE;BACA)

SGUD=(8y:|DOM;SIT;BACE;BACA)

SIT=SCD(A,B,C)

SGUN——语境单元，分为叙述Narrate型、论述Discuss型； DOM——领域；SIT——情景；BAC——背景；BAC[E//A] ——事件//述者背景；SGUD——语境框架；SCD——领域句类。

语境描述的基础来源于对上下文词语的HNC概念符号的解析。在HNC中，概念基元体系网络中的扩展基元概念专门用来描述人类活动。人类不同的领域活动有不同的符号表示。HNC定义了11大类的领域，每一大类都可以有不同的子类，不同的子类也可以进行组合。语境三要素中的领域信息可以通过解析相关词语的HNC语义符号得到。在确定领域信息后，根据不同领域所蕴含的世界知识，通过进行HNC特有的语义句类分析就可以形成对领域句类SCD的判定。此后，再利用人类专家设计完成的领域句类知识为指导，我们就可以确定语境的情景SIT描述。另外，在领域句类知识的指导下，通过分析辅语义块或某些HNC 语义符号，我们就可以用HNC符号形式化描述出背景BAC。语境的三要素（领域、情景、背景）确定之后，语境的表示也就自然出来了。

4 多语言本体知识库构建

4.1 概念节点与词语的映射

HNC语义网络中任何一个节点都代表一个概念，同时也都是概念的基元。虽然在现实生活中概念是无限的，但作为概念的“元素”的基元是有限的，这些概念基元可以组合成无穷无尽的概念，从而描述自然语言的所有概念。

HNC理论认为大脑自然语言理解基因的直接主体构成大约是15,000左右的概念基元，这有限的15,000个概念基元基本可以描述无限的概念[18]。这项理解基因的探索属于大脑研究的战略性课题，目前知识库针对性地选取了5000个节点进行描述(覆盖全部高层节点，有选择地延伸到底层节点)。

概念分具体概念和抽象概念。对于具体概念，只有一个词类与之对应。但是对于抽象概念，却可以有动态、静态、属性、值和效应等多元性的表现。映射到词语层面，也会表现出动词、名词、形容词等不同的特征。本体知识库中一个概念可映射为不同词类的词语与概念。

4.2 概念节点的关系

在HNC理论中，概念节点的关系最重要的是概念节点之间的层次关系，概念节点是被描述概念节点在概念基元体系中所处的位置，指明其上下位关系。延伸类型的另一类重要的关系，包括对比关系、对偶关系、包含关系。如：“强”与“弱”是对比关系，“正”与“反”是对偶关系，“年”、“月”、“日”之间是包含关系。

概念层次网络的符号表示式的高层主要描述层次性，概念延伸结构则主要体现网络性。层次性表达是概念关联性的根本点之一。延伸也有层次性，但主要体现网络性。延伸结构分两类，第一类延伸结构有三种：对偶性、对比性和包含性；第二类延伸结构也是三种，交织性延伸（t延伸）、并列性延伸（\k延伸）和定向性延伸（i延伸）。交织性延伸最重要。交织性延伸项之间具有内在关联性，是交式关联，相互依存而存在，故命名为交织性延伸。所谓并列延伸比较简单，即属于同一类型、具有平等地位、交织关联性较弱而可不予考虑的。定向延伸是针对某一侧面，具有特定性。通常来说，并列延伸为“类型”；交织延伸为“表现”；定向延伸应是“面向特定内容的描述”。

概念关联式用于描述概念之间的内容逻辑关系。每一个概念都有自身的一组概念关联式。概念区分为延伸概念（EC）、概念树（CT）、概念林(CF)和概念范畴(CC)，它们都有自身的概念关联式。概念关联式的常见类型以特定符号加以表示，如表3所示。

Table 3. 概念关联式的10个特定内容逻辑符号

符号	汉语说明	表示
=	强交式关联于	表示两概念间存在足够大的交集
=>	强源式关联于	表示两概念之间的源流（因果）关系
<=	强流式关联于	表示两概念之间的源流（因果）关系

≡	强关联于	表示两概念间存在着最大的交集和最强的因果关系
:=	对应于	表示两概念具有对应关系
==	虚设	针对延伸概念和概念树
=:	等同于	表示两概念语言意义的等同
%=	属于	表示一概念属于另一概念
=%	包含	表示一概念包含另一概念
::=	定义于	表示一概念等价于另一概念

4.3 多语言本体知识库具体描述信息

知识库中描述的概念节点的信息包括：该节点的中英文命名、概念节点之间的层次关系（上位概念，下位概念和同位概念）、该节点捆绑的词语（动态词语、静态词语、属性词语、值词语、效应词语）。如：节点“7101e06”即概念“反抗”，描述的信息如表1所示：

Table 4. 概念“反抗”的具体描述

属性名	**属性值**
概念节点	7101e06
中文命名	【反抗】
英文命名	revolt
上位概念	7101【对广义作用的心理反应】
下位概念	Null
同位概念	7101e05【攻击】；7101e07【顺从】
动态词语	反抗;对抗;抗争;抗拒;抗击;顽抗;抵抗;抵触;顶撞;冲撞;作对;抗御;抵挡
静态词语	冲突；矛盾
值词语	抗性；承受力；忍耐力
效应词语	Null
属性词语	叛逆；逆反；不屈；不满；不平
挂靠类型	p
具体概念	反贼
关联式	≡,=
关联节点	a13i9e0515【镇压】;a10e269【压迫】;40ibe06【抗拒】；426e21a【反击】;426e229【抗拒】； a13i9e06e16【抗争】;a15e06【抗战】;b03【对命运的抗争】
复合概念	抗暴;起义;造反

通过以上信息的描述，“反抗”这一概念就通过概念层次网络的表示方式被计算机理解。

目前，本体广泛应用的一个瓶颈在于本体构建的自动化程度不高，多数本体还是依赖于手工构建。如何提高本体构建的自动化程度，减少本体构建的成本，提高本体的共享程度，是目前亟待解决的问题。我们所构建的多语言本体知识库是一个动态更新的系统，填写者可以按要求填写概念知识，管理员经过审核后可以确认删除或修改填写的概念节点。我们希望不断有新的填写者加入本体知识库的构建中，采用众包的方式，不断扩展、完善知识库，使之成为能被调用的活知识。

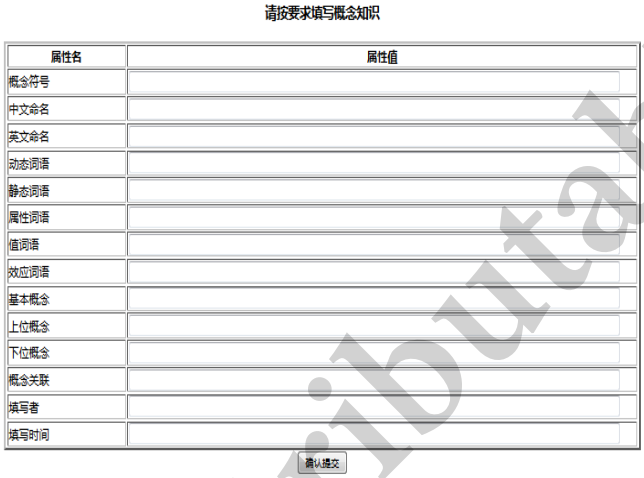


Fig. 1. 多语言本体知识库中概念知识填写细目

填写者可以填写概念符号的属性值，包括中英文命名，此概念捆绑的动态词语、静态词语、属性词语、值词语、效应词语（填写的词语需有中英文对照），基本概念、上下位概念和同位概念。

5 结论

基于概念层次网络的知识表示方法能更好的解决自然语言歧义性这一难题，本文描述了概念层次网络多个层次（词汇、句子、句群、篇章）的语义知识表示方式，根据概念基元体系，我们构建了一个多语言本体词汇知识库，它可以作为自然语言理解系统的基础资源，应用于信息检索、自动问答、机器翻译等领域。

参考文献：

1. Gruber T R. A translation approach to portable ontology specifications[J]. Knowledge acquisition, 1993, 5(2): 199-220.
2. Fensel D. Ontologies[M]//Ontologies. Springer Berlin Heidelberg, 2001: 11-18.

3. Liu Z, Hu R, Jin Y, et al. The Multi-language Knowledge Representation Based on Hierarchical Network of Concepts[C]//Workshop on Chinese Lexical Semantics. Springer International Publishing, 2015: 471-477.
4. Miller G A. WordNet: a lexical database for English[J]. Communications of the ACM, 1995, 38(11): 39-41.
5. Dong Z, Dong Q. HowNet Chinese-English Conceptual Database[R]. Technical Report Online Software Database, Released at ACL, 2000.
6. Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. Proc of ICLR. arXiv:1301.3781, 2013.
7. Le Q V, Mikolov T. Distributed Representations of Sentences and Documents[C]//ICML. 2014, 14: 1188-1196.
8. 刘康,张元哲,纪国良,来斯惟,赵军. 基于表示学习的知识库问答研究进展与展望[J]. 自动化学报,2016,06:807-818.
9. 刘知远,孙茂松,林衍凯,谢若冰. 知识表示学习研究进展[J]. 计算机研究与发展,2016,02:247-261.
10. 黄曾阳. HNC理论全书[M].北京:科学出版社. 2015.
11. 黄曾阳. HNC理论概要[J]. 中文信息学报,1997,04:12-21.
12. 黄曾阳. HNC的发展和未来[J]. HNC 与语言学研究, 2001: 53-68.
13. 熊亮,姚娟. HNC符号与词汇的映射工具的设计[J]. HNC 与语言学研究, 2001: 368-372.
14. 苗传江. HNC(概念层次网络)理论导论[M].北京: 清华大学出版社, 2005.
15. 晋耀红. HNC(概念层次网络)语言理解技术及其应用[M].北京: 科学出版社, 2006.
16. 黄曾阳.语言概念空间的基本定理和数学物理表示式[M] .北京:海洋出版社, 2004.
17. 李伟.基于HNC理论的本体知识表示研究[D].北京师范大学. 2016.
18. 黄曾阳. 语境表示式与记忆[J]. 云南师范大学学报(哲学社会科学版),2010,04:7-14.

基于蔬菜领域中文知识图谱的表示学习方法研究

杜会芳 杜亚茹 陈瑛 赵明

（中国农业大学信息与电气工程学院 北京 100083）

摘要：目前知识图谱主要以网络图的形式进行存储和表示，但基于此方法的查询和搜索都面临较高的计算复杂度和数据稀疏性，如何实现对知识图谱中的实体与关系的有效表示，对知识图谱的研究与应用至关重要。本文以百度百科和互动百科的蔬菜词条为语料构架蔬菜领域知识图谱，采取翻译模型 TransE 对蔬菜三元组进行表示学习，针对蔬菜领域存在的一对多、多对一和多对多的复杂属性关系，提出 PTA 模型，将属性关系和上下位关系进行结合构成关系路径，与蔬菜实体映射到同一个向量空间。实验结果表明，PTA(Path-based TransE for Attribute)模型较 TransE 模型链接预测效果有较大的提高。本研究不仅在其他领域有一定的实用价值，在信息抽取，知识融合和知识推理等方面也提供了参考。

关键词：蔬菜，知识图谱，知识表示学习，TransE，PTA

Knowledge Representation Learning based on Chinese Knowledge Graph in Vegetable Domain

Huifang Du, Yaru Du, Ying Chen, Ming Zhao

College of Information and Electrical Engineering China Agricultural University Beijing, Beiijng 100083

zhaoming@cau.edu.cn

Abstract: Knowledge graph is stored as a graph where each node represents entity and each edge represents relation between entities. Due to the problems of high complexity of the graph algorithms and severe data sparsity, it becomes very important for the researches and applications of knowledge graph that to achieve effective representation the entities and relations on the basis of knowledge graph construction.In this paper, we use vegetable entries of Baidu encyclopedia and HDwiki as data source to study the knowledge representation learning models base on vegetable knowledge graph construction.we adopted TransE model to represent vegetable triples, embedded the entities and relations into a continuous low- dimensional vector space. Thirdly, faced with the complex attribute relations of 1-N, N-1 and N-N, we came up with PTA model, constructed the relation path by combining attribute relations and hyponymy relation, and embedded the relation path into vector space as well. The result showed, without taking into account the relations classification, the link prediction results of PTA model is better than TransE models. And the total value of Hits@10 is higher than TransE model.

Key words: Vegetable, Knowledge Graph, Knowledge Representation Learning, TransE, PTA

1. 前言

知识图谱（Knowledge Graph）是一种新型的海量知识管理与服务模式[1]，其本质为一种语义网络，是描述真实世界中存在的各种实体和实体之间关系的知识库[2]。但是，目前在学术界，研究者针对专业领域

特点，开展知识图谱在生物医学[3]，新闻[4]，影视[5]等领域的应用。在计算机中如何对实体及实体间关系进行表示与存储，从而保证知识的可读性和稳定性，是知识图谱构建与应用的重要课题。近年来，知识的表示方法主要包括基于RDF的图模型表示方法和基于词向量的表示学习技术。

RDF（Resource Description Frame Work）是由W3C提出的对万维网（World Wide Web）上信息进行描述的一个框架[6]，RDF数据是以一个带标记的图来进行较直观的表示，图中的节点和边分别对应三元组中的实体和关系。因此，大规模RDF数据上的查询可以看作是大图上的图匹配问题。然而，由于RDF数据图中包含很多文本信息，节点之间关联多，图的规模巨大，导致RDF数据查询处理复杂、效率低。

基于词向量的表示学习技术以独热（One - hot Representation）表示为开端[7]，逐渐发展到以深度学习为代表的表示学习技术。表示学习是一种分布式表示方法，旨在将研究对象的语义信息表示为稠密实值向量。与图相比，不仅大大降低了向量的维度，还能通过计算实体和关系的语义联系，有效解决数据稀疏问题，使知识图谱构建性能得到显著提升[8]。为了更好地将知识图谱中的实体和关系加以组织和有效利用利用，研究者纷纷致力于知识表示学习提出了多种模型，主要包括距离模型[9]、语义匹配能量模型[10]、隐变量模型、神经网络模型[11]、矩阵分解模型[12]和翻译模型[13]等。

随着 word2vec 的提出，知识的表示学习技术越来越受到重视。word2vec 是一款将词表征为实数值向量的工具包，该工具包主要利用深度学习的思想，通过训练，可以将文本内容转化为向量来完成信息处理，并通过计算向量间的差值来衡量对应文本内容的语义相似度。在此基础上，Bordes 等人提出表示学习的翻译模型 TransE[14]，TransE 模型是一种简单有效的表示学习方案，其主要思想是将知识图谱中的关系看作头实体到尾实体的一种翻译操作。

另外，许多研究工对 TransE 进行扩展，对复杂关系进行建模，具体的改进模型包括 TransH[15]，TransR[16]，TransD[17]等模型以上研究均以翻译模型为基础，做出进一步的研究，但是不同的模型有不同的局限性，还需根据领域和数据集的特点进行针对性的拓展。

但大多数的表示学习技术均针对大规模的英文全局知识图谱，目前基于中文蔬菜领域知识图谱的表示学习技术的研究也鲜见报道。具体说来，蔬菜领域的概念体系结构较复杂，实体的属性关系也复杂和多样，这给表示学习造成了极大的困难，使得蔬菜乃至整个农业领域中，基于知识图谱的表示学习技术研究相对薄弱。

2. 蔬菜领域知识图谱

本阶段，实体的抽取主要借助已有的蔬菜本体，并抽取蔬菜别名关系作为补充。实体间关系主要借助百科网页中的表格信息，并人工从文本中识别有价值的三元组信息。最终构建蔬菜领域知识图谱，为后续的知识表示学习阶段打下基础。

通过复用已有的植物本体，从中提取蔬菜轻量级本体，来建立概念结构体系并以此为标准初步构成蔬菜类概念词典。该本体按照食用器官分类法进行分类，主要涵盖了 14 个大类下的 213 种蔬菜名称，如图 1 所示。

图 1 蔬菜轻量级本体

本文进行知识图谱的表示学习技术采用的数据来自两大主流的百科词条，包括百度百科和互动百科的两大主流百科词条网页进行分批下载。蔬菜词条只涉及诸如番茄、萝卜等传统概念，关系主要包括品种、所属门纲目科等分类关系和包括拉丁学名、英文名、分布地区、栽培技术、病虫害、营养价值、药用价值等多种属性关系和非分类关系。

本部分主要采用半自动和人工的方式，从百度百科词条数据中获取了大约 10650 条三元组信息，共 8780 个实体名称和 187 个关系名称，初步构建了蔬菜领域知识图谱。

3. 表示学习模型介绍

本文主要采取知识表示学习的翻译模型，将知识图谱映射到低维向量空间，对蔬菜领域知识图谱中的三元组进行向量化表示。

3.1 TransE 模型

TransE 属于知识表示学习模型中的翻译模型，其灵感来源于词向量之间的语义平移不变性。该模型是将知识图谱中的关系视为两个不同实体之间的某种平移向量，即利用关系向量 r 作为头实体向量 h 到尾实体向量 t 之间的平移，也可以将向量 r 看作从向量 h 到向量 t 的翻译，所以 TransE 模型也是翻译模型。

由于每一个实体和向量只对应一个低维的向量，TransE 较之先前的模型大大减少了参数的数量，一定程度上大大简化了运算的复杂度。TransE 模型是基于翻译的参数化模型方法，其目的在于构建一个结构化的向量空间。比如，对于（西红柿，病害是，条腐病）三元组来说，有“西红柿”的头实体向量，由关系向量“病害是”，即可利用 TransE 模型翻译出“条腐病”向量。

给定一个三元组 (h,r,t) 的训练集 S，E 为实体集，R 为关系集合，其中 $h,t \in E$ ， $r \in R$ 。对于 S 中的

任何一个三元组(h,r,t)，TransE 希望$h+r\approx t$，其学习目标如图 2 所示：

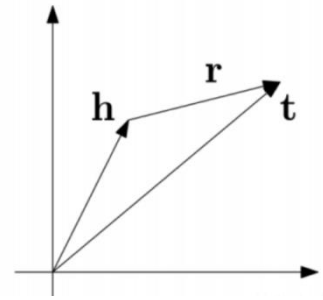


图 2 基于分布式表示的三元组学习目标

3.2 PTA 模型

3.2.1 PTA 模型的提出

上述 TransE 模型主要目的是将知识图谱中的实体和关系映射到一个低维的向量空间。该模型在知识表示学习的过程中发挥了令人满意的效果，但是也有缺陷，主要体现在处理不同类别关系的时候，其学习的效果并不理想。例如，（西红柿，病害是，条腐病）和（西红柿，病害是，细菌性叶斑病），TransE 将这两个三元组映射到同一个低维的向量中，条腐病和细菌性叶斑病趋于重合。条腐病和细菌性叶斑病对于都属于西红柿的病害这一点是相似的，但是，单独来看条腐病和细菌性叶斑病，二者的症状，发病原因和防治办法等都是不一样的。由 TransE 模型的原理可知，它对一个三元组(h,r,t)学习目标是$h+r\approx t$，也就是说，对于同一个头实体和关系来说，不同的尾实体在向量空间中趋于同一点。所以，TransE 忽略了同类别向量的语义差别，在表示 1-1 关系的时候学习效果较好，但是在处理 1-N，N-1，N-N 等复杂的关系这些相似的情况时学习效果较差，从方法本身对于不同类别关系的表示就存在一定的局限性。

本文针对蔬菜领域知识图谱的实体和关系特点，提出 PTA（Path-based TransE for Attribute）模型对 TransE 模型进行改进，主要解决该领域较丰富的复杂属性关系表示学习问题。

从蔬菜领域知识图谱的构建过程中可以看出，除了上下位关系，蔬菜领域包括大量的非分类关系，其中实体间的非分类关系（诸如轮作关系、间作关系等）只占一小部分，绝大部分是属性关系。属性关系具体包括：病害有、虫害有、营养价值、药用价值、生长环境等，而且该领域中，属性关系一般包含两个或两个以上的属性值。

另外，例如，（h 张三，r_1 出生城市，e_1 石家庄），（e1 石家庄，r_2 是省会城市，e_2 河北省），（e_2 河北省，r_3 隶属国家，t 中国）。从以上三个三元组中可以得出，h 实体张三和 t 实体中国之间可以通过 r_1 出生城市，r_2 是省会城市，r_3 隶属国家三个关系共同构成，如此推理可得到一个新的三元组（h 张三，r 出生国家，t 中国）。当然，这属于典型三阶关系推理$h\xrightarrow{r_1}e1\xrightarrow{r_2}e2\xrightarrow{r_3}t$，在蔬菜领域，由于含有的实体数量较少，关系路径没有如此复杂，而且关系主要集中在属性关系，因此，受到路径推理的启发，PTA 模型的目的在于借助路径推理的思想来表示蔬菜领域知识图谱中复杂的属性关系，并将路径也一起映射到低维的实值向量空间中。

3.2.2 PTA 模型介绍

通常情况下，我们采用多个路径关系[88] $P(h,t)=\{p1,...,pn\}$ 来接连头实体 h 和尾实体 t，并且将其看作头实体到尾实体的翻译，其中一个关系路径定义为 $p=\{r_1,...,r_l\}$，该路径表示为 $h\xrightarrow{r_1}...\xrightarrow{r_l}t$。其中，路径由多个关系组成，并以此构成推理模型，发掘知识图谱中两个实体间的隐藏关系。

但是针对蔬菜领域属性关系的复杂性，本文做出两个基本假设：第一，蔬菜领域三元组种所包含的蔬菜实体名称和属性值均归为实体范畴；第二，为属性关系涉及到的上下位关系 ISA 增加方向关系 FISA，例如，三元组（青稞病，ISA，病害名称）和（病害名称，FISA，青稞病）表示的语义关系均为“青稞病是一种病害名称”，知识由于头实体和尾实体互换了位置，导致上下位关系的名称发生了变化。

由于蔬菜领域的特殊性，它与人物关系图谱，歌曲图谱或中医药知识图谱不同，蔬菜领域所包含的实体间关系较少，主要的关系为轮作，间作等关系，除此之外，均可以归为属性关系。属性关系又根据其复杂度进行归类，主要包含以下 4 个类别：

1) 1 - 1 关系。

此类关系主要包含拉丁学名，英文名等，大多为一个属性名称对应一个单一的属性值，路径关系直接表示为一阶路径 $h\xrightarrow{r_1}t$，其中，r1 表示属性关系名称。例如，土豆的拉丁学名为“Solanum tuberosum”，二者对于彼此来说，都是唯一的。

2) 1 - N 关系。

此类关系主要包含别名关系，和类别关系，这些属性关系一般包含 2 个以上的属性值，但这些属性值只可以跟某一种蔬菜对应，其中属性关系路径可表示为二阶路径 $h\xrightarrow{r_2}e\xrightarrow{\text{FISA\_i}}t$，其中 r2 表示属性关系名称。例如，西红柿的别名包括番茄，蕃柿等，这两个别名可只能代表西红柿一种蔬菜。

3) N - 1 关系。

此类关系主要包含界，门，纲，目，科，类别等分类属性和药性，药味，花期果期等这些属性关系只有一个属性值，但是却包含 2 种或两种以上的蔬菜实体，其中属性关系路径可表示为二阶路径 $h\xrightarrow{\text{ISA\_i}}e\xrightarrow{r_3}t$，其中 r3 表示属性关系名称。例如，萝卜和胡萝卜均属于根菜类，根菜类是单一属性值，却对应萝卜和胡萝卜等多个蔬菜名称。

4) N - N 关系。

此类关系主要包含播种方法，栽培技术，病害关系，虫害关系，营养价值，药用价值，食用价值等，其中属性关系路径可表示为三阶路径 $h\xrightarrow{\text{ISA\_i}}e_1\xrightarrow{r_4}e_2\xrightarrow{FISA\_i}t$，其中 r4 表示属性关系名称。这些属性关系的特点是，蔬菜实体和属性值之间存在穿插，比如，牛蒡和山葵的药用价值均包含抗癌和抗衰老的功能。

图 3 的四幅图分别表示了以上四种属性关系的路径规划示意图，实心的圆圈表示蔬菜领域知识图谱中已有的实体，空心的圆圈表示增加的属性模糊实体。该属性模糊实体的增加主要体现在三元组的数据集中，如图 3 的 b 图，三元组（西红柿，别名是，番茄）和（西红柿，别名是，蕃柿）转化为三元组（西红柿，别名是，别名属性），（别名属性，FISA，番茄），（别名属性，FISA，蕃柿），其中，“别名属性”作为属性模糊实体来处理。

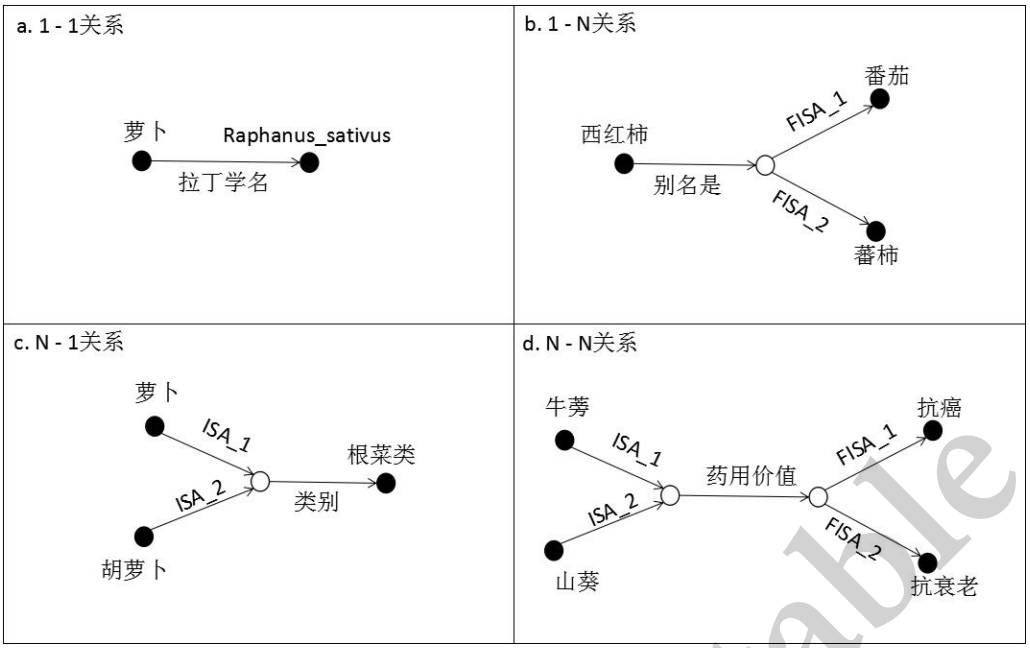


图 3 属性关系分类示例

3.2.3 PTA 关系路径向量化表示

本文需要对三元组(h, p, t)的能量函数$E(h, p, t)$进行明确的定义。与 TransE 模型类似，PTA 模型还需对关系路径进行向量化，与实体、关系和属性值映射到同一个向量空间。该过程依赖于关系路径中的关系向量，需要注意的是，在表示属性的关系路径$h \xrightarrow{r_1} e \xrightarrow{\text{ISA}} t$中，e 为增加的属性模糊实体向量，该实体由多个相关的类似实体求平均值得到。例如，对于三元组$(h, r, t_1) \ldots (h, r, t_n)$:

$$e = \frac{1}{m}(t_1 + \ldots + t_m) \tag{1}$$

其中，m 表示实例数量，关系路径向量$p = (r1, r2)$可以通过相加运算方式得到：

加法运算即将关系路径所包含的所有关系向量进行相加操作，如公式所示：

$$p = r_1 + r_2 \tag{2}$$

其中，r1 和 r2 分别表示两个关系路径，计算过程为矢量相加。

3.3 表示学习目标形式化

对于 TransE 模型，当(h, r, t)存在时，$h + r$和t有较相近的语义关系，当(h, r, t)不存在是，$h + r$和t的语义关系较远。同时，该模型还分别为每一个三元组定义了损失函数，用于计算：$h + r$和t的 L1 或者 L2 距离。

$$d(h + r, t) = | h + r - t |_{L1/L2} \tag{3}$$

其中，$d(h+r,t)$ 表示三元组中 $h+r$ 和 t 之间的语义距离，L1/L2 为两种距离的计算方式，L1 表示元素的绝对值之和，L2 表示元素的平方和。

相对应的，对于 PTA 模型，PTA 对于每一个三元组 (h,r,t) 也定义了相应的能量函数：

$$G(h,r,t)=E(h,r,t)+E(h,p,t) \tag{4}$$

其中，$E(h,r,t)$ 表示三元组实体间存在直接关系的相关性，$E(h,p,t)$ 表示三元组 (h,p,t) 的能量函数，涉及到关系路径 p 的向量化表示。

为进行大规模向量映射，本论文采取基于最大分类间隔的排序准则，为三元组训练集定义了相应的优化目标函数：

$$L=\sum_{(h,p,t)\in S}\sum_{(h',p',t')\in S'_{(h,p,t)}}[\gamma+d(h+p,t)-d(h'+p,t')]_+ \tag{5}$$

其中，+表示取方括号中的正数部分，$\gamma>0$ 为正确三元组和错误三元组之间的最大分类间隔距离。且错误三元组的产生机制并非随机，而是将训练集 S 中的三元组的头实体，尾实体和关系随机且不同时替换成实体集 E 和关系集 R 中的任意一个，以此构成比较有代表性的错误三元组集合 S'：

$$S'_{(h,p,t)}=\{(h',p,t)\,|\,h'\in E\}\cup\{(h,p',t)\,|\,p'\in P\}\cup\{(h,p,t')\,|\,t'\in E\} \tag{6}$$

需要注意的是，在进行实体映射的时候，同一个实体出现在三元组的头实体和尾实体，其所对应的词向量是相同的。

3.4 表示学习模型的优化

本文采用随机梯度下降算法对 TransE 模型和 PTA 模型进行优化处理。梯度下降算法，就是利用负梯度方向来决定每次迭代的新的搜索方向，使得每次迭代能使待优化的目标函数逐步减小。该方法通常也称作最速下降法，常常被用在机器学习和人工智能领域递归性逼近最小偏差模型。

4. 实验及结果分析

4.1 知识表示学习模型训练

4.1.1 表示学习 TransE 模型训练

该阶段主要利用表示学习的 TransE 模型对蔬菜领域知识图谱进行训练，将知识图谱中的实体和关系映射在一个低维稠密的实值向量空间中。本文选取根菜类，薯芋类等十二个类别的蔬菜三元组作为训练数据，食用菌（共 37 种蔬菜）三元组作为测试数据，芽苗菜（共 25 种蔬菜）三元组作为校验数据。具体的数据

条目的统计情况如表 1 所示。

表格 1 知识表示学习 TransE 模型数据集

数据集	实体数目	关系数目	训练数据三元组条数	校验数据三元组条数	测试数据三元组条数	三元组总条数
蔬菜知识图谱	8780	187	7550	1850	1250	10650

该过程涉及到的主要变量及其选值范围为：向量维度 size，取值范围为{50, 100}；向量随机梯度下降算法的学习速率 λ，取值范围为{0.001, 0.01, 0.1}；分类间隔 margin γ，取值范围为{1 , 2, 10}。训练得到的最理想参数配置为：Size = 100，λ = 0.001，γ = 1。

4.1.2 表示学习 PTA 模型训练

对于 PTA 模型，除了对实体进行映射之外，主要对关系路径进行映射，这里的关系路径主要由属性关系和上下位关系构成，一对多和多对一关系为二阶路径翻译模型，多对多关系为三阶路径翻译模型。根据对蔬菜领域关系的研究，包括别名信息，病虫害关系，营养价值，药用价值等属性关系均存在以上的情况，因此，需要为图谱添加属性模糊实体和关系路径，按照关系分类的数据条目的统计情况如表 2 所示，其中训练数据，校验数据和测试数据按照 20：1：1 的概率随机划分。

表格 2 知识表示学习 PTA 模型数据集

分类关系数据集	实体数目	属性关系数目	训练数据三元组	校验数据三元组	测试数据三元组	总条目
1-1 关系	1764	20	1708	50	50	1808
1-N 关系	2921	85	3608	170	170	3948
N-1 关系	2200	39	2984	140	140	3264
N-N 关系	2105	43	2062	136	136	2334
总计	8990	187	10362	496	496	11354

对比表 1 和表 2 可见，实体数目增加了 210 个，三元组总条目增加了 704 条。这些增加的实体数目和三元组条目即为 PTA 模型中增加的属性模糊实体和拓展的相关的三元组。

PTA 模型的训练得到的实体向量，关系向量和关系路径向量。其中，关系路径向量维度也为 100，且训练模型最佳参数配置与 TransE 模型相同。

4.2 评测任务

由于本文涉及到的蔬菜领域知识图谱三元组大多经过人工提取，虽然准确率较高，但是有太多因素会导致图谱的知识覆盖率较低，因此，后续工作亟待解决的一个问题，就是知识图谱的动态更新，对于每一个有所缺失的三元组，均利用知识表示学习模型进行补全。知识图谱的补全选用链接预测（Link Prediction）来作为一个评价指标用来衡量表示学习模型的效果，所谓的连接预测，即给出三元组的其中两个元素，预测出第三个元素，例如，已知蔬菜三元组的头实体为“西红柿”，关系为“别名是”，可根据这两个元素预测出尾实体可能的值为“番茄”，“洋柿子”等。本文主要从实体预测方面分别对 TransE 模型和 PTA 模型在以蔬菜领域知识图谱为数据集的前提下进行对比，从而丰富已有的蔬菜领域知识图谱。

其中，对于每一个确实头实体或确实尾实体的三元组，本文采用随机梯度下降算法计算打分函数，然后对所有的候选实体进行降序排列，并且选取两个评估参数来计算：在预测的实体序列中，测试预料中实体排序的平均值也叫平均秩次（Mean Rank）；正确实体在前十项中的比例（用 Hits@10 表示）。另外，在

对测试三元组进行实体预测评估过程中，我们可以在链接预测排序之前，从知识图谱的训练语料，校验语料和测试语料中过滤出所有此类的三元组，因此，我们将 过滤前的正确三元组数量称为 Raw，过滤后的正确三元组数量称为 Filter。

4.2.1 不考虑关系分类的链接预测

本阶段任务主要是把 TransE 模型应用到蔬菜领域知识图谱中来，验证表示学习模型在小领域中文知识图谱中的学习效果。表 3 所列举的是 TransE 模型和 PTA 模型的三种不同的路径规划算法，在蔬菜领域数据集上的链接预测效果。

另外，表格中 left 指的是在知识表示学习的模型训练过程中进行头实体预测，即按照实体列表中的实体依次替换三元组的头实体从而构成错误三元组集合，并进行语义相似度的计算；与之对应的 right 指的是在知识表示学习的模型训练过程中进行尾实体预测。

表格 3 TransE 模型和 PTA 模型在不区分关系类别前提下的链接预测效果对比

链接预测		Mean Rank		Hits@10（%）	
评测指标		Raw	Filter	Raw	Filter
TransE	left	81.9	81.8	27.7	27.9
模型	right	76.2	75.6	29.1	29.6
PTA	left	74.2	73.1	32.7	35.5
模型	right	75.3	72.0	32.9	33.1

从表 3 可以看出，PTA 模型在路径规划算法预测效果显著优于 TransE 模型，平均秩序 Mean Rank 的 Filter 值下降到 72.6，提前了大约 6 个次序，Hits@10 的 Filter 值达到 34.3%，提高了 5 个百分点。这表明基于路径的属性关系表示学习模型为知识图谱的表示学习提供了一个很好的补充。

4.2.2 考虑关系分类的链接预测

因此，本文主要针对蔬菜领域属性关系的复杂性进行模型的改进。利用属性关系和 ISA 关系的结合构成关系路径来解决 1 - N，N - 1，N - N 的复杂关系的表示学习问题。表 4 所示的是 TransE 模型模型和 PTA 模型在 4 种关系上的 Hits@10 值对比结果。

表格 4 不同表示学习模型在不同类别关系下的链接预测效果

Hits@10(%)	Left（头实体预测）				Right（尾实体预测）				Total
关系类别	1 - 1	1 - N	N - 1	N - N	1 - 1	1 - N	N - 1	N - N	
TransE	30.5	25.2	20.6	25.4	31.9	20.1	25.3	24.0	25.4
PTA(ADD)	32.3	**36.1**	26.5	31.8	33.5	27.0	**38.3**	30.4	32.0

首先，从横向进行比较。分析不同的关系类别和预测方向的关系，表 4 的链接预测结果可被分为四个范畴，其中，由于 1-1 和 N-N 关系具有对称性，三个模型对于头实体预测和尾实体预测效果均无较大差别。但两个模型对于 1-N 和 N-1 的关系的链接预测效果却出现了相反的效果，例如，PTA 模型在 1-N 关系的头实体预测和 N-1 关系的尾实体预测效果分别为 36.1%和 38.3%，二者预测结果相当且在四类属性关系的表示学习中效果最好，而 N-1 关系的头实体预测和 1-N 关系的尾实体预测效果分别为 25.5%和 26.0%，二者预测结果相当但是在四类属性关系的表示学习中效果最差。本文以头实体为例，分析其原因，在 1-N 关系的

头实体预测任务中，关系路径的展开方向与预测方向相反，即对于蔬菜领域三元组来说，多个属性值对应同一个蔬菜实体名称，当头实体缺失时，得相似蔬菜名称的概率由多个属性值确定，因此头实体的预测的准确率较高。而在 N-1 关系的头实体预测任务中，多个蔬菜实体名称对应同一个属性值，当头实体缺失时，预测出某一个蔬菜名称的概率较低，因此预测效果较差。

其次，从纵向进行比较。分析不同模型在同一类别关系下的预测结果。

PTA 模型与 TransE 模型相比，在传统的翻译模型中，增加关系路径的向量化表示，旨在将属性关系与上下位关系进行融合解决复杂关系的表示学习问题。如表 4-7 所示，PTA 模型的最终 Hits@10 值比 TransE 模型提高了 6.6 个百分点，且在关系分类表示中，尾实体预测 N-1 关系结果最优，比 TransE 模型提高了 13 个百分点；

综上所述，PTA 模型对蔬菜领域知识图谱的复杂属性关系有较好的表示效果，在小领域中文知识图谱的表示学习上发挥了较重要的作用。

5. 结论

本文基于百度百科和互动百科的蔬菜词条，研究基于蔬菜领域知识图谱的知识表示学习模型。在快速构建蔬菜领域知识图谱的前提下，利用表示学习的基础翻译模型 TransE 进行中文蔬菜领域知识图谱的表示学习。另外，本研究针对蔬菜领域知识图谱的属性关系特点，将属性关系按照所涉及实体的复杂性进行归类，并提出基于路径算法的 PTA 模型，旨在改善 TransE 模型处理一对多等复杂关系的局限性，有效解决了一对多，多对一和多对多关系的表示学习问题。实验证明，PTA 模型较 TransE 模型预测效果较好，是复杂关系表示学习效果明显增强。

6. 致谢

本论文收到国家自然科学基金(基于弱监督学习的水果品种信息自动抽取方法研究，课题编号：61503386)的资助。感谢匿名审稿人的宝贵意见。

参考文献

[1] 王昊奋. 大规模知识图谱技术. 中国计算机学会通讯, 2014, Vol.10(3), 64-68.

[2] 知识图谱前沿技术研讨,复旦大学，2015.12，http://kw.fudan.edu.cn/workshop2015/

[3] Patrick Ernst, Cynthia Meng, Amy Siu., et al. KnowLife: a Knowledge Graph for Health and Life Sciences. Data Engineering (ICDE), 2014, 1254–1257.

[4] Marco Rospocher, Marieke van Erp, Piek Vossen., et al. Building event-centric knowledge graphs from news. Web Semantics: Science, Services and Agents on the World Wide Web, 2016, Article in Press.

[5] 王巍巍,王志刚,潘亮铭,刘阳,张江涛.双语影视知识图谱的构建研究.北京大学学报(自然科学版),2016(52).

[6] Carroll JJ, Dickinson I, Dollin C, Reynolds D, Seaborne A, Wilkinson K. Jena: Implementing the semantic Web recommendations.In: Feldman SI, ed. Proc. of the 13th Int'l World Wide Web Conf. on Alternate Track Papers & Posters (WWW 2004). New York:ACM Press, 2004. 74 83.

[7] Turian J, Ratinov L, Bengio Y. Word Representations: A Simple and General Method for Semi-Supervised Learning.[C]// Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010:384-394.

[8] 刘知远，孙茂松，林衍凯，谢若冰.知识表示学习研究进展.计算机研究与发展,2016(53).

[9] Bordes A, Weston J, Collobert R, et al. Learning Structured Embeddings of Knowledge Bases[C]. Conference on Artificial Intelligence. 2011.

[10] Bordes A, Glorot X, Weston J, et al. A Semantic Matching Energy Function for Learning with Multi-relational Data[J]. Machine Learning, 2013, 94(2):233-259.

[11] Socher R, Chen D, Manning C D, et al. Reasoning With Neural Tensor Networks for Knowledge Base Completion[C]. In Proceedings of NIPS. Cambridge, MA: MIT Press, 2013: 926-934.

[12] Nickel M, Tresp V, Kriegel H P. Factorizing YAGO: Scalable Machine Learning for Linked Data[C]// WWW. ACM, 2012:271-280.

[13] Fu R, Guo J, Qin B, et al. Learning Semantic Hierarchies via Word Embeddings[C]// Meeting of the Association for Computational Linguistics. 2014:1199-1209.

[14] Bordes A, Usunier N, Garcia-Duran A, et al. Translating Embeddings for Modeling Multi-relational Data[J]. Advances in Neural Information Processing Systems, 2013:2787-2795.

[15] Zhang J. Knowledge Graph Embedding by Translating on Hyperplanes[J]. AAAI - Association for the Advancement of Artificial Intelligence, 2014.

[16] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, Xuan Zhu. Learning Entity and Relation Embeddings for Knowledge Graph Completion. The 29th AAAI Conference on Artificial Intelligence (AAAI 2015).

[17] Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. Knowledge graph embedding via dynamic mapping matrix. In Proceedings of ACL, pages 687–696, 2015.