Knowl Inf Syst
https://doi.org/10.1007/s10115-018-1166-1

REGULAR PAPER

Predicting hypernym–hyponym relations for Chinese taxonomy learning

Chengyu Wang[1] · **Yan Fan**[1] · **Xiaofeng He**[1] · **Aoying Zhou**[2]

Received: 27 December 2016 / Revised: 13 November 2017 / Accepted: 30 January 2018

Abstract Hypernym–hyponym ("*is-a*") relations are key components in taxonomies, object hierarchies and knowledge graphs. Robustly harvesting of such relations requires the analysis of the linguistic characteristics of *is-a* word pairs in the target language. While there is abundant research on *is-a* relation extraction in English, it still remains a challenge to accurately identify such relations from Chinese knowledge sources due to the flexibility of language expression and the significant differences between the two language families. In this paper, we introduce a weakly supervised framework to extract Chinese *is-a* relations from user-generated categories. It employs piecewise linear projection models trained on an existing Chinese taxonomy built from Wikipedia and an iterative learning algorithm to update model parameters incrementally. A pattern-based relation selection method is proposed to prevent "semantic drift" in the learning process using bi-criteria optimization. Experimental results on the publicly available test set illustrate that the proposed approach outperforms state-of-the-art methods.

Keywords Hypernym–hyponym relation extraction · Taxonomy expansion · Weakly supervised learning · Word embedding · User-generated categories

✉ Xiaofeng He
xfhe@sei.ecnu.edu.cn

Chengyu Wang
chywang2013@gmail.com

Yan Fan
eileen940531@gmail.com

Aoying Zhou
ayzhou@sei.ecnu.edu.cn

1 School of Computer Science and Software Engineering, East China Normal University, Shanghai 200062, China

2 School of Data Science and Engineering, East China Normal University, Shanghai 200062, China

Published online: 10 February 2018

1 Introduction

A hypernym–hyponym ("*is-a*") relation is a word/phrase pair (x, y) such that x is a hyponym of y. These relations are extensively employed in machine reading [8], question classification [28], query understanding [12], semantic computation [19] and other NLP tasks. The extraction of *is-a* relations is necessary to construct fine-grained taxonomies for Web-scale knowledge graphs [27,29,33].

In previous work, *is-a* relations were either obtained by using expert-compiled thesauri such as WordNet [23], or automatically harvested from various data sources. Since knowledge in thesauri is usually limited in quantity and variety, it is more prevalent to harvest *is-a* relations from online encyclopedias [25], Web corpora [33], etc. Currently, a majority of existing methods focus on syntactic, lexical and/or semantic analysis on text corpora, but most of these approaches are language dependent. It is not easy to apply methods for one language to knowledge sources in another language directly. For example, in Chinese, the word formation, grammar, semantics and tenses are more flexible and irregular. Thus, pattern-based methods can only cover few linguistic circumstances. As pointed out by Li et al. [17], the performance of syntactic analysis and named entity recognition on Chinese corpora still needs to be improved to support robust relation extraction. Furthermore, it is still difficult to use machine translation-based methods to extract such relations because there are great differences in word orders and grammar between English and Chinese [1].

More recently, word embedding (or distributed word representation) has been empirically proved effective in modeling some of the semantic relations between words by the offsets of word vectors [21,22]. The learning of word embeddings only requires shallow processing of a large text corpus. As Fu et al. [9] suggest, the representation of *is-a* relations is more complicated than vector offsets. By studying the relations of word embeddings between hyponyms and their respective hypernyms, *is-a* relations can be identified by learning semantic prediction models.

In this paper, we take an existing Wikipedia-based Chinese taxonomy that we previously built [18] as the initial knowledge source, and consider the problem of harvesting Chinese *is-a* relations from user-generated categories. User-generated categories are employed as the knowledge source because they are fine-grained classes, concepts or topics manually added by human contributors in online encyclopedias, vertical websites, etc. They provide high-quality candidate hypernyms for entities. For instance, in *Baidu Baike*,[1] the page 奥巴马 (Barack Obama) has the following categories: 政治人物 (Political Figure), 外国 (Foreign Country), 元首 (Leader) and 人物 (Person). Given an entity and its category set, we aim to predict whether each category name is the hypernym of the entity. In this way, new *is-a* knowledge can be continuously added to the taxonomy. However, using a single model is difficult to preserve all the linguistic regularities of *is-a* relations extracted from varied data sources and domains. Furthermore, models learned from one knowledge source are not necessarily effective to extract *is-a* relations from another source, while it is a common practice to construct large-scale taxonomies from multiple Web sources [6,10,31].

To address this problem, we propose a weakly supervised framework to extract *is-a* relations automatically. In the initial stage, we build piecewise linear projection models trained on samples from the initial Chinese taxonomy [18]. In this stage, a K-means-based incremental clustering technique is employed to group *is-a* relations with similar semantics together. In each cluster, a separate model maps entities to their respective hypernyms in the embed-

[1] Baidu Baike (http://baike.baidu.com/) is one of the largest online encyclopedia websites in China. The example is taken from the online version Baidu Baike in June, 2016.

ding space. After that, clustering results are updated incrementally with projection models retrained in an iterative manner. In each iteration, we extract previously unseen *is-a* relations from a collection of unlabeled <entity, category> pairs. To avoid "semantic drift" [3], a bi-criteria optimization method is proposed such that only those extracted *is-a* relations that are validated by three types of Chinese patterns in a corpus can be labeled as "positive" and added to the training set. In this way, projection models for the target knowledge source are trained without human labeling efforts.

In summary, we make the following major contributions in this paper:

- We propose a weakly supervised method to extract *is-a* relations from user-generated categories.
- We introduce an incremental learning method to learn model parameters to map embedding vectors of hyponyms to their respective hypernyms.
- Extensive experiments are conducted on a public Chinese *is-a* test set to illustrate the effectiveness of the proposed approach, with an application presented for the demonstration purpose.

The rest of this paper is organized as follows. Section 2 summarizes the related work. In Sect. 3, we overview our prior work on Chinese taxonomy construction based on Wikipedia. Details of our approach for addressing the *is-a* relation extraction problem are described in Sect. 4. The process of experimental data construction is introduced in Sect. 5. Experimental results are presented in Sect. 6. We conclude our paper and discuss the future work in Sect. 7.

2 Related work

The *is-a* relation extraction or taxonomy learning problem has been addressed by identifying hyponyms and their hypernyms from various data sources. Here, we present a summarization of methods on *is-a* relation extraction and discuss how they can be employed for Chinese *is-a* relation extraction.

Some knowledge graphs have handcraft, fixed taxonomies with fine quality, such as NELL and DBpedia. In these taxonomies, *is-a* knowledge is manually obtained by human experts. In NELL, categories are manually arranged into a hierarchical structure so that entities are extracted from texts and mapped to certain categories by coupled training [2,3]. In DBpedia, entities are mapped to a cross-lingual, universal taxonomy by contributors of the project [15]. The major drawback of manually constructed taxonomies is relatively low coverage, especially in newly emerged areas and specific domains.

Pattern matching-based methods employ syntactic/lexical patterns to extract *is-a* relations. The early work introduced by Hearst [11] utilizes manually designed patterns to obtain *is-a* relations from text corpora. For instance, based on the sentence pattern $\texttt{NP}_1$ `such as` $\texttt{NP}_2$, it can be inferred that $\texttt{NP}_2$ is a hypernym of $\texttt{NP}_1$, where $\texttt{NP}_1$ and $\texttt{NP}_2$ are noun phases. These patterns are effective for English and are used to build the largest taxonomy Probase from a large-scale Web text corpus [33]. However, it is hard to handcraft all valid *is-a* patterns. As Fu et al. [9] suggest, many *is-a* relations are expressed in highly flexible manners in Chinese and these approaches have limited extraction accuracy.

Dictionaries, thesauri and encyclopedias can serve as knowledge sources to construct object hierarchies. Suchanek et al. [27] link concepts in Wikipedia to WordNet synsets [23] by considering the textual patterns of Wikipedia categories. Melo and Weikum [4] utilize the category systems from Wikipedia editions of different languages to integrate multilingual taxonomic data. Lin et al. [20] present an entity detection and typing approach for entities that

are not present in Wikipedia using entities present in Wikipedia and the Freebase semantic types. A drawback of these methods is that Chinese is a relatively low-resourced language and it is difficult to apply these methods directly. For example, there is no Chinese version of Freebase available. For Chinese, our prior work [18] introduces a set of language-specific features to predict *is-a* relations using an SVM classifier and construct a large-scale Chinese taxonomy from Wikipedia. For the Chinese language, a semantic lexicon named HowNet [7] contains over 800 "sememes" (i.e., a basic semantic unit, similar to synsets in WordNet), extracted from 6000 Chinese characters. Fu et al. [10] utilize multiple data sources such as encyclopedias and search engine results to design a ranking function in order to extract the most possible hypernym given an entity. These methods are more precise than free-text extraction approaches, but have limited scope constrained by sources.

Text inference approaches make use of distributional similarity measures, which go beyond pattern matching methods but instead compare the contexts of word pairs in a corpus to infer their relations indirectly. Kotlerman et al. [14] consider the asymmetric property of *is-a* relations and design directional similarity measures to make lexical inference. Wong et al. [32] distinguish non-taxonomic concept pairs from taxonomic pairs based on existing domain ontology and unstructured text. One potential limitation of text inference approaches for Chinese is that the contexts in Chinese are usually flexible and sparse. As a result, these measures are not very effective to distinguish *is-a* or *not-is-a* relations.

To tackle the data sparsity problem, word embedding-based approaches have been proposed, which benefit NLP tasks, such as sentiment classification [37,38], machine translation [36], question answering [34], query expansion [5], etc. In these approaches, words are mapped to a low-dimensional space by training neural network-based language models, such as *CBOW*, *Skip-gram* [21], *GloVe* [24]. The dense word representations are more likely to deal with the context sparsity issue in Chinese stemmed from the flexible expressions. The state-of-the-art method in [9] for the Chinese language is most related to ours, which takes a Chinese thesaurus as a-priori knowledge and train piecewise linear projection models based on word embeddings. Additionally, Yu et al. [35] design a distance-margin neural network model to learn term embeddings as features to identify *is-a* pairs. In this paper, we further improve the performance of the word embedding-based method by iterative learning of projection models and *is-a* relation selection based on Chinese hypernym/hyponym patterns.

3 Prior work: initial Chinese taxonomy construction

The initial Chinese taxonomy is constructed based on our previous work [18]. In that work, we take Chinese Wikipedia as the knowledge source, develop mining methods to extract entities, classes and *is-a* relations from Wikipedia, and build up the entire taxonomy using a bottom-up strategy. Detailed statistics and analysis of our taxonomy will be introduced in Sect. 5.

Wikipedia has a relatively large and well-organized category system in a tree-like structure, which enables us to construct a taxonomy. Formally, a taxonomy is defined as follows:

Definition 1 *(Taxonomy)* A taxonomy $T = (V, R)$ is a rooted, labeled tree where nodes V are entities or classes and edges R represent *is-a* relations. Specifically, for each non-root $x \in V$, there exists a class $y \in V$ where x is a hyponym of y.

Following the work in Fu et al. [9], *is-a* relations are regarded as *asymmetric* and *transitive* relations. However, it is a non-trivial task to identify these *is-a* relations from Wikipedia because most categories express the semantic relatedness to the entity, or the topics or fields

the entity belongs to, instead of *is-a* relations (see also [25,27]). In our work, we extract *is-a* relations from categories by a classifier and build up the entire taxonomy using inference based rules. We briefly summarize these approaches in the next subsections.

3.1 Low-level *is-a* relation classification

To distinguish *is-a* relations from others, we design a classifier-based scoring function. Given an entity x, a category in Wikipedia y, and two sets of features $\mathbf{F_1}$ and $\mathbf{F_2}$, the function outputs a positive number for *is-a* relation; negative otherwise, defined as follows:

$$f(x, y) = \mathbf{w_1}^{\mathrm{T}} \cdot \mathbf{F_1}(x, y) + \mathbf{w_2}^{\mathrm{T}} \cdot \mathbf{F_2}(y) + w_0$$

where $\mathbf{F_1}(x, y)$ considers both information of x and y (called entity-dependent features) while $\mathbf{F_2}(y)$ (called entity-independent features) only takes the properties of y into account. $\mathbf{w_1}$ and $\mathbf{w_2}$ are the respective weight vectors and w_0 is the bias term.

The two feature sets are introduced as follows. Features 1–4 are entity-independent features, while features 5–7 are entity-dependent features.

Feature 1: (Category Length) If the length of a category (i.e., number of words in a category name after word segmentation) is too long or short, it may be too general or too specific to describe the class of an entity.

Feature 2: (POS Tag) A valid class name is usually a noun or a noun phrase. We use the POS tag of the head word of the category as a feature.

Feature 3: (Thematic Category) As is described in [25], some categories, such as finance, politics, entertainment, are thematic categories rather than conceptual classes. We have collected a set of themes in Chinese and take whether a category or the head word of a category is a thematic word as a feature.

Feature 4: (Language Pattern) In English, a conceptual category is often in the form of `premodifier + head word + postmodifier` (see [27]). We have observed that in Chinese, the corresponding pattern is `premodifier +`的`+ head word` where 的 is an auxiliary character in Chinese. We take whether the category name fits this pattern as a feature.

Feature 5: (Common Sequence) In Chinese, an entity and a category may have a common subsequence. For example, the category 政党 `(political party)` is a correct class for the entity 工党 `(Labor Party)`. We take the existence of the common sequence as a feature.

Feature 6: (Head Word) Similar to Feature 5, if the longest common sequence (LCS) of an entity and a category is the head word of the category, the category is likely to be a valid class.

Feature 7: (Purity) Let E_y be an entity set such that for each entity $e \in E_y$, there exists a category named y in that entity page. Intuitively, if most entities E_y are person names, the category y is likely to be a valid class related to people. We employ named entity recognition (NER) to tag entities. The purity of a category y is defined as:

$$\mathrm{purity}(y) = \max_{l \in L} \frac{|E_l \cap E_y|}{|E_y|}$$

where L is a collection of NE tags and E_l is the collections of entities that are labeled as l. We define whether purity(y) is larger than a threshold as a feature.

3.2 High-level *is-a* relation inference

The *is-a* relation classification approach is not sufficient for building the entire taxonomy. This is because most hypernyms harvested by the previous method are relatively low level and specific (e.g., `20th-century Chinese Businessman`), lacking high-level classes (e.g., `Person`).

To solve this issue, we further design two methods to infer high-level *is-a* relations. Implementation details can be found in [18]. The first method is the inference based on relational categories in Wikipedia. Take the Wikipedia page w.r.t. `Albert Einstein` as an example. We may extract a relation `(Albert Einstein, graduate-from, ETH Zurich)` from the category named `ETH Zurich Alumni`. This indicates that the two *is-a* relations holds: `(Albert Einstein,` *`is-a`*`, Person)` and `(ETH Zurich,` *`is-a`*`, Educational Institution)`. Similarity, the *is-a* relation `(Albert Einstein,` *`is-a`*`, Person)` can be extracted from the property `(Albert Einstein, die-in, 1955)`, where this property is learned from the category called `1955 Deaths`. Therefore, we design two inference rules for the subject class and the objective class given a type of relations, and only one inference rule for the subject class given a type of property. The relation and property extraction method is introduced in [27]. In our taxonomy construction system, we implement 70 regular expressions for category pattern matching and only use inference rules with high accuracy. Some examples of inference rules and their respective performance are shown in Table 1.

The second method determines whether there is an *is-a* relation between two classes under the framework of associated rule mining. Briefly, define $\Phi(y)$ as the collection of hyponyms of y extracted previously. For two classes y_i and y_j, the confidence of the relation (y_i, *`is-a`*, y_j) is calculated as:

$$\text{conf}(y_i, y_j) = \frac{|\Phi(y_i) \cap \Phi(y_j)|}{|\Phi(y_i)|}$$

We can see $\text{conf}(y_i, y_j)$ determines whether y_i is a subclass (i.e., hyponym) of y_j. When $\text{conf}(y_i, y_j)$ is no less than a threshold, the relation (y_i, *`is-a`*, y_j) holds. For example, we can infer the relation `(Chinese Pop Music Composer,` *`is-a`*`, Person)` based on the respective numbers of entities with classes `Chinese Pop Music Composer` and `Person`.

Table 1 Examples of inference rules for *is-a* relation inference

Sub. class	Obj. class	Regular expression	#Relations	Accuracy (%)
`City`	`Region`	(.*省)市镇 (.* Province) City	32,091	100
`Politician`	`Position`	(.*(委员|参议员|参政员|议员)) (.*(Committee Member| Congressman))	13,881	100
`Person`	–	(.*? \ d{1, 4}年)逝世 (.*? \ d{1, 4} Year) Deaths	10,148	99
`Monarch`	–	(.*?)(君主|国王) (.*?)(Monarch|King)	3,649	100

Finally, the taxonomy is constructed in a bottom-up manner via incorporating three operations. In summary, the process of taxonomy construction can be divided into three phases, namely node merging, cycle removal and sub-tree merging. Node merging combines several *is-a* relations with the same hypernyms/hyponyms into a tree structure. Cycle removal removes any cycles produced in the node merging process. Sub-tree merging connects all generated sub-trees to form a complete taxonomy. In the taxonomy, we treat all the leaf nodes as entities and others as classes, and label *is-a* relations between classes and entities as *instance-of* relations, others as *subclass-of* relations.

4 Weakly supervised *is-a* relation extraction

In this section, we describe the formal definition of our problem. The motivation of our method is discussed, and the detailed steps are introduced, namely initial model training and iterative learning process. Important notations are summarized in Table 2.

4.1 Our task

The input knowledge source is a collection of known *is-a* relations, sampled from the taxonomy. Given the *is-a* relations R, based on the transitivity property, all correct *is-a* relations are in the transitive closure of R, defined as:

$$R^* = \bigcup_{i=0}^{\infty} R^{(i)}$$
$$R^{(i+1)} = R \circ R^{(i)}$$

with initial condition $R^{(0)} = R$ and $\circ$ being the composition operator of relations.

To extract *is-a* relations from user-generated categories, we obtain the collection of entities E from the knowledge source (such as *Baidu Baike*). The set of user-generated categories for

Table 2 Important notations

Notation	Description
R	Positive *is-a* relations
R^*	Transitive closure of R
U	Unlabeled word pairs
$T = (V, R)$	An existing taxonomy
$\mathbf{v}(x)$	Embedding vector of word x
$\mathbf{M}_k$	Projection matrix of the kth cluster
$\mathbf{b}_k$	Vector offset of the kth cluster
$\mathbf{c}_k$	Centroid vector of the kth cluster
C_k	Collection of word pairs in the kth cluster
$f_M^{(t)}(x_i, y_i)$	Model-based prediction for word pair (x_i, y_i) in tth iteration
$f_P^{(t)}(x_i, y_i)$	Pattern-based prediction for word pair (x_i, y_i) in tth iteration
$\mathrm{PS}^{(t)}(x_i, y_i)$	Positive score for (x_i, y_i) in tth iteration
$\mathrm{NS}^{(t)}(x_i, y_i)$	Negative score for (x_i, y_i) in tth iteration

Table 3 Examples of three observations. We use l_2 norm of vector offsets to quantify the differences

	Example with English Translation	Difference
True positive	$\mathbf{v}$(日本) − $\mathbf{v}$(国家) ≈ $\mathbf{v}$(澳大利亚) − $\mathbf{v}$(国家)	1.03 ≈ 0.99
	$\mathbf{v}$(Japan) − $\mathbf{v}$(Country) ≈ $\mathbf{v}$(Australia) − $\mathbf{v}$(Country)	
Obs. 1	$\mathbf{v}$(日本) − $\mathbf{v}$(国家) $\not\approx$ $\mathbf{v}$(日本) − $\mathbf{v}$(亚洲国家)	1.03 $\not\approx$ 0.71
	$\mathbf{v}$(Japan) − $\mathbf{v}$(Country) $\not\approx$ $\mathbf{v}$(Japan) − $\mathbf{v}$(Asian Country)	
Obs. 2	$\mathbf{v}$(日本) − $\mathbf{v}$(国家) $\not\approx$ $\mathbf{v}$(主权国) − $\mathbf{v}$(国家)	1.03 $\not\approx$ 1.32
	$\mathbf{v}$(Japan) − $\mathbf{v}$(Country) $\not\approx$ $\mathbf{v}$(Sovereign State) − $\mathbf{v}$(Country)	
Obs. 3	$\mathbf{v}$(日本) − $\mathbf{v}$(国家) $\not\approx$ $\mathbf{v}$(西瓜) − $\mathbf{v}$(水果)	1.03 $\not\approx$ 0.39
	$\mathbf{v}$(Japan) − $\mathbf{v}$(Country) $\not\approx$ $\mathbf{v}$(Watermelon) − $\mathbf{v}$(Fruit)	

each $x \in E$ is denoted as $Cat(x)$. Thus, we need to design a learning algorithm F based on R^* to predict whether there is an *is-a* relation between x and y where $x \in E$ and $y \in Cat(x)$. In this way, we harvest new *is-a* knowledge automatically to expand the Chinese taxonomy without any human intervention. Define $U = \{(x, y)|x \in E, y \in Cat(x)\}$ as the unlabeled word pairs (i.e., candidate *is-a* relations). We define the task of taxonomy learning as follows:

Definition 2 *(Taxonomy Learning)* Given a collection of *is-a* relations R^* derived from a taxonomy T and a collection of unlabeled word pairs U, the task is to extract *is-a* relations from U based on T.

It is worth noting that our task definition can be fitted in any language settings. Due to the research challenges in learning relations in Chinese, we only focus on Chinese *is-a* relation extraction in this paper.

4.2 Observations and general framework

To our knowledge, the state-of-the-art method for Chinese *is-a* relation extraction is the word embedding-based approach in [9]. In their work, the projection parameters of a piecewise linear projection model learned from a Chinese thesaurus are used to identify *is-a* relations in encyclopedias. In this paper, we take a deeper look at the word vectors of hyponyms and hypernyms. As a preliminary experiment, we randomly sample *is-a* relations from the initial Chinese taxonomy and a Chinese thesaurus *CilinE*.[2] Denote $\mathbf{v}(x)$ as the embedding vector of word x. We compute the offsets of embedding vectors (i.e., $\mathbf{v}(x) - \mathbf{v}(y)$) where x is the hyponym of y. We have three observations, with examples shown in Table 3.

- *Observation 1* For a fixed x, if y_1 and y_2 are hypernyms of different abstraction levels, it is likely that $\mathbf{v}(x) - \mathbf{v}(y_1) \not\approx \mathbf{v}(x) - \mathbf{v}(y_2)$. For example, `Country` is a high-level hypernym of `Japan` while `Asian Country` covers a narrow spectrum of entities.
- *Observation 2* If $(x_1,$ `instance-of`, $y_1)$ and $(x_2,$ `subclass-of`, $y_2)$ hold, it is probable that $\mathbf{v}(x_1) - \mathbf{v}(y_1) \not\approx \mathbf{v}(x_2) - \mathbf{v}(y_2)$. Although both *instance-of* and *subclass-of* are *is-a* relations in a broad sense, the *is-a* relations between (i) entities and classes, and (ii) classes and classes are different in semantics.
- *Observation 3* For *is-a* pairs in two different domains (x_1, y_1) and (x_2, y_2), it is likely that $\mathbf{v}(x_1) - \mathbf{v}(y_1) \not\approx \mathbf{v}(x_2) - \mathbf{v}(y_2)$. This implies that *is-a* relations can be divided into more

[2] http://www.ltp-cloud.com/download/

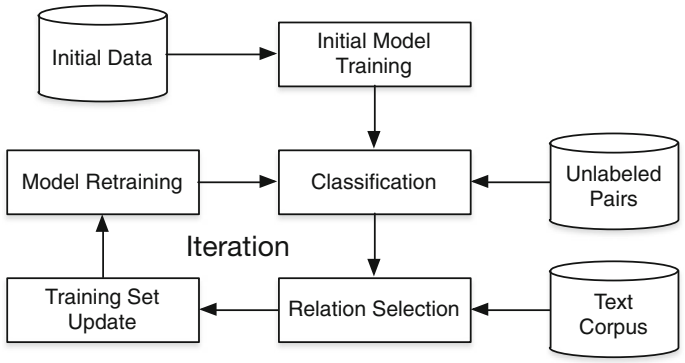


Fig. 1 General framework of the proposed approach

fine-grained relations based on their topics, such as politics, grocery. A similar finding is also presented in [9].

These situations bring the challenges in modeling *is-a* relations correctly. Furthermore, *is-a* relations across different knowledge sources vary in characteristics. For example, *is-a* relations in a Chinese thesaurus such as *CilinE* are mostly *subclass-of* relations between concepts, while a large number of *is-a* relations derived from online encyclopedias are *instance-of* relations, especially in the emerging domains, such as the Internet, new technologies. The differences of *is-a* representations between knowledge sources suggest that a simple model trained on the taxonomy is not effective for *is-a* extraction from encyclopedias. The observations prompt us to take a two-stage process to deal with this problem. In the initial stage, we train piecewise linear projection models based on the taxonomy, aiming to learn prior representations of *is-a* relations in the embedding space. Next, we iteratively extract new *is-a* relations from user-generated categories using models in the previous round and Chinese hypernym/hyponym patterns to adjust our models accordingly. The characteristics of *is-a* relations of the target source are learned in a step-by-step manner. The general framework of this approach is illustrated in Fig. 1.

4.3 Initial model training

To learn word embeddings, Mikolov et al. [21] previously proposed two models (i.e., *CBOW* and *Skip-gram*) that can efficiently capture the semantics of words with low runtime complexity. As their experiments on large corpora show, the *Skip-gram* model has higher performance. Thus, we first train a *Skip-gram* model on a Chinese text corpus with over 1 billion words to obtain word embeddings.

In the *Skip-gram* model, each word x is projected to its low-dimensional embedding vector $\mathbf{v}(x)$. After that, a log-linear classifier takes the embedding vector of the word as input and predicts the context words. Formally, the conditional probability of the context word u given the current word x is defined as:

$$\Pr(u|x) = \frac{\exp(\mathbf{v}(u)^{\mathrm{T}} \cdot \mathbf{v}(x))}{\sum_{u' \in \mathbb{V}} \exp(\mathbf{v}(u')^{\mathrm{T}} \cdot \mathbf{v}(x))}$$

where $\mathbb{V}$ is the vocabulary collection over the entire text corpus.

In previous works, Mikolov et al. [22] and Fu et al. [9] employ vector offsets and projection matrices to map words to their hypernyms, respectively. In this paper, we further combine the

two relation representations together in the embedding space. For a pair (x_i, y_i), we assume a projection matrix $\mathbf{M}$ and an offset vector $\mathbf{b}$ map x_i to y_i in the form:

$$\mathbf{M} \cdot \mathbf{v}(x_i) + \mathbf{b} = \mathbf{v}(y_i)$$

To capture the multiple implicit language regularities in the training data, we follow the piecewise model training technique in [9]. We first partition the *is-a* relations R^* into K groups by K-means, denoted as $R^* = \bigcup_{k=1}^{K} C_k$ where C_k is the collection of *is-a* pairs in the kth cluster and K is the number of clusters. Each pair $(x_i, y_i) \in R^*$ is represented as the vector offset $\mathbf{v}(x_i) - \mathbf{v}(y_i)$ for clustering. In each cluster, we assume *is-a* relations share the same projection matrix and vector offset. Therefore, we aim to learn K projection matrices and offset vectors as representations of *is-a* relations. For each cluster C_k ($k = 1, 2, \cdots, K$), we aim to minimize the following objective function:

$$J(\mathbf{M}_k, \mathbf{b}_k; C_k) = \frac{1}{|C_k|} \sum_{(x_i, y_i) \in C_k} \|\mathbf{M}_k \cdot \mathbf{v}(x_i) + \mathbf{b}_k - \mathbf{v}(y_i)\|^2$$

where $\mathbf{M}_k$ and $\mathbf{b}_k$ are the projection matrix and the offset vector for C_k, learned via Stochastic Gradient Descent (SGD).

4.4 Iterative learning process

In the iterative learning process, we train *is-a* relation projection models on a series of dynamically enlarged training set $R^{(t)}$ ($t = 1, 2, \cdots, T$). The main idea is to update clustering results and prediction models iteratively in order to achieve a better generalization ability on the target knowledge source.

Initialization We have two datasets: (i) the positive *is-a* relation collection $R^{(1)} = R^*$ and (ii) the unlabeled word pair collection $U = \{(x_i, y_i)\}$, which is created from user-generated categories. Usually, we have $|U| \gg |R^{(1)}|$. Denote $C_k^{(t)}$ as the collection of *is-a* pairs, $\mathbf{c}_k^{(t)}$ as the cluster centroid, and $\mathbf{M}_k^{(t)}$ and $\mathbf{b}_k^{(t)}$ as model parameters in the kth cluster of the tth iteration. We set $C_k^{(1)} = C_k$, $\mathbf{c}_k^{(1)} = \frac{1}{|C_k|} \sum_{(x_i, y_i) \in C_k} \mathbf{v}(x_i) - \mathbf{v}(y_i)$, $\mathbf{M}_k^{(1)} = \mathbf{M}_k$ and $\mathbf{b}_k^{(1)} = \mathbf{b}_k$ as the initial values.

Iterative Process For each iteration $t = 1, \ldots, T$, the models and the datasets are updated as follows:

– *Step 1.* Randomly sample $\delta \cdot |U|$ instances from U and denote it as $U^{(t)}$ where δ is a sampling factor, experimentally set to 0.2. For each $(x_i, y_i) \in U^{(t)}$, compute the cluster ID as

$$p_i = \arg \min_{k=1,\cdots,K} \|\mathbf{v}(x_i) - \mathbf{v}(y_i) - \mathbf{c}_k^{(t)}\|$$

We first compute the difference $d^{(t)}(x_i, y_i)$ as

$$d^{(t)}(x_i, y_i) = \|\mathbf{M}_{p_i} \cdot \mathbf{v}(x_i) + \mathbf{b}_{p_i} - \mathbf{v}(y_i)\|$$

The smaller the difference is, the larger the probability of there being an *is-a* relation between x_i and y_i is. The prediction result of our model is:

$$f_M^{(t)}(x_i, y_i) = I(d^{(t)}(x_i, y_i) < \epsilon)$$

where ϵ is a pre-defined threshold and $I(\cdot)$ is an indicator function that outputs 1 if the condition holds and 0 otherwise. We use $U_-^{(t)}$ to represent word pairs in $U^{(t)}$ predicted as "positive" in this step.

– *Step 2.* For each $(x_i, y_i) \in U_-^{(t)}$, predict the label (*is-a* or *not-is-a*) by the pattern-based relation selection method (introduced in Sect. 4.5), denoted as $f_P^{(t)}(x_i, y_i)$. Define $U_+^{(t)}$ to be the extracted *is-a* relations with high confidence at the tth iteration:

$$U_+^{(t)} = \{(x_i, y_i) \in U_-^{(t)} | f_P^{(t)}(x_i, y_i) = 1\}$$

Update the two datasets as follows: (i) $U = U \setminus U_+^{(t)}$ and (ii) $R^{(t+1)} = R^{(t)} \cup U_+^{(t)}$. This means only candidate *is-a* relation instances that are predicted as "positive" by both the updated piecewise linear projection model and the pattern-based method can be added to the training set.

– *Step 3.* Denote the collection of *is-a* pairs in $U_+^{(t)}$ that belongs to the kth cluster as $U_k^{(t)}$. Update the cluster centroid $\mathbf{c}_k^{(t)}$ as follows:

$$\mathbf{c}_k^{(t+1)} = \mathbf{c}_k^{(t)} + \lambda \cdot \frac{1}{|U_k^{(t)}|} \sum_{(x_i, y_i) \in U_k^{(t)}} (\mathbf{v}(x_i) - \mathbf{v}(y_i) - \mathbf{c}_k^{(t)})$$

where λ is a learning rate in (0, 1) that controls the speed of cluster centroid "drift" over time. Re-assign the membership of clusters $C_k^{(t+1)}$ for each $(x_i, y_i) \in R^{(t+1)}$ based on new centroids.

– *Step 4.* For each cluster $C_k^{(t+1)}$, update model parameters by minimizing the objective function:

$$J(\mathbf{M}_k^{(t+1)}, \mathbf{b}_k^{(t+1)}; C_k^{(t+1)}) = \frac{1}{|C_k^{(t+1)}|} \sum_{(x_i, y_i) \in C_k^{(t+1)}} \|\mathbf{M}_k^{(t+1)} \cdot \mathbf{v}(x_i) + \mathbf{b}_k^{(t+1)} - \mathbf{v}(y_i)\|^2$$

with the initial parameter values $\mathbf{M}_k^{(t+1)} = \mathbf{M}_k^{(t)}$ and $\mathbf{b}_k^{(t+1)} = \mathbf{b}_k^{(t)}$.

Model Prediction After the training phase, given a pair (x_i, y_i) in the test set, our method predicts that x_i is the hyponym of y_i if at least one of the following conditions holds:

1. (x_i, y_i) is in the transitive closure of $R^{(T+1)}$ (based on *transitivity* property of *is-a* relations).
2. $f_M^{(T+1)}(x_i, y_i) = 1$ (based on final model prediction).

Discussion The key techniques of the algorithm lie in two aspects: (i) combination of *semantic* and *syntactic-lexico is-a* extraction and (ii) incremental learning. The positive relation selection method in Step 2 can also be regarded as a variant of coupled learning [2]. We ensure that only when the results of semantic projection and pattern-based approach are consistent, these relations are added to our training set. Also, at each iteration, the model parameters are updated incrementally. By solving the recurrent formula, the update rule of centroids in Step 3 is equivalent to:

Table 4 Examples of Chinese hypernym/hyponym patterns

Category	Examples	English Translation
Is-A	x_i是一个y x_i是一种y x_i是y之一	x_i is a y x_i is a kind of y x_i is one of y
Such-As	y，例如x_i、x_j y，包括x_i、x_j x_i、x_j等y y，特别是x_i、x_j	y, such as x_i and x_j y, including x_i and x_j x_i, x_j and other y y, especially x_i and x_j
Co-Hyponym	x_i、x_j等 x_i和x_j x_i以及x_j	x_i, x_j and others x_i and x_j x_i and x_j

y is a candidate hypernym and x_i and x_j are candidate hyponyms appeared in the corpus

$$\mathbf{c}_k^{(T+1)} = (1-\lambda)^{\mathrm{T}} \cdot \mathbf{c}_k^{(1)} + \lambda \cdot \sum_{t=1}^{\mathrm{T}} \left(\frac{(1-\lambda)^{T-t}}{|U_k^{(t)}|} \cdot \sum_{(x_i,y_i)\in U_k^{(t)}} (\mathbf{v}(x_i) - \mathbf{v}(y_i) - \mathbf{c}_k^{(t)})\right)$$

We can see that $\mathbf{c}_k^{(T+1)}$ is a weighted average of vector offsets of *is-a* relations added into the cluster, where the weight increases exponentially over time. With cluster assignments and prediction models updated, our models gradually fit the semantics of new *is-a* relations extracted from the unlabeled dataset.

4.5 Pattern-based relation selection

We now introduce the pattern-based approach used in Step 2 of the iterative learning process. Although Chinese patterns for relation extraction cannot guarantee high Precision and coverage, we employ them as a "validation" source for model-based extraction results. The goal of this method is to select only a small portion of relations as $U_+^{(t)}$ from $U_-^{(t)}$ with high confidence to add to the training set $R^{(t)}$.

Previously, Fu et al. [10] design several Chinese Hearst-style patterns manually for *is-a* extraction. In this paper, we collect a broader spectrum of patterns related to *is-a* relations, and categorize them into three types: "Is-A," "Such-As" and "Co-Hyponym." The examples are shown in Table 4[3]. We can see that an "Is-A" pattern establishes a one-to-one mapping from y to x_i. A "Such-As" pattern establishes a one-to-many mapping from y to x_i and x_j. Additionally, there is a possible co-hyponym relation between x_i and x_j appeared in a "Such-As" or "Co-Hyponym" pattern.

In summary, we have the following two observations:

- *Observation 4* If x_i and y match an "Is-A" or "Such-As" pattern, there is a large probability that x_i is the hyponym of y. Let $n_1(x_i, y)$ be the number of matches for x_i and y in a text corpus.
- *Observation 5* If x_i and x_j match a "Such-As" or "Co-Hyponym" pattern, there is a large probability that no *is-a* relation exists between x_i and x_j. Let $n_2(x_i, x_j)$ be the number of matches for x_i and x_j, and $n_2(x_i)$ be the number of matches for x_i and x^* where x^* is an arbitrary hyponym other than x_i.

[3] In practice, there can be over two candidate hyponyms in "Such-As" and "Co-Hyponym" patterns. For simplicity, we only list two here, denoted as x_i and x_j.

In this algorithm, we utilize the prediction results of projection models and Chinese hypernym/hyponym patterns jointly to decide which relations in $U_-^{(t)}$ should be added into $U_+^{(t)}$. For each $(x_i, y_i) \in U_-^{(t)}$, denote $PS^{(t)}(x_i, y_i)$ and $NS^{(t)}(x_i, y_i)$ as the positive and negative scores that indicate the level of confidence. We define the positive score based on the model prediction and the statistics from Observation 4:

$$PS^{(t)}(x_i, y_i) = \alpha \cdot (1 - \frac{d^{(t)}(x_i, y_i)}{\max_{(x,y)\in U_-^{(t)}} d^{(t)}(x, y)}) + (1-\alpha) \cdot \frac{n_1(x_i, y_i) + \gamma}{\max_{(x,y)\in U_-^{(t)}} n_1(x, y) + \gamma}$$

where $\alpha \in (0, 1)$ is a tuning weight to balance the two factors and γ is a smoothing parameter. For simplicity, we empirically set $\alpha = 0.5$ and $\gamma = 1$ in this paper and leave the optimal settings for future research.

We define the negative score based on the statistics from Observation 5 as follows:

$$NS^{(t)}(x_i, y_i) = \log \frac{n_2(x_i, y_i) + \gamma}{(n_2(x_i) + \gamma) \cdot (n_2(y_i) + \gamma)}$$

A high negative score between x_i and y_i means the strong evidence of the frequent co-occurrence of x_i and y_i in "Such-As" or "Co-Hyponym" patterns. It means that x_i and y_i are likely to be co-hyponyms, indicating that there is a low probability of the existence of an *is-a* relation between them.

A bi-criteria optimization problem can be formed where positive and negative scores should be maximized and minimized simultaneously, which is hard to optimize. We further covert it into a positive score maximization problem with negative score constraints:

$$\max \sum_{(x_i,y_i)\in U_+^{(t)}} PS^{(t)}(x_i, y_i)$$
$$\text{s. t.} \sum_{(x_i,y_i)\in U_+^{(t)}} NS^{(t)}(x_i, y_i) < \theta, U_+^{(t)} \subset U_-^{(t)}, |U_+^{(t)}| = m$$

where m is the size of $U_+^{(t)}$ and θ is used to constrain negative score limits. This problem is a special case of the *budgeted maximum coverage problem* [13], which is NP-hard. Based on the proof in [13], the objective function is *monotone* and *submodular*, indicating a greedy algorithm can be employed to solve this problem efficiently.

We design a greedy relation selection algorithm with the accuracy of $1 - \frac{1}{e}$, shown in Algorithm 1. It starts with the initialization step to set $U_+^{(t)} = \emptyset$. After that, it iteratively adds a pair (x_i, y_i) in $U_-^{(t)}$ to $U_+^{(t)}$ to maximize the objective function, as long as no constraints are violated. The algorithm stops when m pairs are selected and added into $U_+^{(t)}$. Finally, for each $(x_i, y_i) \in U_-^{(t)}$, we make the prediction as: $f_P^{(t)}(x_i, y_i) = I((x_i, y_i) \in U_+^{(t)})$, used in Step 2 in our iterative learning method.

5 Experimental data construction

In this section, we introduce how we construct all the experimental data to evaluate the proposed approach. For the sake of completeness, we begin by describing the detailed statistics

Algorithm 1 Greedy Relation Selection Algorithm

Input: Collection of *is-a* relations $U_-^{(t)}$, a large Chinese text corpus.

Output: Collection of *is-a* relations $U_+^{(t)}$.

Initialize $U_+^{(t)} = \emptyset$;
while $|U_+^{(t)}| < m$ **do**
  Select candidate *is-a* pair with largest PS: $(x_i, y_i) = \arg\max_{(x_i,y_i)\in U_+^{(t)}} PS^{(t)}(x_i, y_i)$;
  Remove the pair from $U_-^{(t)}$: $U_-^{(t)} = U_-^{(t)} \setminus \{(x_i, y_i)\}$;
  if $NS^{(t)}(x_i, y_i) + \sum_{(x,y)\in U_+^{(t)}} NS^{(t)}(x, y) < \theta$ **then**
    Add the pair to $U_+^{(t)}$: $U_+^{(t)} = U_+^{(t)} \cup \{(x_i, y_i)\}$;
  end if
end while
return Collection of *is-a* relations $U_+^{(t)}$;

Table 5 Datasets summarization

Dataset	Positive	Negative	Unlabeled
Wiki taxonomy	7312	–	–
Unlabeled set	–	–	78,080
Development set	349	1071	–
Test set	1042	3223	–

Table 6 Size and accuracy of relations in the taxonomy

Relation	#Relation instances	Accuracy
subclass-of	85,072	$95.85 \pm 2.16\%$
instance-of	1,233,291	$97.80 \pm 0.86\%$
Total	1,317,956	$97.60 \pm 0.71\%$

of our initial taxonomy. Next, we describe all the datasets used in weakly supervised *is-a* relation extraction. The statistics of all our word pair datasets are summarized in Table 5.

5.1 Initial taxonomy

The data source of the initial taxonomy is the Chinese Wikipedia dump from September 12, 2014. Every title of articles in the Wikipedia dump is considered as a candidate entity after we filter out pages without useful information and remove list pages, redirect pages, disambiguation pages, template pages and administrative pages. Every category name in the Wikipedia category system is regarded as a candidate class. In total, we extract 677,246 candidate entities for Chinese taxonomy construction.

In the taxonomy, there are a total of 581,616 entities and 79,470 classes. We randomly select 2000 relations from each set of relations (i.e., *instance-of*, *subclass-of* and the whole *is-a* relations) and ask human annotators to manually label whether a relation instance is correct or not. We calculate the confidence interval of accuracy with significance level $\alpha = 0.05$. As shown in Table 6, the accuracy is over 95% for both *instance-of* and *subclass-of* relations.

To understand the topological structure of the constructed taxonomy, we measure the depth and breadth of the taxonomy tree. We find that the depth ranges from 3 to 9, and the breadth ranges from 87 to 882,473. We also evaluate the ability of the taxonomy on abstraction and

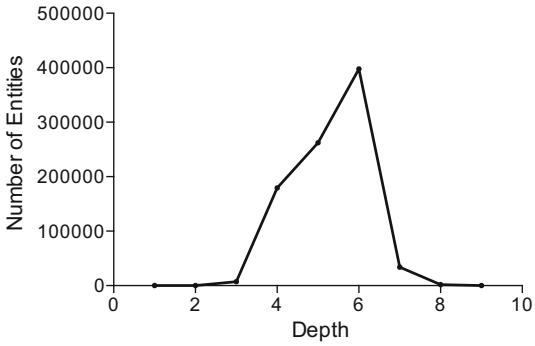


Fig. 2 Distribution of depth of entities in the initial taxonomy

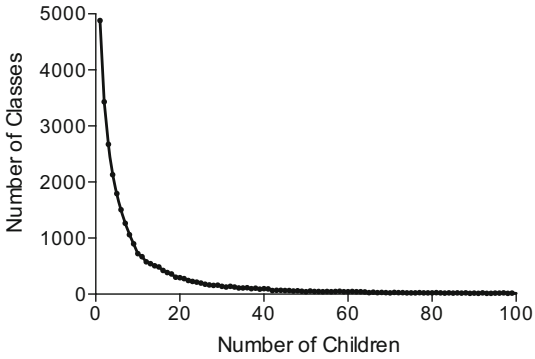


Fig. 3 Distribution of the numbers of children of classes in the initial taxonomy

expression of entities. In Fig. 2, it shows that entities with the depth of 6 account for 44% of the entity set. We also count the number of children (i.e., subclasses and entities) for each class. As shown in Fig. 3, the number of classes decreases rapidly as the number of children increases.

To train the initial projection models, we randomly sample a subset of *is-a* relations from the transitivity closure of the taxonomy as the positive training data. Because the constructed taxonomy is not 100% accurate, we ask human annotators to filter out incorrect *is-a* relation instances. Finally, we obtain 7,312 true *is-a* relations out of 7,500 pairs sampled from the taxonomy.

5.2 Web text corpus

To learn word embeddings, we crawl 1.2M Web pages from *Baidu Baike* and extract the contents to form a Chinese text corpus. We use the open-source toolkit *Ansj*[4] for Chinese word segmentation and filter out noisy contents. The entire text corpus consists of 1.088B words and 5.8M distinct words. Finally, we train a *Skip-gram* model to obtain 100-dimensional embedding vectors of all the 5.8M words. We calculate the positive and negative scores between word pairs for the pattern-based method in Sect. 4.5 using the same corpus.

[4] http://nlpchina.github.io/ansj_seg/.

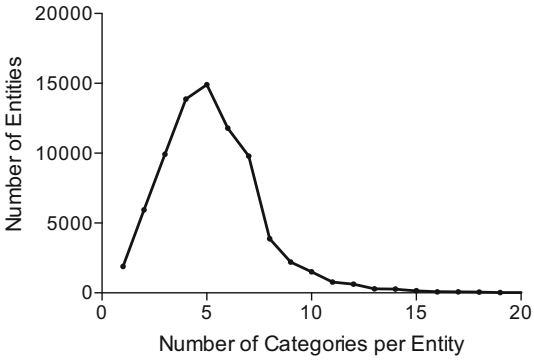


Fig. 4 Distribution of the number of categories per entity in the unlabeled pair set

5.3 Unlabeled user-generated categories

To construct the unlabeled set (i.e., candidate *is-a* relations), we randomly sample 0.1M entities from the *Baidu Baike* corpus. We extract all the user-generated categories of these entities, filter out entities without user-generated categories and finally obtain 78K <entity, category> pairs. Based on the statistical analysis, the average number of categories per entity is 4.10. The distribution of the number of categories per entity is illustrated in Fig. 4.

5.4 Test set

To our knowledge, the only publicly available dataset for evaluating Chinese *is-a* relation extraction is published in [9], containing 1391 *is-a* relations and 4294 unrelated entity pairs. We use it to evaluate our method by splitting the dataset into 1/4 for parameter tuning and 3/4 for testing randomly.

6 Experimental results

In this section, we conduct comprehensive experiments to evaluate the proposed approach. We also compare it with state-of-the-art methods and present an application based on the extended Chinese taxonomy to make the convincing conclusion.

6.1 Initial learning step

In the initial step, we train the piecewise linear projection models based on the initial taxonomy. Given a word pair (x_i, y_i), we predict it as a positive *is-a* relation if $\|\mathbf{M}_{p_i} \cdot \mathbf{v}(x_i) + \mathbf{b}_{p_i} - \mathbf{v}(y_i)\| < \epsilon$ where p_i is the cluster ID of (x_i, y_i) and ϵ is a threshold. To tune the parameters of the initial model, we run the K-means algorithm several times and train projection models. The number of clusters K ranges from 5 to 30, and we vary the threshold ϵ from 0.85 to 1.20. In Fig. 5, we report the detailed evaluation results on the development set. The evaluation metrics that we employ are Precision, Recall and F-Measure.

From the experimental results, we can see that our method is not very sensitive to the number of clusters K. When we set $K = 10$, our initial model achieves the best performance with a 73.9% F-measure, as shown in Fig. 5b. When K is too small, the different linguistic regularities in a collection of *is-a* relations are not well distinguished. On the other hand, if K is too large, the numbers of training data in some clusters tend to be small, leading to poor training effects.

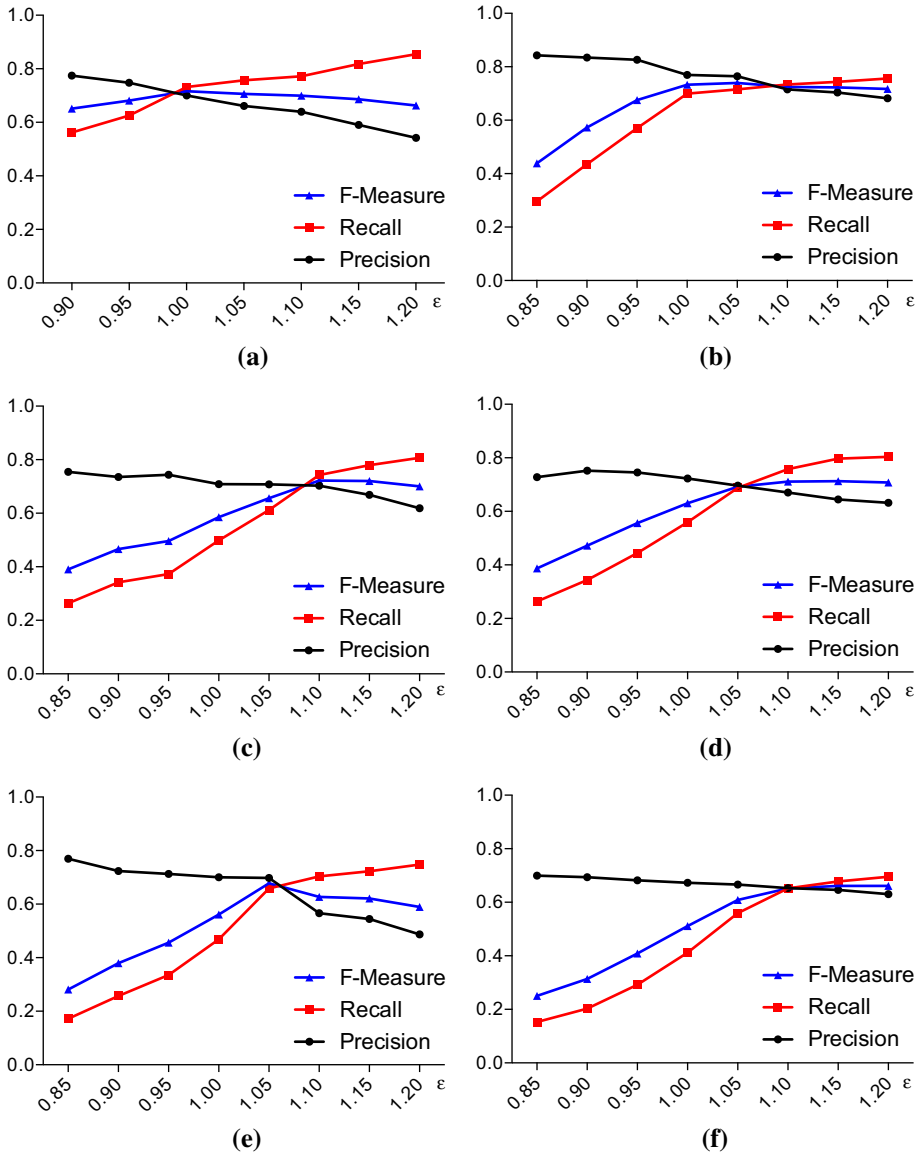


Fig. 5 Tuning of the number of clusters K and the threshold parameter ϵ in the initial model training step over the development set. **a** $K = 5$, **b** $K = 10$, **c** $K = 15$, **d** $K = 20$, **e** $K = 25$, **f** $K = 30$

We also vary the value of parameter ϵ in different settings of the cluster number K. As illustrated in Fig. 5, the Precision decreases and the Recall increases when ϵ becomes larger. In terms of F-Measure, the optimal setting of ϵ ranges from 0.95 to 1.15 in our experiments. When $K = 10$, the highest F-measure is achieved when ϵ is set to 1.05.

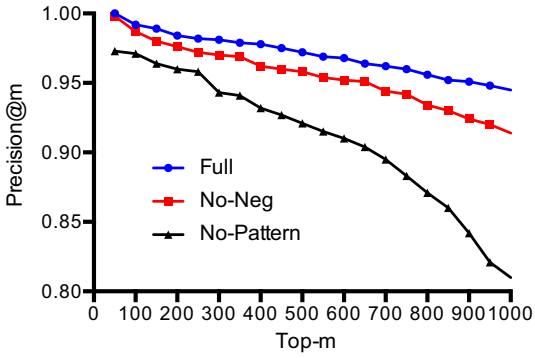


Fig. 6 Precision of selected *is-a* relations based on Chinese hypernym/hyponym patterns

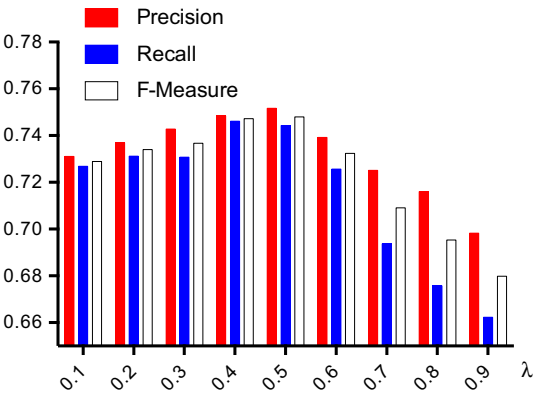


Fig. 7 Tuning of parameter λ over the development set

6.2 Iterative learning step

In this subsection, we illustrate the performance of the iterative learning method in various aspects. The accuracy of the selected *is-a* relations in Sect. 4.5 is essential to guarantee the performance of our iterative learning algorithm. We first use our initial model and Chinese hypernym/hyponym patterns to select top-m word pairs from unlabeled user-generated categories. Each time, we estimate the Precision of the extracted *is-a* relations by randomly sampling and labeling 20% of these pairs. We also implement two variants of our approach. In Fig. 6, *Full* is the complete implementation of our method in Sect. 4.5. *No-Neg* refers to the method which only maximizes the sum of positive scores, ignoring the negative score constraints. *No-Pattern* does not use any Chinese hypernym/hyponym patterns and only considers the prediction results of the projection model. The number m ranges from 50 to 1,000, and we report the performance using the evaluation metric of Precision@m.

From the experimental results, we can see that our method has the Precision of over 95% even when $m = 1000$. This means these extracted *is-a* pairs can be safely put into the training set in the next iteration. Therefore, we can update the projection models and clustering results without human intervention as long as we restrict that the number m is not too large. *No-Neg* performs slightly worse than *Full* (i.e., a 3% drop in Precision with $m = 1000$), indicating the effectiveness of negative score constraints. This means that the "Such-As" and "Co-Hyponym" patterns can help us filter out some of the false positives in the relation selection phase. The Precision of *No-Pattern* drops in a faster rate than *Full* and *No-Neg* when m increases. We set $m = 500$ in the next experiments because the Precision

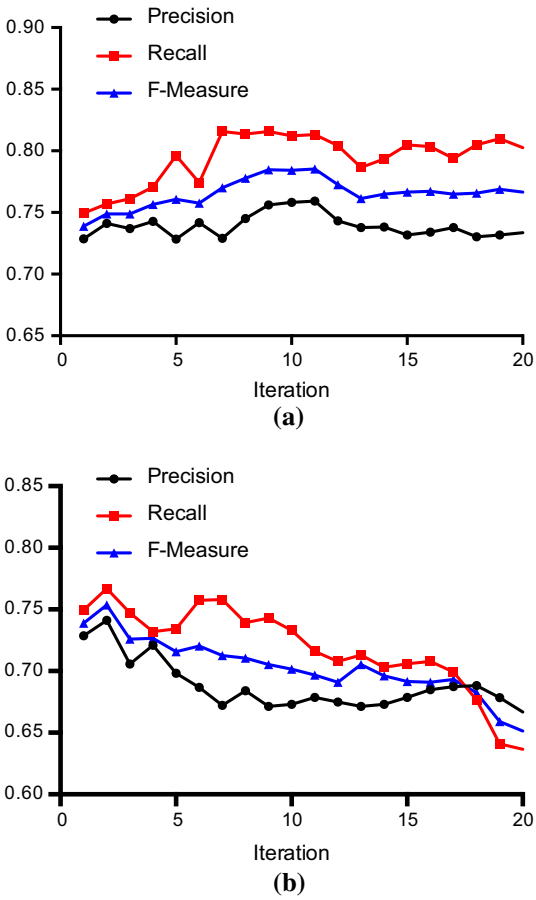


Fig. 8 Performance of the iterative learning method in the first 20 iterations over the development set. **a** With the pattern-based relation selection method, **b** with no pattern-based relation selection method

of extracted *is-a* pairs is over 97%. This prevents the injection of a lot of false positives in a large number of iterations.

We continue to run our iterative algorithm for 10 iterations and test the influence the parameter on λ where λ is a "learning rate" that controls the speed of the drift of cluster centroids. The performance w.r.t. parameter λ on the development set is illustrated in Fig. 7. The algorithm has the highest performance when $\lambda = 0.5$ with the F-Measure of 74.8%.

To have a high-level picture of the iterative algorithm, we report the performance in the first 20 iterations on the development set. The parameter settings are $\lambda = 0.5$ and $m = 500$ based on previous experiments. In Fig. 8a, these new *is-a* relations are selected based on Algorithm 1. The F-measure increases from 74.9 to 78.5% in the first 10 iterations, which shows that newly extracted *is-a* relations can be of help to boost the performance of our models. The F-measure slightly drops and finally keeps stable after 15 iterations with F-measure around 76.7%. The possible cause of the drop is that a few false positive pairs are still inevitably selected by Algorithm 1 and added to the training set. After manual checking of these pairs, the average accuracy is 98.8%. Some of the erroneous cases include:

脂肪(Fat), 健康(Health)
萧亚轩(Elva Hsiao), 时尚(Fashion)
信息(Information), 科学(Science)

They express *topic-of* relations rather than *is-a* relations. The performance becomes stable because the newly selected *is-a* relations tend to be similar to ones already in the training set after a sufficient number of iterations. In Fig. 8b, we directly sample 500 word pairs that are predicted as "positive" into our training set. Despite the slight improvement in the first iteration, the performance drops significantly because a large number of false positive instances are added to the training set for projection learning.

6.3 Comparison with previous methods

We evaluate the proposed approach and compare it with previous state-of-the-art methods on the test set. The results are shown in Table 7.

We first re-implement three corpus-based *is-a* relation extraction methods on the *Baidu Baike* corpus. The pattern matching method for English *is-a* relations is originally proposed in [11]. For a Chinese corpus, we implement this method by employing Chinese Hearst-style patterns translated by Fu et al. [10]. The result shows that hand-craft patterns have a low coverage for Chinese *is-a* relation extraction because the language expressions in Chinese are usually flexible. The automatic pattern detection approach in [26] improves the Recall from 19.8 to 28.1%. However, the Precision is dropped by 28.9% because the syntactic parser for Chinese is still not sufficiently accurate, causing errors in feature extraction. The distributional similarity measure introduced in [16] has a 58.1% F-measure and is not effective for our task because the contexts of entities in the free text are sparse and noisy. We also directly take our initial taxonomy based on Chinese Wikipedia [18] to match *is-a* relations in the test set. The result has a 98.5% Precision but low Recall due to the limited coverage of *is-a* relations in Chinese Wikipedia. The state-of-the-art word embedding-based approach in [9] achieves the highest F-measure 73.3% compared to all the previous methods. It shows the projection of word embeddings can model the semantics of Chinese *is-a* relations well.

We now discuss our weakly supervised relation extraction method (abbreviated as WSRE) and its variants. In Table 7, *WSRE (Initial)* refers to the *is-a* extraction models trained in the initial stage. Although it is similar to [9], F-measure is improved by 2% compared to [9] because we consider both vector offsets and matrix projection in *is-a* representation learning, which is more precise. *WSRE (Random)*, *WSRE (No-Neg)* and *WSRE* employ the iterative

Table 7 Performance comparison between different methods on the test set

Method	Precision (%)	Recall (%)	F-measure (%)
Previous methods			
Hearst [11]	96.2	19.8	32.8
Snow [26]	67.3	28.1	39.6
Taxonomy [18]	**98.5**	25.4	40.4
DSM [16]	48.5	58.1	52.9
Embedding [9]	71.7	74.9	73.3
Our method and its variants			
WSRE (initial)	74.1	76.7	75.3
WSRE (random)	69.0	75.7	72.2
WSRE (No-Neg)	75.4	80.1	77.6
WSRE	75.8	81.4	78.6
WSRE+taxonomy	78.8	**84.7**	**81.6**

Bold numbers mean that the score is the highest among all the methods

learning process for *is-a* extraction. In *WSRE (Random)*, new *is-a* relations added to the training set are selected randomly from word pairs predicted as "positive" by our model. *WSRE (No-Neg)* considers only maximizing positive scores in relation selection, ignoring the effects of negative scores. *WSRE* is the full implementation of our method. Based on the results, the performance of *WSRE (Random)* decreases because of false positives in the training set. The F-measure of the latter two methods is increased by 2.3% and 3.3%, respectively, compared to *WSRE (Initial)*, which indicates that the proposed approach can improve prediction performance and generalization ability. *WSRE* outperforms *WSRE (No-Neg)* by 1% in F-measure, which shows the negative score constraints reduce the error rate in the relation selection process. Overall, our approach outperforms the state-of-the-art method [9] by 5.3% in F-measure. We further combine our method with the initial Chinese taxonomy (*WSRE+Taxonomy*) and achieve an 81.6% F-measure, which is also better than Fu's method combined with the extension of a manually-built hierarchy, as shown in [9].

6.4 Error analysis

We analyze errors occurred in our algorithm. The majority of the errors (approximately 72%) stem from the difficulty in distinguishing *related-to* and is-a relations. Some word pairs in the test set have very close semantic relations but are not strict *is-a* relations. Take the pair 中药 (Traditional Chinese medicine), 药草 (Herb) as an example. Most major components in traditional Chinese medicine are herbs. However, Herb is not a hypernym of Traditional Chinese medicine from a medical point of view. These cases are difficult to handle without additional knowledge. The errors in the iterative learning process (discussed in Sect. 6.2) also contribute to inaccurate prediction of this type.

The rest of the errors are caused by the inaccurate representation learning for fine-grained hypernyms. Take an example of the hyponym 兰科 (Orchids) in the test set, our algorithm recognizes that 植物 (Plant) is a correct hypernym, but it does not regard 单子叶植物纲 (Monocotyledon) as a correct hypernym. The most probable cause of this error is that 单子叶植物纲 (Monocotyledon) rarely appears in the corpus and is not well represented in the embedding space. We will improve learning of word and relation embeddings in the future.

We also provide some examples from the unlabeled word pairs based on the prediction of the proposed approach, shown in Table 8. These examples show that our approach is generally effective to distinguish *is-a* or *not-is-a* relations from user-generated categories with no other contexts available. However, we have to admit that there are still inevitable errors in a few cases.

6.5 Application and case study

To further demonstrate the effectiveness of the proposed approach and visualize the taxonomy, we have applied the *is-a* relation extraction method in *TaxVis*, a taxonomy visualization and query system, illustrated in Fig. 9. The system is implemented in JAVA and employs the MySQL database to manage the taxonomy data. Besides providing the basic statistics of the constructed Chinese taxonomy, the system provides an interface for users to look deeply into the taxonomy by two types of queries, i.e., taxonomy query and *is-a* relation query. The screenshots are shown in Fig. 9a, b, respectively.

The taxonomy query takes a class name and a layer size as input and outputs a sub-taxonomy rooted from that class. Due to space limitation, for each class in the sub-taxonomy, the system only visualizes at most k hyponyms of the class which again have the top-k

Table 8 Examples of model prediction for user-generated categories

Category	P	T	Category	P	T
Entity: 北京大学(Peking University)			**Entity**: 黄贯中(Paul Wong)		
中国大学(University in China)	√	√	歌手(Singer)	√	√
大学(University)	√	√	港台明星(Star in Hong Kong/Taiwan)	√	√
机构(Organization)	√	√	香港(Hong Kong)	×	×
胡适(Hu Shih)	×	×	演员(Actor)	√	√
学校(School)	√	√	明星(Star)	√	√
Entity: 红火蚁(Solenopsis Invicta)			**Entity**: 新陈代谢(Process of Metabolism)		
动物(Animal)	√	√	疾病(Disease)	×	×
昆虫(Insect)	√	√	常见疾病(Common Disease)	×	×
节肢动物(Arthropod)	√	√	现象(Phenomenon)	√	√
蚂蚁(Ant)	√	√	自然现象(Natural Phenomenon)	√	√
Entity: 氰化钾(Potassium Cyanide)			**Entity**: 胰岛素(Insulin)		
化学品(Chemical)	√	√	药品(Drug)	√	√
无机物(Inorganic Substance)	√	√	**糖尿病**(Diabetes)	√	×
毒理学(Toxicology)	×	×	内科学(Internal Medicine)	×	×
钾盐(Sylvite)	√	√	**内分泌**(Endocrine)	√	×
Entity: 歼击机(Fighter)			**Entity**: 橄榄石(Olivine)		
飞机(Plane)	√	√	**自然科学**(Natural Science)	×	×
军用飞机(Military Plane)	√	√	矿石(Ore)	√	√
军事装备(Military Equipment)	√	√	矿物(Mineral)	√	√
航空(Aviation)	×	√	火成岩(Igneous Rock)	√	√
Entity: 可口可乐(Coca Cola)			**Entity**: 肾结石(Kidney Stone)		
品牌(Brand)	√	√	疾病(Disease)	√	√
生活(Life)	×	×	结石(Calculus)	√	√
软饮料(Soft Drink)	√	√	肾病(Kidney Disease)	√	√
食品(Food)	√	×	**肾脏**(Kidney)	√	×

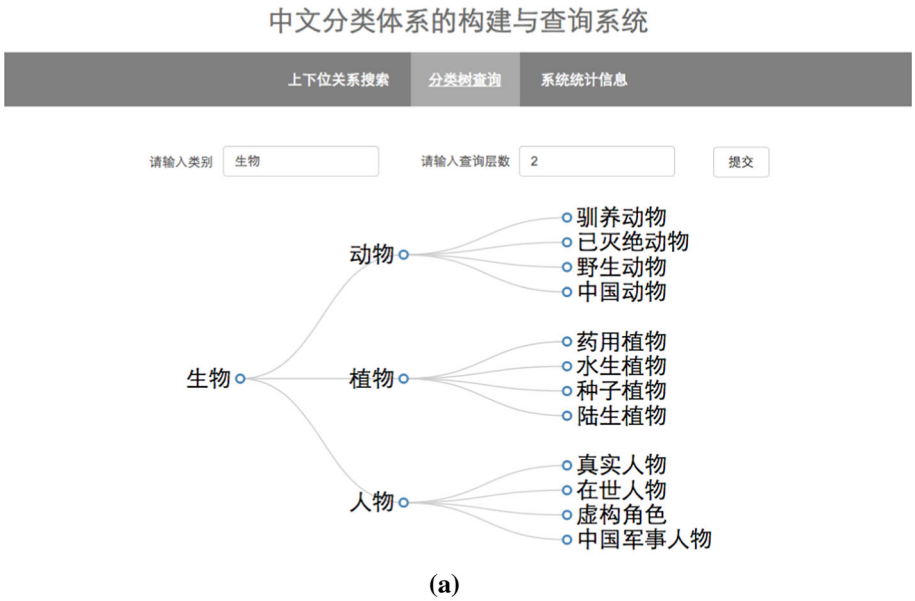
Bold category names indicate errors in prediction. (√: Positive, ×: Negative, P: Prediction result, T: Ground truth)

largest numbers of hyponyms in the entire taxonomy. This process repeats until a sub-taxonomy of the given layer size is retrieved or the leaves of taxonomy are reached. Figure 9a illustrates the two-layer sub-taxonomy for 生物 (Creature). Three hyponyms of 生物 (Creature) are 动物 (Animal), 植物 (Plant) and 人物 (Human). The second layer shows hyponyms of these hyponyms, such as 驯养动物 (Domesticated Animal), 野生动物 (Wild Animal), 已灭绝动物 (Extinct Animal), 中国动物 (Animal in China), etc.

The *is-a* relation query returns hyponyms and hypernyms given a certain entity or class. Similar to the taxonomy query, it retrieves at most k hyponyms and at most k hypernyms of that entity or class. As shown in Fig. 9b, the class 城市 (City) has three hypernyms (i.e., 行政区划 (Administrative Division), 聚居地 (Settlement of Different Types) and 地区 (Region)), and several hyponyms, including 巨型都市 (Huge City), 全球城市 (Global City), 独立市 (Independent City), 中国城市 (City in China), etc.

7 Conclusion and future work

In this paper, we propose to extract Chinese *is-a* relations from user-generated categories based on an initial Chinese taxonomy. Specifically, the task can be divided into two steps: initial model training and iterative learning. In the initial stage, word embedding-based piecewise linear projection models are trained on the Chinese taxonomy to map entities to hypernyms. Next, an iterative learning process combined with a pattern-based relation selection algo-



(a)

(b)

Fig. 9 *TaxVis*: a taxonomy visualization and query system. **a** Screenshot of taxonomy query, **b** screenshot of *is-a* relation query

rithm is introduced to update models without human supervision. Experimental results show that this approach outperforms state-of-the-art methods. However, our experiments illustrate that free-text Chinese relation extraction still suffers from low coverage. In the future, we aim at addressing this issue by learning better entity and relations representations under the guidance of existing knowledge.

Acknowledgements We thank anonymous reviewers for their very useful comments and suggestions. This work is supported by the National Key Research and Development Program of China under Grant No. 2016YFB1000904. Chengyu Wang is partially supported by the ECNU Outstanding Doctoral Dissertation Cultivation Plan of Action under Grant No. YB2016040. This manuscript is an extended version of the paper "Chinese Hypernym-Hyponym Extraction from User Generated Categories" presented at COLING 2016 [30]. The Chinese taxonomy construction technique is based on our previous work, which was presented at APWeb 2015, entitled "User Generated Content Oriented Chinese Taxonomy Construction" [18].

References

1. Cai J, Utiyama M, Sumita E, Zhang Y (2014) Dependency-based pre-ordering for chinese-english machine translation. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, pp 155–160
2. Carlson A, Betteridge J, Kisiel B, Settles B, Hruschka ER., Jr., Mitchell, TM (2010) Toward an architecture for never-ending language learning. In: Proceedings of the twenty-fourth AAAI conference on artificial intelligence
3. Carlson A, Betteridge J, Wang RC, Hruschka Jr. ER, Mitchell TM (2010) Coupled semi-supervised learning for information extraction. In: Proceedings of the third international conference on web search and web data mining, pp 101–110
4. de Melo G, Weikum G (2014) Taxonomic data integration from multilingual wikipedia editions. Knowl Inf Syst 39(1):1–39
5. Diaz F, Mitra B, Craswell N (2016) Query expansion with locally-trained word embeddings. In: Proceedings of the 54th annual meeting of the association for computational linguistics
6. Dong X, Gabrilovich E, Heitz G, Horn W, Lao N, Murphy K, Strohmann T, Sun S, Zhang W (2014) Knowledge vault: a web-scale approach to probabilistic knowledge fusion. In: 20th ACM SIGKDD international conference on knowledge discovery and data mining, pp 601–610
7. Dong Z, Dong Q, Hao C (2010) Hownet and its computation of meaning. In: Proceedings of the 23rd International Conference on Computational Linguistics, Demonstrations Volume, pp 53–56
8. Etzioni O, Fader A, Christensen J, Soderland S, Mausam M (2011) Open information extraction: The second generation. In: Proceedings of the 22nd international joint conference on artificial intelligence, pp 3–10
9. Fu R, Guo J, Qin B, Che W, Wang H, Liu T (2014) Learning semantic hierarchies via word embeddings. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, pp 1199–1209
10. Fu R, Qin B, Liu T (2013) Exploiting multiple sources for open-domain hypernym discovery. In: Proceedings of the 2013 conference on empirical methods in natural language processing, pp 1224–1234
11. Hearst MA (1992) Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th international conference on computational linguistics, pp 539–545
12. Hua W, Wang Z, Wang H, Zheng K, Zhou X (2015) Short text understanding through lexical-semantic analysis. In: 31st IEEE international conference on data engineering, pp 495–506
13. Khuller S, Moss A, Naor J (1999) The budgeted maximum coverage problem. Inf Process Lett 70(1):39–45
14. Kotlerman L, Dagan I, Szpektor I, Zhitomirsky-Geffet M (2010) Directional distributional similarity for lexical inference. Nat Lang Eng 16(4):359–389
15. Lehmann J, Isele R, Jakob M, Jentzsch A, Kontokostas D, Mendes PN, Hellmann S, Morsey M, van Kleef P, Auer S, Bizer C (2015) Dbpedia—a large-scale, multilingual knowledge base extracted from wikipedia. Semant Web 6(2):167–195
16. Lenci A, Benotto G (2012) Identifying hypernyms in distributional semantic spaces. In: Proceedings of the sixth international workshop on semantic evaluation, pp 543–546
17. Li H-G, Wu X, Li Z, Wu G (2013) A relation extraction method of chinese named entities based on location and semantic features. Appl Intell 38(1):1–15
18. Li J, Wang C, He X, Zhang R, Gao M (2015) User generated content oriented chinese taxonomy construction. In: Web technologies and applications—17th Asia-Pacific web conference, pp 623–634
19. Li PP, Wang H, Zhu KQ, Wang Z, Wu X (2013) Computing term similarity by large probabilistic isa knowledge. In: Proceedings of 22nd ACM international conference on information and knowledge management, pp 1401–1410
20. Lin T, Mausam, Etzioni O (2012) No noun phrase left behind: Detecting and typing unlinkable entities. In: Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning, pp 893–903

21. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv:1301.3781
22. Mikolov T, Yih W, Zweig G (2013) Geoffrey Linguistic regularities in continuous space word representations. In: Human language technologies: conference of the North American chapter of the association of computational linguistics, pp 746–751
23. Miller GA (1995) Wordnet: a lexical database for english. Commun ACM 38(11):39–41
24. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing, pp 1532–1543
25. Ponzetto SP, Strube M (2007) Deriving a large-scale taxonomy from wikipedia. In: Proceedings of the twenty-second AAAI conference on artificial intelligence, pp 1440–1445
26. Snow R, Jurafsky D, Ng AY (2004) Learning syntactic patterns for automatic hypernym discovery. In: Advances in neural information processing systems 17, NIPS 2004, pp 1297–1304
27. Suchanek FM, Kasneci G, Weikum G (2007) Yago: a core of semantic knowledge. In: Proceedings of the 16th international conference on world wide web, pp 697–706
28. Tomás D, González JLV (2013) Minimally supervised question classification on fine-grained taxonomies. Knowl Inf Syst 36(2):303–334
29. Wang C, Gao M, He X, Zhang R (2015) Challenges in chinese knowledge graph construction. In: 31st IEEE international conference on data engineering workshops, pp 59–61
30. Wang C, He X (2016) Chinese hypernym-hyponym extraction from user generated categories. In: Proceedings of the 26th international conference on computational linguistics, pp 1350–1361
31. Wang Z, Li J, Li S, Li M, Tang J, Zhang K, Zhang K (2014) Cross-lingual knowledge validation based taxonomy derivation from heterogeneous online wikis. In: Proceedings of the twenty-eighth AAAI conference on artificial intelligence, pp 180–186
32. Wong MK, Abidi SSR, Jonsen ID (2014) A multi-phase correlation search framework for mining non-taxonomic relations from unstructured text. Knowl Inf Syst 38(3):641–667
33. Wu W, Li H, Wang H, Zhu KQ (2012) Probase: a probabilistic taxonomy for text understanding. In: Proceedings of the ACM SIGMOD international conference on management of data, pp 481–492
34. Yang MC, Duan N, Zhou M, Rim HC (2014) Joint relational embeddings for knowledge-based question answering. In: Proceedings of the 2014 conference on empirical methods in natural language processing, pp 645–650
35. Yu Z, Wang H, Lin X, Wang M (2015) Learning term embeddings for hypernymy identification. In: Proceedings of the twenty-fourth international joint conference on artificial intelligence, pp 1390–1397
36. Zhang J, Liu S, Li Mu, Zhou M, Zong C (2014) Bilingually-constrained phrase embeddings for machine translation. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, pp 111–121
37. Zhou G, Zhu Z, He T, Hu XT (2016) Cross-lingual sentiment classification with stacked autoencoders. Knowl Inf Syst 47(1):27–44
38. Zhou H, Chen L, Shi F, Huang D (2015) Learning bilingual sentiment word embeddings for cross-language sentiment classification. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing of the asian federation of natural language processing, pp 430–440

Chengyu Wang is a Ph.D. candidate in School of Computer Science and Software Engineering, East China Normal University (ECNU), China. He received his BE degree in software engineering from ECNU in 2015. His research interests include Web data mining, information extraction and natural language processing. He is working on the construction and application of large-scale knowledge graphs.

Yan Fan is a master student in School of Computer Science and Software Engineering, East China Normal University (ECNU), China. She received her BE degree in software engineering from ECNU in 2016. Her research interests include relation extraction and natural language processing for large-scale knowledge graphs.

Xiaofeng He is a professor in computer science at School of Computer Science and Software Engineering, East China Normal University, China. He obtained his Ph.D. degree from Pennsylvania State University, USA. His research interests include machine learning, data mining and information retrieval. Prior to joining ECNU, he worked at Microsoft, Yahoo Labs and Lawrence Berkeley National Laboratory.

Aoying Zhou is a professor in computer science at East China Normal University (ECNU), where he is heading School of Data Science and Engineering. Before joining ECNU in 2008, Aoying worked for Fudan University at the Computer Science Department for 15 years. He is the winner of the National Science Fund for Distinguished Young Scholars supported by NSFC and the professorship appointment under Changjiang Scholars Program of Ministry of Education. He is now acting as a vice-director of ACM SIGMOD China and Database Technology Committee of China Computer Federation. He is serving as a member of the editorial boards VLDB Journal, WWW Journal, and etc. His research interests include data management, memory cluster computing, big data benchmarking and performance optimization.