Decoding Chinese User Generated Categories for Fine-Grained Knowledge Harvesting

Chengyu Wang, Yan Fan, Xiaofeng He, *Member, IEEE*, and Aoying Zhou, *Member, IEEE*

Abstract—User Generated Categories (UGCs) are short but informative phrases that reflect how people describe and organize entities. UGCs express semantic relations among entities implicitly hence serve as a rich data source for knowledge harvesting. However, most UGC relation extraction methods focus on English and heavily rely on lexical and syntactic patterns. Applying them directly to Chinese UGCs poses significant challenges because Chinese is an analytic language with flexible language expressions. In this paper, we aim at harvesting fine-grained relations from Chinese UGCs automatically. Based on neural networks and negative sampling, we introduce two word embedding projection models to identify *is-a* relations. The accuracy of prediction results is improved via a collective refinement algorithm and a hypernym expansion method. We further propose a graph clique mining algorithm to harvest *non-taxonomic* relations from UGCs, together with their textual patterns. Two experiments are conducted to validate our approach based on Chinese Wikipedia. The first experiment verifies the *is-a* relation extraction approach achieves high accuracy, outperforming state-of-the-art methods. The second experiment shows that the proposed method can harvest *non-taxonomic* relations of large quantity and high accuracy, with minimal human intervention.

Index Terms—User generated category, relation extraction, weakly supervised learning, word embedding, graph clique mining

1 Introduction

1.1 Motivation

User Generated Categories (UGCs) are short but descriptive phrases related to entities, frequently appearing in online encyclopedias and vertical websites. These texts are concise and informative, reflecting the way people organize and characterize entities [1].

UGCs (especially Wikipedia categories) are important data sources for knowledge harvesting. A variety of research works have been conducted to turn such human-generated short texts into machine-readable structured knowledge. For example, several approaches [2], [3], [4] focus on inferring *is-a* relations between entities and categories in Wikipedia, in order to construct a large-scale taxonomy. A few other methods aim at extracting multiple types of relations or properties from Wikipedia categories, decoding UGCs into attribute-value pairs [5], [6], [7]. The extracted facts (usually in the form of $<$subject, predicate, object$>$ triples) serve as indispensable building blocks to construct knowledge bases.

However, methods for decoding UGCs are highly language dependent. Existing approaches are mostly designed for English by employing textual patterns and linguistic rules [4], [5], and handcrafting regular expressions [6]. For Chinese, harvesting fine-grained semantic relations from UGCs poses different challenges. This is because Chinese is an analytic language with very flexible expressions [8]. For example, there is no distinction between singular and plural forms and no word spaces in Chinese. Word orders can be arranged in multiple ways in a phrase. As illustrated in [9], [10], the research of relation extraction from Chinese texts makes less significant process.

Although a few approaches have been proposed to construct Chinese taxonomies from Wikipedia categories [11], [12], extracting fine-grained and multi-typed relations from UGCs still needs further study. This is because there exist few high-quality lexical patterns for relation extraction in Chinese UGCs (in contrast to [5], [6]). Decoding Chinese UGCs will not only benefit the population of existing Chinese taxonomies and knowledge bases, but also enables the deep understanding of Chinese short texts.

1.2 Overview of Our Approach

In this paper, we aim at harvesting fine-grained relations from Chinese UGCs in a weakly supervised manner, without pre-defined relation types. Consider the simple example taken from Chinese Wikipedia page "Tim Berners-Lee", illustrated in Fig. 1. An *is-a* relation was predicted between Tim Berners-Lee and Londoner. We extract a "born-in" relation between him and the year of 1955 from "1955 births". The category "Winner of Turing Award" is more complicated, which can either serve as a class (hypernym) of "Tim Berners-Lee" (similar to Probase [13]), or be treated as a relational category expressing the relation "win-prize" between Tim Berners-Lee and Turing Award (similar to YAGO [6]). We regard both settings are correct and extract the two relations. The category "History of the Internet" roughly indicates the implicit and weak connection between him and the Internet, without a clear mention

- *C. Wang, Y. Fan, and X. He are with the School of Computer Science and Software Engineering, East China Normal University, Shanghai 200062, China. E-mail: {chywang2013, eileen940531}@gmail.com, xfhe@sei.ecnu.edu.cn.*
- *A. Zhou is with the School of Data Science and Engineering, East China Normal University, Shanghai 200062, China. E-mail: ayzhou@dase.ecnu.edu.cn.*

Manuscript received 26 Feb. 2018; revised 26 June 2018; accepted 13 Aug. 2018. Date of publication 17 Aug. 2018; date of current version 3 July 2019. (Corresponding author: Xiaofeng He.)
Recommended for acceptance by J. Chen.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TKDE.2018.2865942

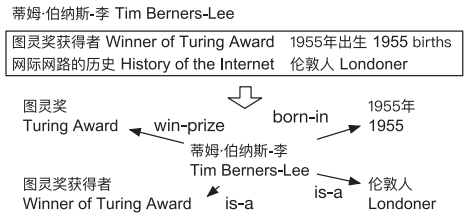


Fig. 1. An illustrative example w.r.t. the entity "Tim Berners-Lee". The four UGCs are taken from Chinese Wikipedia with English translations.

of semantic relations. Therefore, we simply discard this category.

For *is-a* relations, due to the lack of high-coverage Chinese textual patterns, we propose a three-step method based on word embeddings, summarized as follows.

1) Inspired by [14], [15], we propose two word embedding projection models to classify entity-category pairs into two labels: *is-a* and *not-is-a*. The first model is based on regularized linear projections, and the second one employs a neural network architecture with negative sampling techniques.
2) We observe that some UGCs are naturally suitable to be hypernyms for entities (e.g., "Londoner" in Fig. 1), while others are not hypernyms no matter what the entities are (e.g., "1955 births"). Thus, a collective refinement algorithm is proposed to refine the prediction results calculated in Step 1.
3) The head words of some hypernyms are also valid hypernyms of their entities. We introduce a hypernym expansion step to generate more hypernyms for entities with a high level of abstraction.

It is more challenging to harvest *non-taxonomic* relations because both relation types and relation patterns are unknown. We propose a weakly supervised approach based on graph clique mining as follows.

1) A single-pass category pattern miner is employed to extract category patterns appearing frequently in Chinese Wikipedia. They have a high probability to represent semantic relations.
2) We model the discovery of seed relation instances as a graph clique mining problem, propose an approximate algorithm for detecting maximum edge weight cliques. More relation instances are extracted based on the similarity between candidate and seed relation instances.
3) The generated "raw" relations are mapped to canonicalized relation triples. Based on syntax and semantics of category patterns, the relation predicates are either generated automatically or defined manually.

1.3 Contributions and Organization

In summary, we make the following major contributions:

- We propose a word embedding based approach to identify *is-a* relations from Chinese entity-category pairs. It includes two projection models, collective refinement and hypernym expansion.
- We design a graph clique mining algorithm to extract *non-taxonomic* relation patterns and relation instances from Chinese UGCs jointly.
- We conduct extensive experiments over multiple datasets generated from Chinese Wikipedia to illustrate the effectiveness of the proposed approach.

The rest of this paper is as follows. Section 2 summarizes the related work. We introduce our framework and notations in Section 3. Details of the proposed methods are described in Sections 4 and 5. The experimental results are presented in Section 6. Finally, we conclude our paper in Section 7.

2 Related Work

In this section, we overview the related work on *is-a* and *non-taxonomic* relation extraction from UGCs.

2.1 *Is-a* Relation Extraction

Is-a relations (also called hypernym-hyponym relations) express the type or class of entities, which are backbones in taxonomies and knowledge bases. In YAGO and its improved version YAGO3 [6], [16], a Wikipedia category is regarded as a conceptual category if it matches the pattern "*pre-modifier + head word + post-modifier*" (e.g., German people of Jewish descent). WikiTaxonomy [3] constructs a taxonomy from Wikipedia categories based on the link connectivity in the Wikipedia concept network and lexico-syntactic matching techniques. This taxonomy is reconstructed and improved in [4] by combining the Wikipedia category system and the top-level taxonomy from WordNet. In the Wikipedia Bitaxonomy project [2], Flati et al. propose a self-contained approach to build two taxonomies simultaneously, for pages and categories of Wikipedia respectively. Other similar projects use lexico-syntactic patterns, classifiers and rule based inference to predict *is-a* relations for taxonomy learning [17], [18]. Since harvesting English *is-a* relations from UGCs is not our focus, we do not elaborate here. Readers can refer to a survey for more details [19].

In terms of Chinese, this task is more challenging because there are few category patterns that can be used to extract *is-a* relations from UGCs. Based on the word formation of Wikipedia categories, Li et al. [11] translate the patterns used in [6] into Chinese and propose a classification method to build a large Chinese taxonomy from Wikipedia. Besides encyclopedias, Fu et al. [20] generate candidate hypernyms using several linguistic heuristics and employ an SVM-based ranking model to predict the most likely hypernym of an entity. These methods have high precision but require careful feature engineering involving large amount of human work.

Another thread of related work is cross-lingual approaches which use larger English knowledge sources to supervise Chinese *is-a* relations extraction. For example, Wang et al. [12] propose a dynamic adaptive boosting model to learn taxonomic prediction functions for English and Chinese. In their method, the training sets of entity-category pairs in English and Chinese are continuously expanded during a cross-lingual knowledge validation mechanism. Wu et al. [21] propose a bilingual biterm topic model to align English and Chinese taxonomies. These methods take advantages of languages with richer resources but are constrained by cross-lingual links. In fact, they are not suitable for cases where human-annotated cross-lingual links are missing and can not deal with language-specific or culture-specific entities in Chinese.

To capture linguistic regularities of *is-a* relations, deep learning approaches map the embedding vectors of entities to the vectors of their hypernyms. Fu et al. [14]'s pioneer work in this field employs piecewise linear projection models to learn Chinese semantic hierarchies based on word embeddings. Wang et al. [22] improve this approach by adding an iterative update strategy and a pattern-based validation mechanism.

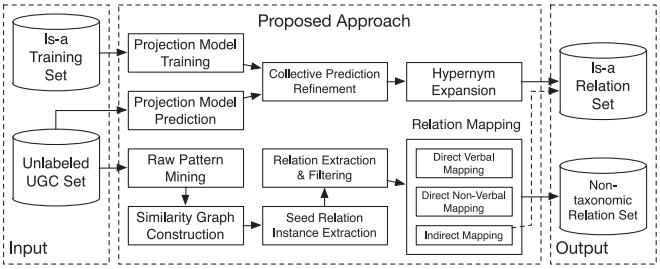


Fig. 2. The system architecture of the proposed approach.

Based on the widely-used Hearst patterns [23], they propose the counterparts in Chinese, and find that using Chinese Hearst-style patterns can improve the semi-supervised representation learning of *is-a* relations. Another following-up work is proposed by Yamane et al. [24]. In this work, they learn the number of clusters and projection matrices jointly, rather than using a fixed number of clusters as in [14]. Additionally, a transductive learning model is introduced by Wang et al. [15], which considers the semantics of both *is-a* and *not-is-a* relations, linguistic rules and the unlabeled data jointly. Empirical study shows that the use of word embeddings benefits *is-a* relation extraction for Chinese. In this work, we further propose two word embedding based projection models and a collective refinement algorithm that considers the word formation of UGCs to improve prediction results.

2.2 *Non-Taxonomic* Relation Extraction

Unlike the case of *is-a* relations, the task of extracting *non-taxonomic* relations from UGCs has rarely been addressed. A possible reason is that harvesting relations from short texts is more challenging. Suchanek et al. [6] use regular expression based matching to harvest relations from relational categories. A similar work [5] extracts relations by lexical pattern matching and inference. Pasca [7] studies how to decompose Wikipedia categories into attribute-value pairs. Compared to more regular patterns in English, enumerating patterns for Chinese requires a large amount of human labor. In our work, we solve this problem by graph mining, which has high precision with minimal human intervention.

Semantic parsing is a similar task for short text understanding. It turns noun phrases into logical expressions aligned with knowledge bases. For example, Surtani and Paul [25] introduce a vector space based statistical model to interpret noun-modifier pairs. In their work, a relation is represented by a weighted vector of prepositional and verbal paraphrases. Choi et al. [26] consider the situation of incomplete knowledge bases. Their approach learns partial mapping from texts to knowledge bases without requiring all input words be mapped to concepts in Freebase. However, this approach is not suitable for Chinese due to the lack of high-coverage Chinese knowledge bases [8]. Our experiments also reveal that most of the extracted relations from Chinese UGCs are not present in a large Chinese knowledge base CN-DBpedia [27].

There is another line of research called "Open Information Extraction" (OIE) [28], which aims at extracting relation triples from free texts, without pre-defined relation types. However, algorithms for OIE are not applicable to address the task of relation extraction from Chinese UGCs. OIE requires the syntax and dependency parsing of sentences in a text corpus to generate relation candidates. The subject, verb and object in a sentence are usually treated as a candidate relation instance. In contrast, UGCs are basically short phrases instead of complete sentences.

TABLE 1
Important Notations

Notation	Description
e	An entity in Wikipedia
$\mathbf{v}(e)$	The embedding vector of entity e
$Cat(e)$	The collection of UGCs of entity e in the Wikipedia category system
$f(e,c)$	The final prediction score of entity e and category c
H	The head word set of all Wikipedia categories
c_h	The head word of category c
$Proj^+(e)$	The *is-a* neural projection result of entity e
$Proj^-(e)$	The *not-is-a* neural projection result of entity e
$s(e,c)$	The model prediction score of entity e and category c
$g(h)$	The global prediction score of head word h
p	A pattern generated from Wikipedia categories, consisting of words and an entity tag "[E]"
(e_p, c_p)	An entity pair such that a category $c \in Cat(e_p)$ matches pattern p and c_p is in the place of "[E]"
R_p	Candidate relation instances w.r.t. pattern p
R_p^*	Seed relation instances w.r.t. pattern p
$R_p^{'}$	Extracted relation instances w.r.t. pattern p
$G_p^{'}$	The similarity graph constructed from R_p
$supp(p)$	The support score of pattern p
$conf(p)$	The confidence score of pattern p

3 General Framework and Notations

In this section, we present an overall framework of the proposed approach. The architecture of our system is illustrated in Fig. 2, with important notations summarized in Table 1.

In Wikipedia, each entity e is associated with a collection of UGCs $Cat(e)$ in the category system. The goal for *is-a* relation extraction is to learn a model $f(e,c)$ to predict whether there is an *is-a* relation between an entity e and its Wikipedia category c where $c \in Cat(e)$ (Section 4). For example, we can obtain the *is-a* relation "(蒂姆·伯纳斯-李 (Tim Berners-Lee), *is-a*, 伦敦人 (Londoner))", as shown in Fig. 1. Because applying classification to distinguish *is-a* and *not-is-a* relations directly can cause the problem of lexical memorization [19], [29], we propose two projection models (i.e., regularized linear projection and neural projection with negative sampling) to calculate the prediction score $s(e,c)$, indicating the probability of the existence of an *is-a* relation between entity e and category c. Considering the word formation of Wikipedia UGCs, we compute a global head word prediction score $g(h)$ for all the head words in the categories in the collective prediction refinement step. The model $f(e,c)$ is obtained by combining $s(e,c)$ and $g(h)$ where h is the head word of c. Finally, we expand the extracted *is-a* relation set using hypernym expansion heuristics.

For *non-taxonomic* relations, relational patterns in English are mostly based on prepositional expressions [5]. In Chinese, prepositions are usually expressed implicitly and hence English patterns are not applicable for Chinese. We present an automatic pattern mining approach for UGCs. Our algorithm first makes a single pass over all categories to mine significant category patterns (Section 5.1). For example, the pattern "[E]获得者 (Winner of [E])", which

frequently appears in UGCs, is extracted. It may refer to a type of relation where "[E]" is a placeholder for entities. Candidate relation instances for such patterns are obtained by a graph clique mining algorithm (Section 5.2). The example instances extracted based on the previous pattern are "(蒂姆·伯纳斯-李 (Tim Berners-Lee), 图灵奖 (Turing Award))", "(约翰·科克 (John Cocke), 美国国家科学奖 (National Medal of Science))". Finally, the extracted "raw" instances are mapped to canonicalized triples using three types of relation mapping schemes (Section 5.3). In this step, a relation predicate "获奖 (win-prize)" is defined for the pattern and these pairs are mapped to "获奖 (win-prize)" relations. Note that this algorithm is mainly designed for *non-taxonomic* relations, with a few *is-a* relations extracted by indirect mapping rules.

4 Mining *Is-A* Relations

In this section, we introduce the three-step learning process to extract *is-a* relations from Chinese UGCs.

4.1 Projection-Based Model Prediction

We first design two heuristic rules to generate positive pairs from Wikipedia categories. We treat a pair (e, c) as positive if the following two conditions hold:

Rule 1. The category c matches the pattern "*pre-modifier* + 的 + *head word*" or the head words of e and c are the same. The head word of a category name is the root word in the dependency parsing tree. "的" is an auxiliary word in Chinese, usually appearing between adjectives and nouns.

Rule 2. The head word of a category name is a noun and is *not* in a Chinese thematic lexicon extended from the dictionary used in [11], containing 184 thematic words, e.g., "军事 (Military)", "娱乐 (Entertainment)", etc.

Except the previous pattern, other Chinese *is-a* relations can not be directly extracted. An intuitive method is to design a classifier to distinguish *is-a* and *not-is-a* relations, which fits in the category of supervised relation classification based on distributional semantics. However, this method has one drawback: it tends to learn the existence of prototypical hypernyms rather than the actual relations between the two terms. It is also called the problem of lexical memorization, studied in [29]. Readers can also refer to a recent survey for detailed discussion [19]. Inspired by our previous work [15], we present two projection models to learn semantics of *is-a* and *not-is-a* relations in the embedding space. In this way, we explicitly learn the representations of *is-a* and *not-is-a* relations and avoid the problem pointed out by [29].

4.1.1 Model One: Regularized Linear Projection

The regularized linear projection model maps the embedding vector of a word to the vector of another where the two words satisfy a particular relation. In Wikipedia, most category names are relatively long and fine-grained, making it difficult to learn the embeddings precisely. Based on the transitivity property of *is-a* relations [14], we only need to deal with the head words of categories. For example, if we predict that "科学家 (scientist)" is a hypernym of "蒂姆·伯纳斯-李 (Tim Berners-Lee)", we can infer that "英格兰计算机科学家 (computer scientist in England)" is also a valid hypernym. Formally, the rule can be stated as follows:

Rule 3. Given an entity-category pair (e, c) where $c \in Cat(e)$, if the head word c_h of category c is a valid hypernym of entity e, then the category c is a valid hypernym of entity e, too.

We first obtain the embedding vectors by training a Skip-gram model over a large text corpus. Denote $\mathbf{v}(e)$ as the embedding vector of entity e, with the dimensionality of n. For each pair (e, c) in the positive training set D^+, assume that the following linear projection model holds:

$$\mathbf{M}^+\mathbf{v}(e) + \mathbf{B}^+ \approx \mathbf{v}(c_h),$$

where $\mathbf{M}^+$ is an $n \times n$ projection matrix and $\mathbf{B}^+$ is an $n \times 1$ bias vector. Similarly, for pairs in the negative training set $(e', c') \in D^-$, we learn a negative model $\mathbf{M}^-\mathbf{v}(e') + \mathbf{B}^- \approx \mathbf{v}(c'_h)$. We do not impose explicit connections between two models because the semantics of Chinese *is-a* and *not-is-a* relations are complicated and difficult to model [14], [22]. Instead, we let the algorithms to learn representations of *is-a*/*not-is-a* relations. This approach does not require deep NLP analysis on UGCs, which is suitable to deal with the flexible expressions in Chinese.

In the training phase, we aim to minimize the objective function for positive projection learning

$$J^+(\mathbf{M}^+, \mathbf{B}^+) = \sum_{(e,c)\in D^+} \|\mathbf{M}^+\mathbf{v}(e) + \mathbf{B}^+ - \mathbf{v}(c_h)\|_F^2 + \frac{\lambda}{2}(\|\mathbf{M}^+\|_F^2 + \|\mathbf{B}^+\|_F^2),$$

where $\|\cdot\|_F$ is the Frobenius norm and $\lambda > 0$ gives an additional Tikhonov smoothness effect on the projection matrices. For the negative model, we have the objective

$$J^-(\mathbf{M}^-, \mathbf{B}^-) = \sum_{(e,c)\in D^-} \|\mathbf{M}^-\mathbf{v}(e) + \mathbf{B}^- - \mathbf{v}(c_h)\|_F^2 + \frac{\lambda}{2}(\|\mathbf{M}^-\|_F^2 + \|\mathbf{B}^-\|_F^2).$$

Minimizing the two objective functions $J^+(\mathbf{M}^+, \mathbf{B}^+)$ and $J^-(\mathbf{M}^-, \mathbf{B}^-)$ can be efficiently done based on matrix computation. We optimize $J^+(\mathbf{M}^+, \mathbf{B}^+)$ using the gradient descent algorithm where the partial derivatives are computed as

$$\frac{\partial J^+(\mathbf{M}^+, \mathbf{B}^+)}{\partial \mathbf{M}^+} = 2\sum_{(e,c)\in D^+} (\mathbf{M}^+\mathbf{v}(e) + \mathbf{B}^+ - \mathbf{v}(c_h))\mathbf{v}(e)^T + \lambda\mathbf{M}^+$$

$$\frac{\partial J^+(\mathbf{M}^+, \mathbf{B}^+)}{\partial \mathbf{B}^+} = 2\sum_{(e,c)\in D^+} (\mathbf{M}^+\mathbf{v}(e) + \mathbf{B}^+ - \mathbf{v}(c_h)) + \lambda\mathbf{B}^+.$$

$J^-(\mathbf{M}^-, \mathbf{B}^-)$ can be optimized similarly. Let D^U be the collection of unlabeled entity-category pairs. After model training, for each pair $(e, c) \in D^U$, if the category c is the correct hypernym of the entity e, the vector $\mathbf{v}(c_h)$ will be close to $\mathbf{M}^+\mathbf{v}(e) + \mathbf{B}^+$ and far away from $\mathbf{M}^-\mathbf{v}(e) + \mathbf{B}^-$. Denote $d^+(e, c)$ and $d^-(e, c)$ as

$$d^+(e, c) = \|\mathbf{M}^+\mathbf{v}(e) + \mathbf{B}^+ - \mathbf{v}(c_h)\|_2 \quad (1)$$

$$d^-(e, c) = \|\mathbf{M}^-\mathbf{v}(e) + \mathbf{B}^- - \mathbf{v}(c_h)\|_2. \quad (2)$$

The prediction score is calculated as follows:

$$s(e, c) = \tanh(d^-(e, c) - d^+(e, c)), \quad (3)$$

where $s(e, c) \in (-1, 1)$. High prediction score means that there is a large probability of the existence of an *is-a* relation

between entity e and category c. We use the hyperbolic tangent function here to avoid the widespread saturation of the sigmoid function.

4.1.2 Model Two: Neural Projection with Negative Sampling

As shown in [15], using non-linear projections can enhance the representation learning of *is-a* and *not-is-a* relations. In this part, we propose a neural network based architecture for projection learning.

Model Architecture. Given the positive training set $D^+ = \{(e, c)\}$, this approach tries to learn a non-linear mapping from the entity embedding $\mathbf{v}(e)$ to the embedding of the head word of its hypernym $\mathbf{v}(c_h)$. Let Θ be the set of parameters of the neural network. $Proj^+(e)$ is the non-linear *is-a* projection vector of entity e, which is the hypernym embedding of entity e predicted by the neural network. The loss function is defined as follows:

$$L^+(\Theta) = \frac{1}{2} \sum_{(e,c)\in D^+} \|Proj^+(e) - \mathbf{v}(c_h)\|_F^2 + \frac{\mu_1}{2} L_\Theta^+ + \mu_2 L_r^+,$$

where μ_1 and μ_2 are regularization hyper-parameters. $L_\Theta^+ = \|\Theta\|_2^2$ is the l_2 regularizer on Θ. L_r^+ gives an additional, implicit regularization effect on Θ, introduced as follows.

As discovered by Biemann et al. [30], the use of explicit negative examples significantly improves the performance of linear projection models. Here, we propose a generalized negative exampling method and integrate it into projection based neural networks. For a pair $(e, c) \in D^+$, denote $\tilde{c}^{(1)}$, $\tilde{c}^{(2)}, \ldots, \tilde{c}^{(k)}$ as k non-hypernym categories of entity e, which are treated as negative samples. The sampling process is introduced in detail in the next section. Denote the embeddings of their head words as: $\mathbf{v}(\tilde{c}_h^{(1)}), \mathbf{v}(\tilde{c}_h^{(2)}), \ldots, \mathbf{v}(\tilde{c}_h^{(k)})$. The "centroid" embedding of the negative samples is calculated as follows: $\mathbf{v}(\tilde{c}_h) = \frac{1}{k}\sum_{i=1}^{k} \mathbf{v}(\tilde{c}_h^{(i)})$. Because *is-a* relations are asymmetric relations, in the model, we force the predicted hypernym embedding $Proj^+(e)$ to be as dis-similar to $\mathbf{v}(\tilde{c}_h)$ as possible. We define the regularization function as follows:

$$L_r^+ = \sum_{(e,c)\in D^+} Proj^+(e)^T \cdot \mathbf{v}(\tilde{c}_h). \tag{4}$$

To minimize $L^+(\Theta)$, we design a neural network architecture illustrated in Fig. 3. It consists of two subnetworks: projection network and regularization network. The input of the projection network is $\mathbf{v}(e)$. It passes through m hidden layers $h^{(1)}, h^{(2)}, \ldots, h^{(m)}$, with output as $Proj^+(e)$. Next, $Proj^+(e)$ is fed into a regularization network which simply samples $\mathbf{v}(\tilde{c}_h^{(1)}), \mathbf{v}(\tilde{c}_h^{(2)}), \ldots, \mathbf{v}(\tilde{c}_h^{(k)})$ and computes $r = Proj^+(e)^T \cdot \mathbf{v}(\tilde{c}_h)$ as output. We employ the gradient descent algorithm to train the model. The partial derivative of $L^+(\Theta)$ is derived as follows:

$$\frac{\partial L^+(\Theta)}{\partial \Theta} = \sum_{(e,c)\in D^+} (Proj^+(e) - \mathbf{v}(c_h))^T \cdot \frac{\partial Proj^+(e)}{\partial \Theta} + \mu_1 \Theta + \mu_2 \sum_{(e,c)\in D^+} \mathbf{v}(\tilde{c}_h)^T \cdot \frac{\partial Proj^+(e)}{\partial \Theta},$$

where $\frac{\partial Proj^+(e)}{\partial \Theta}$ can be computed based on the back propagation of the projection neural network. The detailed implementations of hidden layers are quite flexible, which can be either fully connected layers, or more advanced ones such as convolutional and pooling layers. The last hidden layer uses the linear function to map $h^{(m)}$ to $Proj^+(e)$ in order to generate the predicted hypernym embeddings.

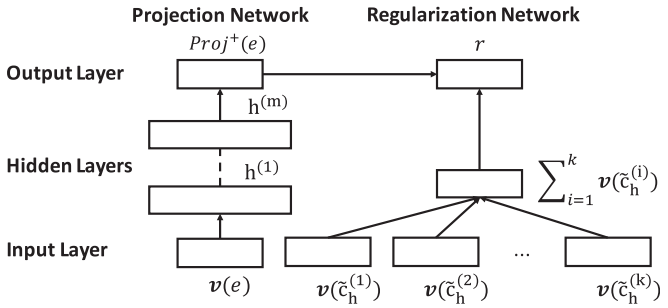


Fig. 3. The general neural network architecture to minimize $L^+(\Theta)$.

Similarly, to learn *not-is-a* relation projection, denote $Proj^-(e)$ as the non-linear *not-is-a* projection vector of entity e (i.e., the model predicted non-hypernym embedding of entity e). We minimize the loss function $L^-(\Theta)$, which is defined as follows:

$$L^-(\Theta) = \frac{1}{2} \sum_{(e,c)\in D^-} \|Proj^-(e) - \mathbf{v}(c_h)\|_F^2 + \frac{\mu_1}{2} L_\Theta^- + \mu_2 L_r^-,$$

where L_Θ^- and L_r^- are model regularizers which are defined the same as L_Θ^+ and L_r^+, only with the dataset changed from D^+ to D^-. The architecture of the model for *not-is-a* relation projection $Proj^-(e)$ is also the same as that of *is-a* relations. Due to space limitation, we do not elaborate the details here. Hence, we need to minimize the loss functions $L^+(\Theta)$ and $L^-(\Theta)$ of the two models.

In the testing phase, for each pair $(e, c) \in D^U$, we calculate $Proj^+(e)$ and $Proj^-(e)$ using the two models. The distance metrics are similar to Eqs. (1) and (2) of Model One, defined as

$$d^+(e, c) = \|Proj^+(e) - \mathbf{v}(c_h)\|_2$$

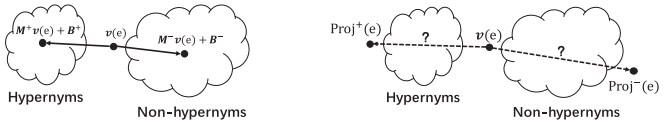
$$d^-(e, c) = \|Proj^-(e) - \mathbf{v}(c_h)\|_2.$$

Finally, we use Eq. (3) to calculate the prediction score.

Algorithm 1. Negative Sampling Algorithm

Input: Parameter η, sample size k, non-hypernym categories set $\tilde{C}_e$, head word set $\tilde{H}_e$.
Output: Sampled head word set $\tilde{C}_h$.

```
Initialize C̃_h = ∅;
while |C̃_h| < k do
  if random < η then
    Sample c from C̃_e with prob 1/|C̃_e|;
    Extract head word c_h = getHeadWord(c);
    Update C̃_h = C̃_h ∪ {c_h};
  else
    Sample h from H̃_e with prob ∝ count(h);
    Update C̃_h = C̃_h ∪ {h};
  end if
end while
return Sampled head word set C̃_h;
```



(a) Linear projection (b) Neural projection

Fig. 4. A graphical comparison between regularized linear projection and neural projection with negative sampling.

Negative Sampling Techniques. For each pair $(e, c) \in D^+$, there are two types of negative samples to be generated. The first type can be directly sampled from categories in D^-, which are the non-hypernym categories of entity e, i.e., $\tilde{C}_e = \{c|(e, c) \in D^-\}$.

However, we observe that non-hypernyms in the training set are limited and can not cover all linguistic circumstances. Hence, the second type of negative samples are generated following the "open world assumption" that any word outside the training set could be a non-hypernym of a certain entity. Denote H as the head word set of all Wikipedia UGCs. $\tilde{H}_e$ is the collection of head words to be sampled as non-hypernyms for entity e, defined as: $\tilde{H}_e = \{h|h \in H \wedge \forall(e, c) \in D^+, c_h \neq h\}$. The probability of sampling h from $\tilde{H}_e$ is linearly proportional to the number of occurrences of the head word h in all Wikipedia categories (denoted as $count(h)$), i.e.,

$$\Pr(h) = \frac{count(h)}{\sum_{h' \in \tilde{H}_e} count(h')}. \tag{5}$$

We use $\eta \in (0, 1)$ to control the ratio of the two types of negative samples. With probability η, we sample a category uniformly from $\tilde{C}_e$. With probability $1 - \eta$, we sample a head word from $\tilde{H}_e$ based on Eq. (5). The detailed sampling process is presented in Algorithm 1. $random$ is uniformly distributed in (0,1). It is worth noting that although we sample different types of textual units, in projection learning, we use head words of categories in both cases.

For negative (*not-is-a*) pairs, the negative samples are *is-a* relation pairs. For each pair $(e, c) \in D^-$, because the number of hypernyms of each entity is limited, we only sample from hypernym categories of entity e uniformly and do not consider "open world assumption" here.

4.1.3 Comparison

Fig. 4 illustrates the comparison between the two projection models. Linear projection model assumes that there is a linear transformation from the entity embedding vector $\mathbf{v}(e)$ to that of its hypernym $\mathbf{v}(\tilde{c}_h)$. This practice is also applied in other works [14], [22], [24]. However, the disadvantage is that it can not encode more complicated transformation and has a small parameter space. On the contrary, neural projection models allow algorithms to learn non-linear and complicated transformation from entities to their hypernyms and non-hypernyms in embedding space. The transformation does not need to be defined explicitly. Hence, this model has a large hypothesis space.

Another difference is how word vectors are projected in the embedding space. Take an *is-a* relation (e, c) as an example. As discussed earlier, directing applying a single classifier to distinguish *is-a* and *not-is-a* relations does not work from the view of computational linguistics. It causes the lexical memorization problem [29]. Therefore, we use two models to capture the semantics of *is-a* and *not-is-a* relations separately. The simple linear model maps the entity embedding vector $\mathbf{v}(e)$ to that of its hypernym and non-hypernym. By utilizing implicit regularization defined in Eq. (4), neural projection maps the entity embedding $\mathbf{v}(e)$ to $Proj^+(e)$, which is farther away from the negative sampling embedding $\mathbf{v}(\tilde{c}_h)$, hence the embeddings of its non-hypernyms. Although neural projection does not enforce a precise projection of *is-a* relations compared to linear projection, it makes the prediction embeddings of hypernym $Proj^+(e)$ and non-hypernym $Proj^-(e)$ more separable. Therefore, the negative sampling technique makes *is-a* and *not-is-a* relations easier to be separated. Readers can also refer to the paper [30] for a detailed study on how negative sampling improves *is-a* relation prediction. By using such technique, the performance of the relation classification can be improved.

4.2 Collective Prediction Refinement

As studied in previous research [20], [29], some categories naturally serve as "prototypical hypernyms", regardless of what the corresponding entities are. To encode this assumption into our method, we refine the previous prediction results by collective inference.

Take the category "伦敦人 (Londoner)" in Fig. 1 as an example. It can be literally translated as "伦敦 (London)人 (person)". "人 (person)" is the "prototypical hypernym" here. Other categories whose head words are "人 (person)" such as "巴黎(Paris)人 (person, Parisian)" and "纽约 (New York)人 (person, New Yorker)" are likely to be conceptual categories, too. Based on the prediction that there is an *is-a* relation between "蒂姆·伯纳斯-李 (Tim Berners-Lee)" and "伦敦人 (Londoner)", we can infer it is likely that "维克多·雨果 (Victor Hugo)" is a "巴黎人 (Parisian)".

Recall that H is the head word set of all Wikipedia UGCs. For each head word $h \in H$, let D_h^U be the collection of unlabeled pairs (i.e., pairs not in the training set) where the head word of category c is h. Besides the model prediction scores of unlabeled pairs, the training set generated based on Section 4.1 is also useful for collective prediction inference. Denote D_h^+ as the collections of positive pairs with h as the head word of c in the training set. We extend Eq. (3) to accommodate both training and testing data

$$s(e, c) = \begin{cases} \tanh(d^-(e, c) - d^+(e, c)) & (e, c) \in D_h^U \\ 1 & (e, c) \in D_h^+. \end{cases}$$

The global prediction score $\tilde{g}(h)$ for each head word $h \in H$ is defined as

$$\tilde{g}(h) = \ln(1 + |D_h^U| + |D_h^+|) \frac{\sum_{(e,c) \in D_h^U \cup D_h^+} s(e, c)}{|D_h^U| + |D_h^+|}.$$

In this formula, the weight of each unlabeled pair $(e, c) \in D_h^U$ is set according to the respective model prediction score. Each positive training data instance has the weight of 1. $\frac{\sum_{(e,c) \in D_h^U \cup D_h^+} s(e,c)}{|D_h^U| + |D_h^+|}$ is the average prediction score for categories with the head word h. $\ln(1 + |D_h^U| + |D_h^+|)$ gives a larger impact to $\tilde{g}(h)$ when the head word h appears more frequently in Wikipedia categories. This heuristic setting is related to transductive learning, which assumes that leveraging both training and unlabeled data can improve the prediction accuracy. It is also similar to the prior probability feature used in [20].

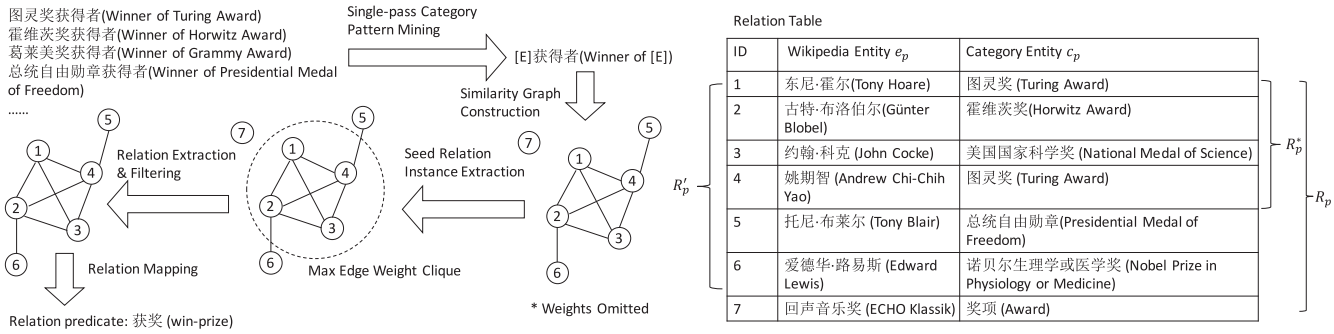


Fig. 5. The extraction process of the category pattern and relation instances of the "win-prize" relation from Chinese Wikipedia UGCs.

Because the range of the score $\tilde{g}(h)$ is unconstrained, we normalize the global prediction score $g(h)$ as follows:

$$g(h) = \frac{\tilde{g}(h)}{\max_{h' \in H \wedge \tilde{g}(h') > 0} |\tilde{g}(h')|}. \quad (6)$$

The prediction function $f(e, c)$ for the entity e and the category c with the head word h is defined in a linear combination of the model prediction score $s(e, c)$ and the head word global prediction score $g(h)$

$$f(e, c) = \beta s(e, c) + (1 - \beta) g(h), \quad (7)$$

where $\beta \in (0, 1)$ is a tuning parameter that controls the relative importance of the two scores.

Finally, we predict that there is an *is-a* relation between entity e and category $c \in Cat(e)$ if it satisfies at least one of the conditions: (i) The pair (e, c) satisfies Rule 1 and Rule 2 introduced in Section 4.1; and (ii) The score $f(e, c) > \theta$ where θ is a tuning threshold ($-1 < \theta < 1$).

4.3 Hypernym Expansion

We observe that the head words of some hypernyms are also valid hypernyms of their respective entities. For example, in Fig. 1, "人 (person)" extracted from "伦敦人 (Londoner)" is a hypernym of "蒂姆·伯纳斯-李 (Tim Berners-Lee)", but "获得者 (winner)", the head word of "图灵奖获得者 (Winner of Turing Award)" is not a suitable hypernym due to its incomplete semantics. Additionally, word segmentation and parsing errors may occur when we extract head words by NLP tools. In this study, we regard c_h as a valid hypernym of e if c is predicted as a hypernym of e and c_h is also a Wikipedia concept. This is because if c_h is a Wikipedia concept, the semantics of c_h is usually complete, which also means that the head word extraction process is probably correct. This step increases the number of *is-a* relations.

Note that more hypernyms can be generated by interpreting Chinese noun phrases correctly. For example, three hypernyms "英格兰科学家 (scientist in England)", "计算机科学家 (computer scientist)" and "科学家 (scientist)" can be generated from the category "英格兰计算机科学家 (computer scientist in England)". Currently, methods for noun phrase interpretation mostly focus on English (e.g., [6], [31]). We leave the deeper study for Chinese as future work.

5 Mining *Non-Taxonomic* Relations

In this section, we present the graph clique mining based approach to extract *non-taxonomic* relations from Wikipedia UGCs. An example of the "获奖 (win-prize)" relation extraction process is illustrated in Fig. 5.

5.1 Single-Pass Category Pattern Mining

This module automatically learns important category patterns that appear frequently in Wikipedia and have a probability to represent certain underlying semantic relations. Formally, a category pattern p consists of i) an ordered sequence of words that are a substring of a category and ii) an entity tag. For example, the pattern of the category "图灵奖获得者 (Winner of Turing Award)" is "[E]获得者 (Winner of [E])" where "[E]" is an entity tag that can represent any type of Wikipedia entities.

Define $R_p = \{(e_p, c_p)\}$ as the collection of entity pairs such that in Wikipedia page e_p, a category $c \in Cat(e_p)$ matches the pattern p. Without ambiguity, we use e_p to represent both the Wikipedia page with the title as e_p and the entity e_p itself. c_p is the entity in the place of "[E]" of the pattern p. Consider the previous example. In Wikipedia page "Tim Berners-Lee", there is a category "图灵奖获得者 (Winner of Turing Award)" that matches the pattern "[E]获得者 (Winner of [E])". "图灵奖 (Turing Award)" is the "[E]" here. Thus we have $e_p =$ "蒂姆·伯纳斯-李 (Tim Berners-Lee)" and $c_p =$ "图灵奖 (Turing Award)" as an entity pair $(e_p, c_p) \in R_p$. We can see that R_p is the collection of all candidate relation instances that may have the relation that p represents. Readers can refer to Table 2 for more examples.

Let $length(p)$ be the number of words (excluding entity tags) in pattern p. We define the support of the pattern $supp(p)$ as follows: $supp(p) = |R_p| \cdot \ln(1 + length(p))$ where $\ln(1 + length(p))$ gives larger support values to longer patterns because longer patterns tend to be more specific and may contain richer semantics.

In the implementation, we employ a CRF-based Chinese named entity tagger [32] and a dictionary consisting of all Wikipedia entities to recognize the entities and obtain these patterns. This step processes all the categories within a single pass and calculates the support values of all these patterns. It keeps top-k highest support patterns as the input of the next step, together with the matched entity pairs as candidate relation instances.

5.2 Graph-Based Raw Relation Extraction

Candidate relation instances R_p extracted from the previous step are not necessarily correct. For example, in Fig. 5, there is a category "奖项获得者 (Award Winner)" in the Wikipedia page "回声音乐奖 (ECHO Klassik)". Thus the pair "(回声音乐奖 (ECHO Klassik), 奖项 (Award))" will be extracted from the pattern "[E]获得者 (Winner of [E])".

In this part, for each top-k highest support pattern p, we select a subset of pairs R^*_p from R_p as seed relation instances for an underlying relation that the pattern p may represent. The seed relation instances R^*_p are most likely correct. After

TABLE 2
Examples of Relation Patterns and Their Candidate Relation Instances

Wikipedia Entity e_p	Category c	Pattern p	Category Entity c_p
黑客帝国2:重装上阵 The Matrix Reloaded	人工智能题材作品 Artificial intelligence theme works	[E]题材作品 [E] theme works	人工智能 Artificial intelligence
奥尔良 Orléans	卢瓦雷省市镇 City/town in Loiret	[E]省市镇 City/town in [E]	卢瓦雷 Loiret
教父 (电影) The Godfather (film)	奥斯卡最佳男主角获奖电影 Oscar Best Actor Award winning movie	[E]获奖电影 [E] winning movie	奥斯卡最佳男主角 Oscar Best Actor Award

that, we filter out low quality patterns and extract relation instances $R_p{}'$ from R_p as the final result based on the similarity between R_p^* and $R_p{}'$. The high-level relation extraction process is summarized in Algorithm 2.

Algorithm 2. Raw Relation Extraction Algorithm

Input: Candidate relation instance collection R_p.
Output: Extracted relation instance collection $R_p{}'$.
1: Construct similarity graph $G_p = (C_p, L_p, W_p)$ from R_p;
2: Detect maximum edge weight clique C_p^* by Algorithm 3;
3: Generate seed relation instance collection R_p^* by Eq. (8);
4: Initialize $R_p{}' = R_p^*$;
5: **for** each $(e_p, c_p) \in R_p \backslash R_p^*$ **do**
6: **if** (e_p, c_p) satisfies the criterion defined by Eq. (9) **then**
7: Update $R_p{}' = R_p{}' \cup \{(e_p, c_p)\}$;
8: **end if**
9: **end for**
10: **return** Extracted relation instance collection $R_p{}'$;

5.2.1 Seed Relation Instance Extraction

To select seed relation instances R_p^*, we propose an unsupervised graph mining approach. Let $G_p = (C_p, L_p, W_p)$ be a weighted, undirected graph (called similarity graph), where C_p, L_p, and W_p denote vertices, edges and edge weights, respectively. The vertices C_p correspond to the matched category entities for pattern p, i.e., $C_p = \{c_p | (e_p, c_p) \in R_p\}$. The edge weights W_p reflect the semantic similarities among entities in C_p. Because the link structure in Chinese Wikipedia is relatively sparse, the semantic similarity between entities c_p and $c_p{}'$ used in this paper is defined as follows:

$$sim(c_p, c_p{}') = \frac{\sum_{c \in Cat(c_p)} \sum_{c' \in Cat(c_p{}')} \cos(\mathbf{v}(c_h), \mathbf{v}(c'_h))}{|Cat(c_p)| \cdot |Cat(c_p{}')|},$$

where $\cos(\cdot)$ is a cosine function. Given a similarity threshold τ, iff $sim(c_p, c_p{}') > \tau$, we have $(c_p, c_p{}') \in L_p$ and $w(c_p, c_p{}') = sim(c_p, c_p{}')$. In this way, entities in C_p are interconnected if they are similar in semantics. Take the previous pattern "[E]获得者 (Winner of [E])" as an example. Entities such as "图灵奖 (Turing Award)", "霍维茨奖 (Horwitz Award)" and "诺贝尔生理学或医学奖 (Nobel Prize in Physiology or Medicine)" are very similar in semantics and should be inter-connected in G_p pairwisely.

In this paper, we model the problem of mining R_p^* from R_p as a Maximum Edge Weight Clique Problem (MEWCP) [33]. The goal of NEWCP is to detect a maximum edge weight clique from an undirected graph with edge weights. Recall that a maximum edge weight clique is a clique in which the sum of edge weights in the clique is the largest among all the cliques. In our work, we detect a maximum edge weight clique C_p^* from C_p in R_p to form R_p^*. The objective function is defined as follows:

$$\max \sum_{(c_p, c_p{}') \in L_p{}'} w(c_p, c_p{}')$$
$$\text{s.t. } L_p{}' \subseteq L_p \text{ and } \forall c_p \forall c'_p \in C_p^* (c_p \neq c'_p), (c_p, c'_p) \in L_p{}',$$

where $L_p{}'$ is the collection of edges such that both nodes are in the desired clique C_p^*.

Algorithm 3. Approximate Algorithm for Solving MEWCP

Input: Similarity graph $G_p = (C_p, L_p, W_p)$.
Output: Maximum edge weight clique C_p^*.
1: Initialize temp graph $G_p^* = (C_p^*, L_p^*)$ with $C_p^* = \emptyset$ and $L_p^* = \emptyset$;
2: **while** $L_p \neq \emptyset$ **do**
3: Sample (c_p, c'_p) from L_p with prob $\propto w(c_p, c'_p)$;
4: $C_p = C_p \setminus \{c_p, c'_p\}, C_p^* = C_p^* \cup \{c_p, c'_p\}$;
5: $L_p = L_p \setminus \{(c_p, c'_p)\}, L_p^* = L_p^* \cup \{(c_p, c'_p)\}$;
6: **for** each $(\tilde{c}_p, \tilde{c}'_p) \in L_p$ **do**
7: **if** $\tilde{c}_p \notin C_p^*$ and $\tilde{c}'_p \notin C_p^*$ **then**
8: $C_p = C_p \setminus \{\tilde{c}_p, \tilde{c}'_p\}$;
9: $L_p = L_p \setminus \{(\tilde{c}_p, \tilde{c}'_p)\}$;
10: **end if**
11: **end for**
12: **end while**
13: **return** Maximum edge weight clique C_p^*;

To produce a solution for MEWCP, several algorithms have been proposed in the optimization research community, e.g., unconstrained quadratic programming [33]. However, they suffer from high computational complexity due to the NP-Hardness of the problem. In this paper, we introduce an approximate algorithm based on Monte Carlo methods. The general procedure is shown in Algorithm 3. It starts with an empty graph G_p^* to represent the clique to be extracted. In each iteration, the algorithm selects an edge (c_p, c'_p) from G_p with the probability proportional to its weight $w(c_p, c'_p)$. After a particular edge (c_p, c'_p) is chosen, the algorithm adds the edge to G_p^*, and removes the edge and other edges that do not connect with any nodes in C_p^* from G_p. This process iterates until no more edges in G_p can be added to G_p^*. Thus, the vertices in G_p^* form the desired clique C_p^*. Based on the nodes in C_p^*, we can extract the seed relation instance collection R_p^* as follows:

$$R_p^* = \{(e_p, c_p) | c_p \in C_p^*, (e_p, c_p) \in R_p\}. \quad (8)$$

Because it is a random and approximate algorithm, the average runtime complexity depends on the input graph

structure. We can see that the worst-case runtime complexity is $O(|L_p|^2)$. We run it k times and produce multiple results. We select the clique with largest edge weights as the maximum edge weight clique for G_p. Thus the total runtime complexity is $O(k|L_p|^2)$. In this way, the NP-hard problem is effectively solved in quadratic time. Although this algorithm does not guarantee accurate results, we experimentally find that the seed relation instances are accurate even though the generated clique is not the largest in terms of the sum of edge weights. Therefore, the proposed algorithm is efficient and effective for solving our problem. Further research on MEWCP is beyond the scope of this paper.

5.2.2 Relation Extraction and Filtering

After the seed relation instances R_p^* are detected, we employ a confidence score to quantify the quality of pattern p. Intuitively, if pattern p represents entity pairs bearing the same clear semantic relation, both the size of R_p^* and the sum of edge weights in C_p^* will be sufficiently large. Here, we define the unnormalized confidence score of p as

$$\tilde{conf}(p) = \frac{\ln(1+|R_p^*|)}{|R_p^*| \cdot (|R_p^*|-1)} \sum_{c_p, c_p{}' \in C_p^*, c_p \neq c_p{}'} sim(c_p, c_p{}'),$$

where $\frac{\sum_{c_p, c_p{}' \in C_p^*, c_p \neq c_p{}'} sim(c_p, c_p{}')}{|R_p^*| \cdot (|R_p^*|-1)}$ is the average pairwise entity similarity in the clique, and $\ln(1+|R_p^*|)$ increases the confidence score of patterns with large cliques. To guarantee the range of confidence values in [0,1], we normalize the score as: $conf(p) = \frac{\tilde{conf}(p)}{\max_{p_i \in P} \tilde{conf}(p_i)}$ where P is the collection of all patterns. Based on the formula, patterns with low confidence scores can be filtered.

For the remaining patterns, given each $(e_p, c_p) \in R_p$, we add it to the final extracted relation instance collection $R_p{}'$ if $(e_p, c_p) \in R_p^*$ or it is similar enough to entity pairs in R_p^*. For example, in Fig. 5, our method extracts the pair "(托尼 • 布莱尔 (Tony Blair), 总统自由勋章 (Presidential Medal of Freedom))" in the final result and discard the pair "(回声音乐奖 (ECHO Klassik), 奖项 (Award))" based on the finding that "总统自由勋章 (Presidential Medal of Freedom))" is more similar to entities in the clique (i.e., full names of awards or prizes), rather than the abstract concept "奖项 (Award)". Denote γ as a parameter that controls the precision-recall trade-off where larger γ results in higher precision. The criteria is defined as follows:'

$$\frac{\sum_{c_p{}' \in C_p^*} sim(c_p, c'_p)}{|C_p^*|} > \frac{\gamma \sum_{c_p{}', c_p'' \in C_p^*, c_p{}' \neq c_p''} sim(c_p{}', c_p'')}{|R_p^*| \cdot (|R_p^*|-1)}. \quad (9)$$

In general, our method detects most probably correct pairs as "seeds" and extract other pairs that are similar enough to seeds. Because it is difficult to ensure high precision for short text relation extraction, we do not use iterative extraction method to avoid "semantic drift" [34]. A further discussion on this issue is presented in Section 6.

5.3 Relation Mapping

The final step is to map $R_p{}'$ to relation triples with a proper relation predicate. Based on the semantics of category patterns, we have three types of mappings:

Direct Verbal Mapping. If the head word of the pattern is a verb, we can use it as the relation predicate. For example, in "[E]出生 ([E] births)", "出生 (born in)" is expressed as a verb in Chinese and is taken as a predicate. The relation triple "(蒂姆 • 伯纳斯-李, 出生, 1955年) (Tim Berners-Lee, born-in, 1955)" can be generated automatically by adding the detected relation predicate.

Direct Non-Verbal Mapping. If the category pattern does not contain a verb but expresses a relation by one/many non-verbs, we define the relation predicate and map the entity pairs to relation triples by logical rules. For example, in the pattern "[E]获得者 (Winner of [E])", "获得者 (winner)" is a noun that indicates the "得奖 (win-prize)" relation. By defining the mapping rule, we can generate the relation triple "(蒂姆 • 伯纳斯-李, 获奖, 图灵奖) (Tim Berners-Lee, win-prize, Turing Award)".

Indirect Mapping. Similar to [6], a few patterns do not describe relations between entity pairs, but should be mapped to other relations indirectly. In "[E]军事 ([E] military)" the pattern indicates that the entity is related to the topic "军事(military)". Thus, we define a new relation predicate "话题(topic-of)" and establish the relations between entities and "军事 (military)".

There are also a few *is-a* relations that can be generated via indirect mapping. For example, the pattern "[E]单曲 ([E] digital single)" infers that the entity associated with the category is a song. However, most of the cases are related to *non-taxonomic* relations. We add such *is-a* relations into the extracted *is-a* relation set and do not elaborate here.

As seen, the only manual work in our approach is to define relation predicates for direct non-verbal mappings and indirect mappings. In our work, such logical mapping rules are required for only a couple of relation types. Therefore, the proposed approach needs very minimal human work.

6 Experiments

In this section, we conduct extensive experiments to evaluate our method and compare it with state-of-the-art approaches. We also present the overall extraction performance to make a convincing conclusion.

6.1 Data Source and Experimental Settings

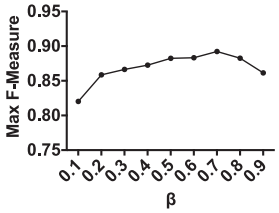
The Wikipedia data source is downloaded from the Chinese Wikipedia dump of the version January 20, 2017.[1] Because some Wikipedia pages are not related to entities, we design several heuristic rules to filter out disambiguation, redirect, template and list pages. Finally, we obtain 0.6M entities and 2.4M entity-category pairs. The open-source toolkit FudanNLP [32] is employed for Chinese NLP analysis. The word embeddings of 5.8M distinct Chinese words are trained via a Skip-gram model using a large Chinese text corpus from [22] and set to 100 dimensions.

6.2 Evaluation of *Is-a* Relation Extraction

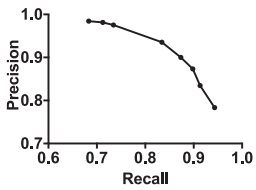
6.2.1 Dataset Generation

To avoid the time-consuming human labeling process, we generate the training set automatically. The first part of the training set is borrowed from [14], which is the first widely used dataset for predicting *is-a* relations among Chinese word pairs. It is also generated from UGCs, containing

1. http://download.wikipedia.com/zhwiki/20170120/



(a) Maximum F-measure when β varies from 0.1 to 0.9



(b) Precision-recall curve when θ varies from -1 to 1

Fig. 6. Parameter analysis of model one over development set.

1,391 positive pairs and 4,294 negative pairs. However, this dataset is relatively unbalanced and the number of positive pairs is not sufficient. Additionally, we sample 5,000 positive pairs that satisfy Rule 1 and Rule 2 to add to our training set. The true positive (TP) rate is 98.7 percent, estimated over 300 pairs, indicating the effectiveness of rules.

There exist a few testing sets of Chinese *is-a* relations (e.g., [20]). But they aim to learn *is-a* relations between short concepts/terms instead of (relatively long) categories and are not suitable for evaluating our work. For the testing set, we further expand the labeled dataset in our prior work [35]. We randomly select 2,800 entity-category pairs from Chinese Wikipedia and ask three human annotators to label the relations (i.e., *is-a* and *not-is-a*). We discard all the pairs that have inconsistent labels across different annotators and obtain a dataset of 2,473 pairs, consisting of 1,612 positive pairs and 861 negative pairs. In the experiments, 30 percent of the data are used for parameter tuning (called the development set) and the rest for testing.

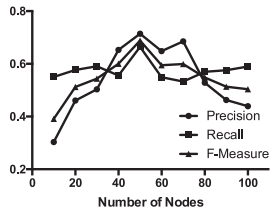
6.2.2 Parameter Analysis of Model One

We first explore the performance of *is-a* relation extraction using Model One as the underlying projection models. In this approach, two parameters are required to be tuned, i.e., β and θ. We report the parameter analysis results over the development set. We vary the value of β from 0.1 to 0.9. With a fixed value of β, we change the value of θ to achieve the best performance over the development set. Fig. 6a illustrates the maximum F-measure when β varies. When β is close to 1, the final prediction score $f(e, c)$ largely depends on the projection-based model prediction $s(e, c)$, ignoring the effect of the collective prediction module. When β is close to 0, the final prediction is biased towards collective prediction. Experimental results show that our method is generally not very sensitive to the selection of β. When $\beta = 0.7$, it has the highest performance, indicating a good balance between the local and global prediction scores. Additionally, Fig. 6b illustrates the precision-recall curve with respect to the change of θ ($0 < \theta < 1$) when $\beta = 0.7$. The highest F-measure is achieved when we set $\theta = 0.05$.

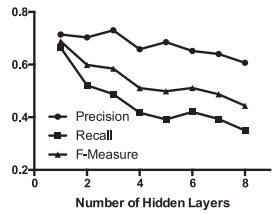
6.2.3 Network Structure Analysis of Model Two

In this part, we study how different neural network architectures affect the performance of *is-a* relation extraction. For simplicity, we apply the same architecture for both positive and negative projection learning. We report the scores in Fig. 7 using the model only, without considering the collective prediction and hypernym expansion step, in order to achieve a better understanding of the model.

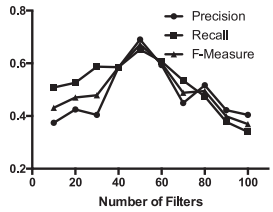
We first implement a neural network with one fully connected layer as the hidden layer, using the Rectified Linear



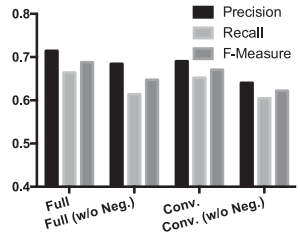
(a) Performance change with one hidden layer of different numbers of hidden nodes



(b) Performance change with different numbers of hidden layers layers



(c) Performance change with one convolutional hidden layer with different numbers of filters



(d) Performance comparison of Model Two with and without the negative sampling technique

Fig. 7. Network structure analysis of model two over development set.

Unit (ReLU) as the activation function for the hidden layer and the linear function for the output layer. The hyper-parameter settings are $\mu_1 = \mu_2 = 0.01$, and $\eta = 0.5$. For each pair in the training data, we sample 10 negative samples if they exists. Fig. 7a illustrates the change of performance with the different numbers of hidden units. It shows the F-measure peaks when we use 50 units. We continue to add more fully connected hidden layers, each having 50 ReLUs, with the performance shown in Fig. 7b. The F-measure drops from 68.6 to 44.3 percent when the number of hidden layers is set from 1 to 10. It means that using deeper neural networks does not improve the accuracy of *is-a* relation prediction. In contrast, it causes the model to have a large tendency to overfit. Next, we replace the fully connected layer with the convolutional layer. Fig. 8 gives a simple example of the neural network with one convolutional layer. In the experiments, we leverage the 1D convolutional layer with filter size 20, stride size 1 and the narrow convolution technique. Fig. 7c shows how the performance changes when we use 10 to 100 filters. In general, the trend of the F-measure is similar to an inverted "V" shape, indicating the model goes through the status of underfitting, well-fitting and overfitting. We also try removing the regularization subnetwork to check whether the negative sampling technique improves the performance. Fig. 7d shows that by using negative sampling, the performance is increased by 4.1 and 4.8 percent in F-measure, when we use the fully connected layer and the convolutional layer.

In summary, this part of the study reveals that i) using the fully connected layer and the convolutional layer have a similar effect on *is-a* relation prediction; ii) using deeper neural networks can not improve the performance and is likely to cause overfitting, and iii) using the proposed negative sampling technique can improve the performance.

6.2.4 Comparative Study

We set up the following strong baselines to compare our method with state-of-the-art approaches. The experimental results are shown in Table 4. The results are reported in the

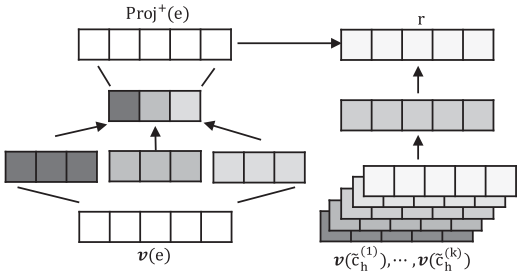
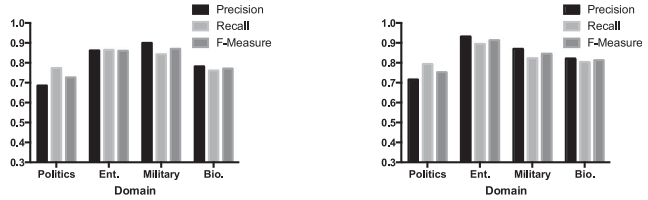


Fig. 8. The neural network architecture using one convolution layer as the hidden layer.



(a) Our method with Model 1 (b) Our method with Model 2

Fig. 9. Performance of the proposed method across different domains.

weighted average of the two classes. To represent entity-category pairs with word embedding based features, we implement several state-of-the-art methods: the *concat* (concatenation) model $\mathbf{v}(e) \oplus \mathbf{v}(c_h)$, the *sum* model $\mathbf{v}(e) + \mathbf{v}(c_h)$ and the *diff* (difference) model $\mathbf{v}(e) - \mathbf{v}(c_h)$ [36], [37]. l_2-regularized logistic regression is trained to make the prediction due to the high performance in previous research. This approach achieves the highest F-measure of 73.8 percent. We also test the *piecewise projection* model proposed in [22] over the Chinese Wikipedia. It has a slight improvement in performance of 1.7 percent in F-measure. The transductive learning approach [15] outperforms previous deep learning approaches in terms of F-measure. As seen, our method with Model One (i.e., "Our Method (w. Model 1)" in Table 4) increases the F-measure by 11.8 percent (with $p < 0.01$) compared to [22]. The implementation of our method with Model Two (using one convolutional layer) has the F-measure of 88.9 percent. Our method beats the state-of-the-art approach [15] due to two reasons: (i) the negative sampling based neural projection makes it easier to distinguish is-a and not-is-a relations; (ii) the collective inference technique is specifically designed for Wikipedia data, further improving the performance.

6.2.5 Domain and Error Analysis

We further analyze the performance of our approach in different domains and summarize errors that frequently occur. We take four domains as examples (i.e., politics, entertainment, military and biomedicine). For each domain, we sample 100 pairs from the test set that are related to the domain and report the performance in Fig. 9. As seen, the proposed method is generally robust across different domains. However, we observe that it performs better in "closer" domains than in "more open" domains. For example, concepts in the politics domain (e.g., political groups and organizations, politicians, campaigns, policies, ideologies, etc.) tend to be large in numbers and diverse in semantics than those in the entertainment domain (e.g., actors, directors, movies, etc.). As a consequence, the semantics of *is-a* relations are more difficult for the models to learn.

The illy-represented entities (IRE) and the projection error propagation (PEP) are the two types of errors occurring in the prediction. IRE accounts for a majority of the erroneous cases (71 percent). It refers to the situation where entities rarely appear in the corpus, causing the embeddings of such entities to be inaccurate. Hence, the effectiveness of projection learning is likely to suffer, making it difficult for the model to distinguish *is-a* and *not-is-a* relations. We also find that most errors of this type occur when the proposed method treats *topic-of* relations as *is-a* relations and *is-a* relations as *topic-of* relations. PEP is related to the collective prediction refinement. Although this mechanism improves the performance in general, there is a possibility of error propagation in the learning process. Cases of prediction errors of different domains and types are presented in Table 3.

6.2.6 Overall Results

We use the improved neural model together with the subsequent steps (including collective refinement, hypernym expansion and a few indirect mapping rules) to extract all the *is-a* relations from all Chinese Wikipedia categories. In total, we harvest 1.48M *is-a* relations from Chinese Wikipedia categories, consisting 563 K entities and 153 K distinct categories.[2] In Fig. 10a, we present how many entities have a particular number of hypernyms (i.e., taxonomic categories). In average, each entity has 2.64 hypernyms. We can see that this distribution fits in a semi-log line, defined by a log scale on the y-axis and a linear scale on the x-axis. Similarly, each hypernym has 9.66 entities in average, with the distribution illustrated in Fig. 10b. The number of entities per hypernym follows the power-law distribution with a long tail, defined by log scales on both x and y-axes.

6.3 Evaluation of *Non-Taxonomic* Relation Extraction

6.3.1 Detailed Steps

We first run the single-pass pattern miner to extract category patterns. We find that patterns associated with fewer than 20 category entities have little semantic generalization ability. The distribution is plotted in Fig. 11a. Numbers out of the scope (10, 100] are omitted in the figure due to extreme small or large values. We select patterns with top-500 highest support values to be used in the next step. This is because most of these patterns are associated with more than 20 category entities. In Table 5, we present category patterns with extreme support values. We can see that high-support patterns usually have very clear semantics, indicating the existence of a relation. In contrast, the semantics low-support patterns tend to be blurry.

For each of these selected patterns, we set $\tau = 0.7$ based on human inspection and run the MEWCP algorithm three times to ensure the high reliability of the seed relation instances. The distribution of confidence values of these patterns is illustrated in Fig. 11b. Readers can also refer to category patterns with extreme confidence values in Table 5. We select top-250 most confident patterns with confidence scores larger than 0.9 for the next step of relation extraction. To determine the value of γ, we carry out a preliminary

2. The statistics include 0.12M *is-a* relations extracted from the indirect mapping modules in Section 5.3.

TABLE 3
Cases of Error Predictions in Different Domains and Types

Domain	Entity	Category	Predicted	Truth	Error Type
Politics	巴尔干半岛 (Balkan Peninsula)	地缘政治 (geopolitics)	1	0	IRE
	绥靖主义 (appeasement)	政治术语 (political term)	0	1	PEP
Entertainment	养鸭人家 (Beautiful Duckling)	台湾剧情片 (story film in Taiwan)	0	1	IRE
	孙越 (演员) (Sun Yueh (actor))	台北市荣誉市民 (Honorary citizen of the city of Taipei)	0	1	PEP
Military	地头 (Jitō, medieval land stewards)	日本军事史 (military history of Japan)	1	0	IRE
	游击战 (guerrilla)	战术 (tactics)	0	1	PEP
Biomedicine	银环蛇 (Bungarus multicintus)	中国爬行动物 (reptile in China)	0	1	IRE
	细胞核 (nucleus)	细胞膜 (cell membrane)	1	0	PEP

"1" and "0" refer to is-a *and* not-is-a *relation labels in columns "Predicted" and "Truth".*

experiment, which samples 200 entity pairs to estimate the accuracy. It shows that even we set γ to a relatively low value (i.e., 0.2), the accuracy is over 90 percent. Finally, 26 relation predicates are created automatically based on direct verb mapping, such as "建立 (established-in)", etc. We design the mapping rules and relation predicates for the remaining 16 relation types manually, with examples shown in Table 6.

6.3.2 Accuracy and Coverage of Extracted Relations

To evaluate the correctness of extracted relations, we carry out two experimental tests: accuracy test and coverage test. Following [6], in the accuracy test, we randomly sample 200 relation instances for each relation type and ask human annotators to determine their correctness. We discard the results if human annotators disagree. The coverage test is to determine whether the extracted relations already exist in Chinese knowledge bases. Low coverage score means these relations are not present in existing Chinese knowledge bases and thus are novel and worth extracting. In the experiments, we take CN-DBpedia V2.0 [27] as the ground truth knowledge base. Up till February 2017, it contains 41M explicit semantic relations of 9M entities. We use the CN-DBpedia API[3] to obtain relations for each entity and report the coverage of relation r as

$$cov(r) = \frac{\#\text{Matched extractions in CN-DBpedia}}{\#\text{Correct extractions generated by our approach}}.$$

To make a fair comparison, because relations in different knowledge base systems may express differently, we ask human annotators to determine whether the relations extracted by our approach and CN-DBpedia match or not. In Table 7, we present the size, accuracy and coverage values of eight *non-taxonomic* relations, each with over three thousand relation instances.

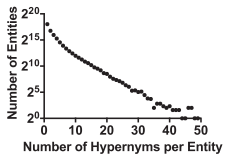
From the experimental results, we can see that the accuracy is over 90 percent for all the eight relations. Especially the accuracy values of some relations are over 98 percent or even equal to 100 percent. This means it is reliable to extract relations from Chinese UGCs based category pattern mining. The results of the coverage tests present a large variance among different relations. While some relations such as "出生 (born-in)" have a relatively high coverage in CN-DBpedia, other relation instances that we extract are rarely present in the knowledge base. Overall, the average coverage is approximately 21.1 percent. This means although the Chinese knowledge base is relatively large in size, it is far from complete. Furthermore, most relations in Chinese knowledge bases are extracted from infoboxes, in the form of attribute-value pairs [38], [39]. Thus, the knowledge harvested from UGCs can be an important supplementary for these systems. Currently, we only focus on Chinese Wikipedia categories. We will study how to extend our approach to UGCs for other knowledge sources, especially domain-specific sources in the future.

6.3.3 Comparison with Existing Systems and Baselines

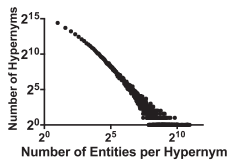
Harvesting *non-taxonomic* relations from UGCs is a non-trivial task with no standard evaluation frameworks available. As discussed in the related work, OIE methods can

TABLE 4
Performance Comparison of Different *is-a* Relation Extraction Methods Over the Test Set

Method	Precision	Recall	F-Measure
Concat Model	78.3%	66.3%	71.8%
Sum Model	76.7%	71.2%	73.8%
Diff Model	76.4%	67.5%	71.6%
Piecewise Projection	77.6%	73.6%	75.5%
Transductive Learning	78.2%	76.5%	77.3%
Our Method (w. Model 1)	**88.4%**	**86.3%**	**87.3%**
Our Method (w. Model 2)	**89.3%**	**88.6%**	**88.9%**

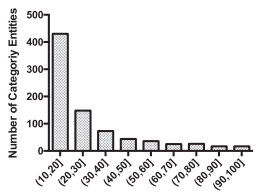


(a) Distribution of number of hypernyms per entity

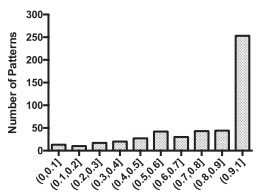


(b) Distribution of number of entities per hypernym

Fig. 10. Distributional analysis on extracted *is-a* relations.



(a) Distribution of number of category entities per pattern



(b) Distribution of confidence of high-support patterns

Fig. 11. Analysis of support and confidence of category patterns.

3. http://knowledgeworks.cn:20313/cndbpedia/api/entityAVP

TABLE 5
Examples of Category Patterns with High and Low Support/Confidence Values

Type	Category Pattern	Score
High support	[E]校友 (Alumni of [E])	2,316
	[E]出生 ([E] births)	1,253
Low support	[E]地区 ([E] region)	14
	国际[E] (International [E])	12
High confidence	[E]州城市 (City in state [E])	0.99
	[E]副校长 (Vice president of [E])	0.98
Low confidence	[E]地理 ([E] geography)	0.10
	[E]事故 ([E] accident)	0.06

Score refers to "support" for high/low support patterns and "confidence" for high/low confidence patterns.

TABLE 6
Examples of Manually Defined Relation Mappings

Category Pattern	Relation Predicate
[E]校友 (Alumni of [E])	毕业 (graduated-from)
[E]队教练 (Coach of [E])	执教 (coach-team)
[E]省市镇 (City/Town in Province [E])	位于 (located-in)
[E]获得者 (Winner of [E])	获奖 (win-prize)

TABLE 7
Size, Accuracy and Coverage Values of Eight Extracted Relation

Relation Type	Size	Accuracy	Coverage
毕业 (graduated-from)	44,118	98.0%	22.9%
位于 (located-in)	29,460	97.2%	8.5%
建立 (established-in)	20,154	95.0%	31.5%
出生 (born-in)	11,671	98.3%	41.4%
成员 (member-of)	8,445	96.0%	4.2%
启用 (open-in)	8,956	98.2%	21.6%
逝世 (died-in)	5,597	100.0%	18.4%
得奖 (win-prize)	3,262	90.0%	27.3%

not be regarded as baselines of our work. Furthermore, the significant difference between English and Chinese makes it difficult to compare our method with similar research. The work [7] focuses on modifiers in categories and is not directly comparable to ours. In YAGO [6], relations in categories are extracted by handcrafting regular expressions. They extract nine *non-taxonomic* relations, with accuracy values of around 90-98 percent. Our approach avoids the manual work to a large extent and harvests more types of relations with a comparable accuracy.

Next, we compare our work with the system [5], which heavily relies on prepositions in patterns to discover relations. In Chinese, prepositions are usually expressed implicitly and hence these patterns are not directly applicable. We implement a variant for Chinese (denoted as CN-WikiRe). The patterns that we used in CN-WikiRe are shown in Table 9. In the experiments, we extract 165,048 *non-taxonomic* relation instances using CN-WikiRe, containing 631 relation types. Although the number of relation types may seem large at the first glance, only 14 percent of them are actual relation predicates, with the rest being either incorrect or uninformative. The reasons are twofold: i) word segmentation and POS tagging for Chinese short texts still suffer from low accuracy and ii) not all verbs extracted by CN-WikiRe can

TABLE 8
Comparison of Different Methods for *Non-Taxonomic* Relation Extraction

Method	Estimated Accuracy
CN-WikiRe	58.6%
Our Method (w/o Conf.)	74.4%
Our Method (w/o Filter)	94.2%
Our Method	**97.4%**

TABLE 9
Three Types of Category Patterns That We Design for CN-WikiRe

Type	Category Pattern (with Example)
Member pattern	[E]成员/总统 (Member/President of [E]) (中国科学院成员 (Member of Chinese Academy of Sciences))
Verb-NP pattern	[E]+Verb+(的)+Noun Phrase (1990年建立的组织 (Organization founded in 1990))
Verb pattern	[E]+Verb (1980年出生 (1980 births))

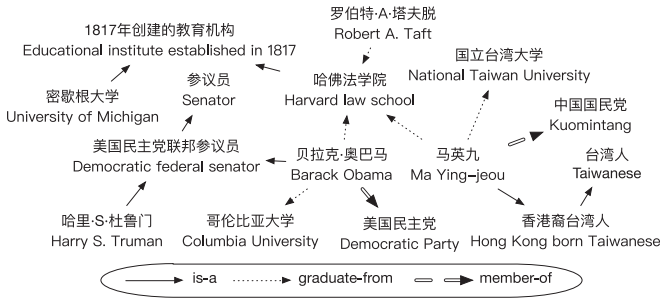


Fig. 12. A clip of the knowledge graph in the politics domain extracted from Chinese Wikipedia UGCs.

serve as relation predicates (e.g., "传导(transmit)", "缩小(shrink)"). We sample 500 relations from the collection where the extracted verbs are labeled as real relation predicates. The accuracy is 58.6 percent, much lower than our method. Furthermore, the partially explicit and implicit patterns (see [5]) do not have their counterparts in Chinese. Therefore, our method is superior to existing systems.

We also implement two variants of the proposed method: "Our Method (w/o Conf.)" and "Our Method (w/o Filter)". "Our Method (w/o Conf.)" and "Our Method (w/o Filter)" refer to the slight modification of the proposed method without the confidence-based pattern selection step and the relation filtering step, respectively. We estimate the accuracy of extracted relations by the two variants using the same approach as CN-WikiRe. The results are illustrated in Table 8. It shows that the proposed method outperforms the two baselines at the accuracy of 23.0 and 3.2 percent.

6.4 Case Study and Released Resources

Fig. 12 illustrates a clip of the political knowledge graph constructed by the proposed approach. It includes multiple types of entities, such as people, universities, parties, etc. We can see that by using relations extracted by Wikipedia UGCs only, we are able to create a dense knowledge graph. The knowledge is also an important, complementary data source for existing Chinese knowledge bases (e.g., [38], [39]).

We have released all the extracted relations from Wikipedia UGCs to public in the form of $<$subject, predicate, object$>$ triples. The relations can be downloaded from https://chywang.github.io/data/tkde.zip.

7 Conclusion and Perspectives

In this paper, we propose a weakly supervised framework to extract fine-grained relations from Chinese UGCs. For *is-a* relations, we introduce two word embedding based projection models and refine prediction results using collective inference. To extract *non-taxonomic* relations, we design a graph mining technique to harvest relation types and category patterns with minimal human supervision. In summary, our approach extracts 1.84M relations, including 1.48M *is-a* relations and 0.36M others. The accuracy values of *is-a*, other and all relations are 93.8, 97.4 and 94.5 percent respectively, estimated over random samples of 500 relations. Comparative studies are also conducted to show that our approach outperforms previous methods for both *is-a* and *non-taxonomic* relation extraction. The extracted relations can be of help for Chinese knowledge base completion.

However, we admit that although we have achieved some success in UGC relation extraction, understanding Chinese UGCs for machines still faces challenges. The key barriers lie in two aspects: i) the lack of (relatively) fixed syntactic/lexical expressions in Chinese and ii) the difficulty in interpreting Chinese noun phrase for machines. As a consequence, we have to make some compromises in algorithm design in order to achieve high accuracy. For example, we have to abandon the iterative pattern learning mechanism during the *non-taxonomic* relation extraction process. Additionally, Wikipedia UGCs are more regular and fixed in pattern expressions and of higher quality than categories in other data sources. We believe that there is still no effective solution (including ours) for decoding very short and noisy Chinese UGCs. A further clarification is that the *is-a* relations extracted by our work are mostly *instance-of* relations. These relations can be of help to populate existing Chinese semantic resources, such as BigCilin,[4] SinicaBOW,[5] by linking *instance-of* relations to existing taxonomies or semantic hierarchies.

In the future, our work can be extended by addressing the following issues: i) improving our work for short text knowledge extraction by mapping lexical patterns to more general and conceptual patterns and integrating the iterative pattern learning technique into the model, ii) designing a cross-lingual approach to transfer models for English UGC relation extraction to Chinese, iii) learning more high-level hypernyms from existing taxonomic categories based on accurate Chinese noun phrase interpretation, iv) investigating whether linguistic theories (such as speech acts theory) can benefit short text understanding, and v) linking the extracted relations to existing Chinese semantic resources.

Acknowledgments

This work is supported by the National Key Research and Development Program of China under Grant No. 2016YFB1000904. Chengyu Wang would also like to thank the ECNU Outstanding Doctoral Dissertation Cultivation Plan of Action (No. YB2016040) for the support of his research. This work is the extension of our previous work published in a conference paper of EMNLP 2017 [35].

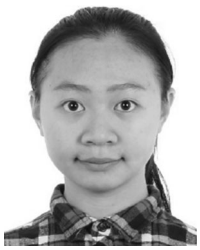
4. http://www.bigcilin.com
5. http://bow.ling.sinica.edu.tw

References

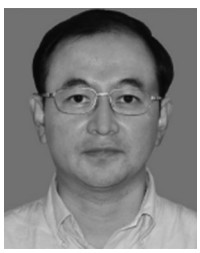
[1] B. Xu, C. Xie, Y. Zhang, Y. Xiao, H. Wang, and W. Wang, "Learning defining features for categories," in *Proc. Int. Joint Conf. Artif. Intell.*, 2016, pp. 3924–3930.
[2] T. Flati, D. Vannella, T. Pasini, and R. Navigli, "Two is bigger (and better) than one: The Wikipedia bitaxonomy project," in *Proc. Conf. Appl. Natural Language Process.*, 2014, pp. 945–955.
[3] S. P. Ponzetto and M. Strube, "Deriving a large-scale taxonomy from Wikipedia," in *Proc. AAAI Conf. Artif. Intell.*, 2007, pp. 1440–1445.
[4] S. P. Ponzetto and R. Navigli, "Large-scale taxonomy mapping for restructuring and integrating Wikipedia," in *Proc. Int. Joint Conf. Artif. Intell.*, 2009, pp. 2083–2088.
[5] V. Nastase and M. Strube, "Decoding Wikipedia categories for knowledge acquisition," in *Proc. AAAI Conf. Artif. Intell.*, 2008, pp. 1219–1224.
[6] F. M. Suchanek, G. Kasneci, and G. Weikum, "YAGO: A core of semantic knowledge," in *Proc. Int. Conf. World Wide Web*, 2007, pp. 697–706.
[7] M. Pasca, "German typographers versus German grammar: Decomposition of Wikipedia category labels into attribute-value pairs," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2017, pp. 315–324.
[8] C. Wang, M. Gao, X. He, and R. Zhang, "Challenges in chinese knowledge graph construction," in *Proc. IEEE Int. Conf. Data Eng. Workshops*, 2015, pp. 59–61.
[9] L. Qiu and Y. Zhang, "ZORE: A syntax-based system for chinese open relation extraction," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2014, pp. 1870–1880.
[10] Y. Chen, Q. Zheng, and W. Zhang, "Omni-word feature and soft constraint for Chinese relation extraction," in *Proc. Conf. Appl. Natural Language Process.*, 2014, pp. 572–581.
[11] J. Li, C. Wang, X. He, R. Zhang, and M. Gao, "User generated content oriented Chinese taxonomy construction," in *Proc. Asia-Pacific Web Conf.*, 2015, pp. 623–634.
[12] Z. Wang, J. Li, S. Li, M. Li, J. Tang, K. Zhang, and K. Zhang, "Cross-lingual knowledge validation based taxonomy derivation from heterogeneous online Wikis," in *Proc. AAAI Conf. Artif. Intell.*, 2014, pp. 180–186.
[13] W. Wu, H. Li, H. Wang, and K. Q. Zhu, "Probase: A probabilistic taxonomy for text understanding," in *Proc. Int. Conf. Manage. Data*, 2012, pp. 481–492.
[14] R. Fu, J. Guo, B. Qin, W. Che, H. Wang, and T. Liu, "Learning semantic hierarchies via word embeddings," in *Proc. Conf. Appl. Natural Language Process.*, 2014, pp. 1199–1209.
[15] C. Wang, J. Yan, A. Zhou, and X. He, "Transductive non-linear learning for Chinese hypernym prediction," in *Proc. Conf. Appl. Natural Language Process.*, 2017, pp. 1394-1404.
[16] F. Mahdisoltani, J. Biega, and F. M. Suchanek, "YAGO3: A knowledge base from multilingual Wikipedias," in *Proc. Conf. Innovative Database Res.*, 2015.
[17] D. Alfarone and J. Davis, "Unsupervised learning of an IS-A taxonomy from a limited domain-specific corpus," in *Proc. Int. Joint Conf. Artif. Intell.*, 2015, pp. 1434–1441.
[18] A. Gupta, F. Piccinno, M. Kozhevnikov, M. Pasca, and D. Pighin, "Revisiting taxonomy induction over Wikipedia," in *Proc. Int. Conf. Comput. Linguistics*, 2016, pp. 2300–2309.
[19] C. Wang, X. He, and A. Zhou, "A short survey on taxonomy learning from text corpora: Issues, resources and recent advances," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2017, pp. 1190–1203.
[20] R. Fu, B. Qin, and T. Liu, "Exploiting multiple sources for open-domain hypernym discovery," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2013, pp. 1224–1234.
[21] T. Wu, G. Qi, H. Wang, K. Xu, and X. Cui, "Cross-lingual taxonomy alignment with Bilingual Biterm topic model," in *Proc. AAAI Conf. Artif. Intell.*, 2016, pp. 287–293.
[22] C. Wang and X. He, "Chinese hypernym-hyponym extraction from user generated categories," in *Proc. Int. Conf. Comput. Linguistics*, 2016, pp. 1350–1361.
[23] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *Proc. Int. Conf. Comput. Linguistics*, 1992, pp. 539–545.

[24] J. Yamane, T. Takatani, H. Yamada, M. Miwa, and Y. Sasaki, "Distributional hypernym generation by jointly learning clusters and projections," in *Proc. Int. Conf. Comput. Linguistics*, 2016, pp. 1871–1879.
[25] N. Surtani and S. Paul, "A VSM-based statistical model for the semantic relation interpretation of noun-modifier pairs," in *Proc. Int. Conf. Recent Advances Natural Language Process.*, 2015, pp. 636–645.
[26] E. Choi, T. Kwiatkowski, and L. Zettlemoyer, "Scalable semantic parsing with partial ontologies," in *Proc. Conf. Appl. Natural Language Process.*, 2015, pp. 1311–1320.
[27] B. Xu, Y. Xu, J. Liang, C. Xie, B. Liang, W. Cui, and Y. Xiao, "CN-DBpedia: A never-ending Chinese knowledge extraction system," in *Proc. Int. Conf. Ind. Eng. Other Appl. Appl. Intell. Syst.*, 2017, pp. 428–438.
[28] O. Etzioni, A. Fader, J. Christensen, S. Soderland, and Mausam, "Open information extraction: The second generation," in *Proc. Int. Joint Conf. Artif. Intell.*, 2011, pp. 3–10.
[29] O. Levy, S. Remus, C. Biemann, and I. Dagan, "Do supervised distributional methods really learn lexical inference relations?" in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Language Technol.*, 2015, pp. 970–976.
[30] C. Biemann, D. Ustalov, A. Panchenko, and N. Arefyev, "Negative sampling improves hypernymy extraction based on projection learning," in *Proc. Meet. Eur. Chapter Assoc. Comput. Linguistics*, 2017, pp. 543–550.
[31] E. Pavlick and M. Pasca, "Identifying 1950s American Jazz musicians: Fine-grained IsA extraction via modifier composition," in *Proc. Conf. Appl. Natural Language Process.*, 2017, pp. 2099–2109.
[32] X. Qiu, Q. Zhang, and X. Huang, "FudanNLP: A toolkit for chinese natural language processing," in *Proc. Conf. Appl. Natural Language Process.*, 2013, pp. 49–54.
[33] B. Alidaee, F. Glover, G. A. Kochenberger, and H. Wang, "Solving the maximum edge weight clique problem via unconstrained quadratic programming," *Eur. J. Oper. Res.*, vol. 181, no. 2, pp. 592–597, 2007.
[34] A. Carlson, J. Betteridge, R. C. Wang, E. R. H. Jr., and T. M. Mitchell, "Coupled semi-supervised learning for information extraction," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2010, pp. 101–110.
[35] C. Wang, Y. Fan, X. He, and A. Zhou, "Learning fine-grained relations from Chinese user generated categories," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2017, pp. 2577-2587.
[36] S. Roller, K. Erk, and G. Boleda, "Inclusive yet selective: Supervised distributional hypernymy detection," in *Proc. Int. Conf. Comput. Linguistics*, 2014, pp. 1025–1036.
[37] P. Mirza and S. Tonelli, "On the contribution of word embeddings to temporal relation classification," in *Proc. Int. Conf. Comput. Linguistics*, 2016, pp. 2818–2828.
[38] Z. Fang, H. Wang, J. Gracia, J. Bosque-Gil, and T. Ruan, "Zhishi. lemon: On publishing Zhishi.me as linguistic linked open data," in *Proc. Int. Semantic Web Conf.*, 2016, pp. 47–55.
[39] Z. Wang, J. Li, Z. Wang, S. Li, M. Li, D. Zhang, Y. Shi, Y. Liu, P. Zhang, and J. Tang, "XLore: A large-scale English-Chinese bilingual knowledge graph," in *Proc. Int. Semantic Web Conf.*, 2013, pp. 121–124.

Chengyu Wang received the BE degree in software engineering from East China Normal University (ECNU), in 2015. He is working toward the PhD degree in the School of Computer Science and Software Engineering, East China Normal University, China. 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 received the BE degree in software engineering from East China Normal University (ECNU), in 2016. She is working toward the master's degree in the School of Computer Science and Software Engineering, East China Normal University, China. Her research interests include relation extraction and natural language processing for large-scale knowledge graphs.



Xiaofeng He received the PhD degree from Pennsylvania State University. He is a professor of computer science with the School of Computer Science and Software Engineering, East China Normal University, China. His research interests include machine learning, data mining, and information retrieval. Prior to joining ECNU, he worked with Microsoft, Yahoo Labs, and Lawrence Berkeley National Laboratory. He is a member of the IEEE.

Aoying Zhou is a professor with East China Normal University. He is now acting as a vice-director of ACM SIGMOD China and the Database Technology Committee of the China Computer Federation. He is serving as a member of the editorial boards of the *VLDB Journal*, the *WWW Journal*, etc. His research interests include data management for data-intensive computing, and memory cluster computing. He is a member of the IEEE.