IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 24, 2021, accepted February 20, 2021, date of publication March 2, 2021, date of current version March 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3063234

Neural Network-Based Tree Translation for Knowledge Base Construction

HAIJUN ZHANG
Department of Computer Science, Shanghai Jiao Tong University, Shanghai 200240, China
e-mail: haijun@sjtu.edu.cn

ABSTRACT Knowledge bases (KB), such as Probase and ConceptNet, play an important role in many natural language processing tasks. Compared with resource-poor languages such as Chinese, the scale and quality of English knowledge bases are obviously superior. To expand Chinese KBs by using English KB resources, translating English KBs into Chinese is an effective way. In this direction, two major challenges are how to model more structure semantics to improve translation quality and how to avoid labor-intensive feature engineering. We address these challenges by presenting a neural network approach, which learns tree representation by different structure features. We also build a new dataset for English-Chinese KB translation from Probase and ConceptNet, and compare our proposed approach with several baselines on it. Experimental results show that the proposed method improves the translation accuracy compared with baseline methods. Meanwhile, we translate Probase and ConceptNet into Zh-Probase and Zh-ConceptNet by our proposed model, and release them to the public, in hope of speeding up the research in Chinese natural language processing tasks.

INDEX TERMS Knowledge base construction, machine translation, named entity disambiguation, natural language processing, representation learning, word sense disambiguation, neural networks.

I. INTRODUCTION

Knowledge bases like ConceptNet [1] and Probase [2] have always been playing the central role in artificial intelligence. ConceptNet, a well-known commonsense knowledge base, has been successfully leveraged in many applications, such as commonsense reasoning [1], query expansion [3], word embedding improving [4], and sentiment analysis [5]. Probase, a well-known taxonomic knowledge base, has been successfully leveraged in many applications, such as taxonomy keyword search [6], semantic web search [7], short text understanding [8], and understanding web tables [7]. However, most of these important knowledge bases are in English, not in other languages, such as Chinese. For example, in ConceptNet, there are 2.9 million edges containing both English concepts while only 0.4 million edges containing both Chinese concepts.[1] Besides, Probase is an English-only knowledge base consisting of 13,949,064 "IsA" pairs. Therefore, alleviating this language resource imbalance is valuable and urgent for non-English researchers community, like Chinese, Spanish or Hindi, especially for a large amount of Chinese researchers.

One way to deal with the resource imbalance problem is to acquire these resources from scratch, but it faces the following problems: first, it takes great human effort to build an equivalent number of Chinese ConceptNet, which is a cost of money and time that most researchers cannot afford. Second, although some Chinese taxonomic knowledge bases, such as CN-Probase [9] and zhishi.me [10], have been built, they still suffer two serious problems: first, because their data sources come from online encyclopedias, their concepts are not as numerous and broad as those in Probase, shown in Table 4, which is the basis for some explicit topic model applications [8]. Second, they have no probabilistic characteristic, which is crucial in some applications [8], [11].

Another straightforward way to deal with the resource imbalance problem is through translation. We can use the out-of-the-box translator to directly translate these knowledge bases, but this method has the problem of word sense disambiguation and named entity disambiguation. For example, the word "plant" can mean "工厂 (a factory)", "植物 (a living thing that grows in the earth like flowers)", "设备 (machine or device)", and "栽赃物品 (something planted

The associate editor coordinating the review of this manuscript and approving it for publication was Felix Albu.

[1]These statistics come from ConceptNet 5.6.

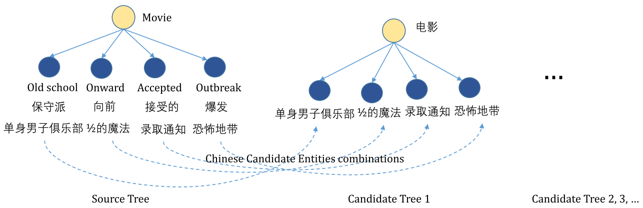

FIGURE 1. A data example existing in Probase.

secretly for discovery by another)". The word "apple" can mean "苹果 (a kind of fruit)" and "苹果公司 (the phone company)". The triple (apple, IsA, plant) in Probase can totally have 8 different Chinese translation results because of the word sense ambiguity and named entity ambiguity. Ambiguity problem is common in knowledge bases, such as Freebase [12], where more than half of translations (from English to Chinese) are ambiguous, shown in the preliminary statistical analysis [13]. To handle the disambiguation problem, an adaptive neural network is adopted to translate English knowledge bases into Chinese, which maps both English triples and Chinese triples in the same semantic space and chooses the nearest Chinese triple as the translation result for each English triple [13]. However, this neural network can only capture the semantic of the triple structure rather than some more complex structures, such as trees or graphs. Meanwhile, we need to avoid doing labor-intensive work, such as feature engineering, to get Chinese knowledge bases.

Therefore, we propose a neural network that can capture the semantic of tree structure and translate the entire tree structure at once. Specifically, we cluster the triples in the knowledge base into trees, enumerate all the candidate trees, as shown in Fig. 1, then use the designed neural network to score each candidate tree, and get the tree with the highest score as the final translation result. The neural network integrates the features of mention, context, coherence, and relation to score each candidate tree.

We evaluate the effectiveness of our method on a manually created dataset. Empirical results show that the proposed method consistently outperforms baseline methods and the state-of-the-art method. We also show the effectiveness of the combined features in our model. Based on this neural network, we translate Probase and ConceptNet to Zh-Probase and Zh-ConceptNet, respectively, and show their coverage and accuracy are both satisfactory. The main contributions of this work are as follows:

- We are the first to model knowledge base translation as scoring on source and candidate trees and we present a novel neural model that effectively captures the tree structure semantic between source tree and candidate tree during translation.
- We have done a lot of experiments to study the influence of different settings on the acquisition of structural tree information by the neural network, such as the number of child nodes, various feature combination methods and final feature fusion methods.
- We build a dataset to evaluate the translation and present the superior performance of our method over the state-of-the-art method, improving accuracy by 3.1%.
- Furthermore, we use the designed neural network to translate Probase and ConceptNet into Zh-Probase[2] and Zh-ConceptNet,[3] and their coverage and accuracy are both satisfactory. Zh-ConceptNet is the first large Chinese commonsense knowledge base with 2 million triples.

In the next three sections, we list some highlights and present and formulate the focal problem, then present the design of our proposed framework and experimental results, discuss related work, and conclude with a discussion and future directions.

II. PROBLEM FORMULATION

Most knowledge bases, such as Probase and ConceptNet, are composed of triples, each of which is a fact consisting of two arguments and one relation, and these triples can be clustered into trees, where all triples inside share one argument. Here, we only consider the tree of depth 1. We denote the tree structure by $X = (Arg_1, Arg_{21}, \ldots, Arg_{2n}, rel)$, where rel is the relation between the root node Arg_1 and these child nodes Arg_{2i}, $i = 1, 2, \ldots, n$. Each node is a word or short text written in language L_1(e.g. English). Our task is to find the corresponding tree $E = (Arg'_1, Arg'_{21}, \ldots, Arg'_{2n}, rel)$ in language L_2(e.g. Chinese) so that each item in E correctly disambiguates the surface form of each item of X.

The result of this task depends on the quality of word sense and named entity disambiguation. Traditional named entity disambiguation or word sense disambiguation is usually formulated by defining a scoring function $S(x, e)$, which indicates how relevant between surface form mention x and target word sense or entity e. Such techniques consider each item independently, but, in our case, they ignore the interactions between siblings and the relation in the tree structure. To capture more structural information in the tree, we define another scoring function for this task as follow:

$$\hat{E} = \underset{E \in GEN(X)}{\operatorname{argmax}} \; S(X, E), \tag{1}$$

where $GEN(X)$ denotes the set of all candidate trees, and the function measures the overall correlation score between the whole source tree and the candidate tree.

III. APPROACH

In this section, we describe our framework for knowledge base translation, as shown in Fig. 2. Specifically, we first generate the Chinese word senses and entities for each argument in these triples using existing translators and the heuristic extraction result from Wikipedia. Next, we cluster the triples in the knowledge base to build source trees and construct the candidate trees for each source tree. Then we use the

[2]https://dx.doi.org/10.21227/0wq5-6v48
[3]https://dx.doi.org/10.21227/mbz1-xj52

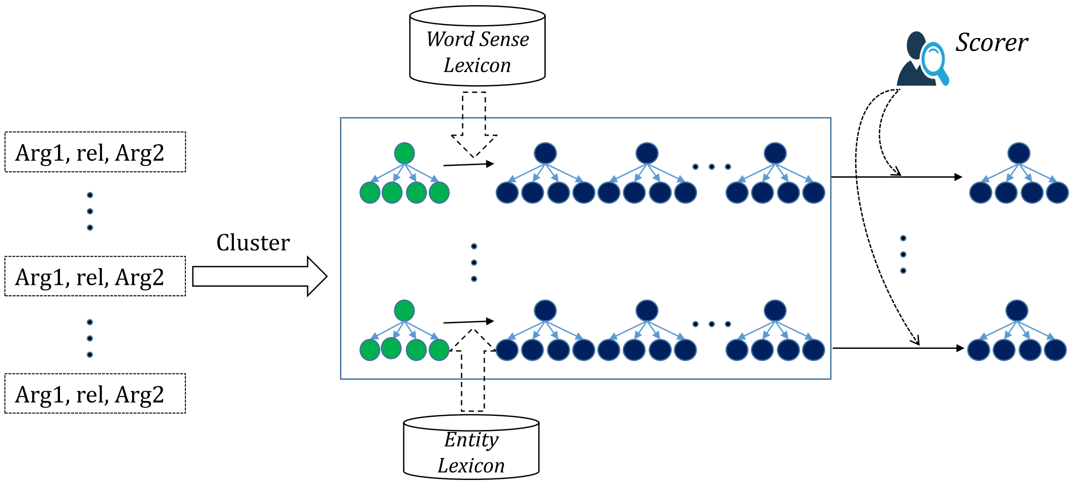

FIGURE 2. Overview of our proposed framework.

proposed neural network to score each candidate tree, and the candidate tree with the highest score will be chosen as the final translation result.

A. CANDIDATE LEXICON GENERATION

We first try to get all Chinese word senses and entities for each argument in the triples. For Chinese word sense generation, because there is no existing Chinese word sense lexicon, we intuitively use the Youdao[4] translator to obtain the Chinese word senses. However, it is hard for these entity words since translators may mistake these English entity words for common words and return the unwanted literal translations. For example, the movie name "Onward" is translated into "向前", which means forth or forward, rather than its correct Chinese entity meaning "≪1/2的魔法≫", which literally means half of the magic. Therefore, to supplement more Chinese entity meanings, we use the heuristic extraction method by designing 3 regular patterns to extract the Chinese entities from Chinese Wikipedia. For example, from the sentence in Wikipedia "≪1/2的魔法≫(英语Onward)是一部于2020年上映的美国3D计算机动画奇幻电影", we design the pattern "(.*)(英语.*)(.*)" to extract the Chinese entity "≪1/2的魔法≫" for the English mention "Onward".

B. CANDIDATE TREES GENERATION

We group triples with the same $Arg1$ and the same relation rel into a tree of depth 1. However, in Probase, a tree can have thousands of children, just as a tree rooted in the argument "movie" can have more than 6,000 children, which can result in an explosive number of candidate trees. We set the number of children $C = 4$, because the statistical result shows that if it is greater than 4, it can at most have over 10,000 candidate trees for some source trees, which is a big computational cost, and if it is less than 4, it will be more difficult to capture the tree structure semantic than when it is 4, see section IV. Besides, we try to divide each tree into small trees also for another two reasons: one reason is to disambiguate the root by making the root of the tree have only one meaning with these children, the other reason is to provide some closer context for each child node in a small tree. Hopefully, after clustering, the root of the tree has only one meaning and the meanings of these children are close.

In detail, the hierarchical agglomerative clustering (HAC) algorithm is used to cluster the child nodes by choosing these close nodes. The number of clusters is determined by the number of root word meanings, which ensures that the root has only one meaning in each tree, and also by the number of child nodes, which should be equal to or less than 4. In clustering, we use real value vectors to represent nodes. If it is an entity, we average the named entity embedding with the surface mention embedding. For example, when the root of the tree is "fruit", the word "banana" can be grouped with "apple", and when the root of the tree is "company", the word "Microsoft" can be grouped with "apple". After clustering, we try to enumerate combinations of all the Chinese word senses and entities of each node in the source tree to build all candidate trees, as shown in Fig. 1.

C. SCORING MODEL

In this section, we describe the scoring model for the tree structure, shown in Fig. 3. Generally, the scoring model captures the semantic correlation between source tree and candidate tree through the mention feature and context feature, and captures the inner structure information in the candidate tree through coherence feature and relation feature.

1) MENTION FEATURE (MF)

As shown in Fig. 3, mention feature shows the semantic correlation between English mention surface text in the source tree and Chinese mention surface text in the candidate tree. Here, we add a translation layer to translate English space to

[4]http://dict.youdao.com/w/

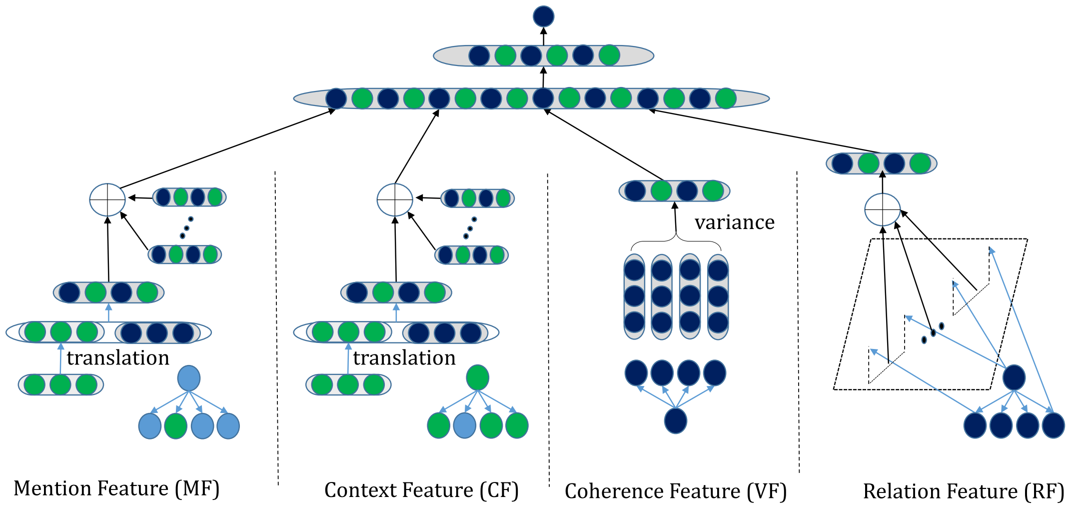

FIGURE 3. **Overview of our proposed neural network based joint model.**

Chinese space, formulated as follows.

$$v_i^{(m)} = W_t x_i^{(m)} + b_t, \tag{2}$$

where W_t and b_t are model parameters, and v_i is the projection vector of the mention surface text from English space to Chinese space.

Same as the work [14], we pre-train the translation parameters, W_t and b_t, by leveraging a small number of bilingual word pairs $(w^{(ch)}, w^{(en)})$), see training detail in Experiment IV. The loss function of this pre-train step is defined as follows

$$L(W_t, b_t) = \sum_i ||W_t w_i^{(en)} + b_t - w_i^{(ch)}||. \tag{3}$$

After getting W_t and b_t from training. we can calculate v_t from (2). Then, we concatenate the translated embedding $v_i^{(m)}$ with the entity embedding e_i, then feed it into a fully connected layer, obtaining the hidden feature between e_i and x_i at mention level. We apply vector averaging over all nodes and get the hidden mention feature $f^{(m)}$ between the whole source tree and candidate tree. We formulate the steps as follows:

$$\begin{aligned} f_i^{(m)} &= Tanh(W_m[v_i^{(m)}; e_i] + b_m), \\ f^{(m)} &= \frac{1}{C} \sum_{1 \le i \le C} f_i^{(m)}, \end{aligned} \tag{4}$$

where W_m and b_m are model parameters.

2) CONTEXT FEATURE (CF)

Given a mention node in the tree structure, we treat all other sibling nodes as context information because they share the same relation with one root argument and also have similar meanings after clustering. This feature has a similar structure to the mention feature, except that the input is replaced by its context. We define the context embedding $x_i^{(ctx)}$ as the average mention embedding of those sibling nodes, as described below:

$$x_i^{(ctx)} = \frac{1}{C-1} \sum_{1 \le j \le C, j != i} x_j^{(m)}. \tag{5}$$

Then, the context embedding goes through steps formulated as follows:

$$\begin{aligned} v_i^{(ctx)} &= W_t x_i^{(ctx)} + b_t, \\ f_i^{(ctx)} &= Tanh(W_{ctx}[v_i^{(ctx)}; e_i] + b_{ctx}), \\ f^{(ctx)} &= \frac{1}{C} \sum_{1 \le i \le C} f_i^{(ctx)}, \end{aligned} \tag{6}$$

where W_t, b_t share the same parameters as mention feature, and W_{ctx} and b_{ctx} are model parameters.

3) COHERENCE FEATURE (VF)

The work [14] exploits the coherence feature that exists in the rows or columns of the table. Inspired by that, we focus on the coherence of the child nodes in the tree, which could help align the child nodes in the candidate tree.

$$v^{coh} = var(\{e_i | e_i \in E\}). \tag{7}$$

If the source tree has only one child, the feature vector will be set to zero vector. Here we also add a fully connected layer, so that the dimension of the feature vector corresponds to the previous features.

$$f^{(coh)} = Tanh(W_{coh} v^{coh} + b_{coh}), \tag{8}$$

where W_{coh} and b_{coh} are model parameters.

4) RELATION FEATURE (RF)

The coherence feature helps align the child nodes in the candidate tree, beyond that, the relation feature can also help align the root node based on these child nodes. This intuition comes from TransH [15], which tries to capture the semantic distance difference between two arguments in different hyperplanes of the relations. Hopefully, the 4 child nodes can help align the root node with the correct meaning. Here, we have one relation in Probase and 44 relations in ConceptNet, then we use r_i to represent the $i-th$ relation. Last, a full connected layer is added.

$$v^{(rel)} = \frac{1}{C}\sum_{1\leq i\leq C}(e_{root} - e_i - w_{r_i}^{\top}(e_{root} - e_i)w_{r_i}),$$
$$f^{(rel)} = Than(W_{rel}v^{(rel)} + b_{rel}), \quad (9)$$

where the w_{r_i} translates the node embedding into different hyperplanes corresponding to relation r_i [15].

D. MODEL TRAINING AND PREDICTION

As mentioned in Section II, we handle the structural tree translation by defining a scoring function of the source tree and candidate tree pair. These four features, including mention feature, context feature, coherence feature, and relation feature, are concatenated and fed into a two-layer fully connected network to obtain the final score.

$$f_1 = W_1[f^{(m)}; f^{(ctx)}; f^{(coh)}; f^{(rel)}] + b_1,$$
$$S(X,E) = W_2f_1 + b_2, \quad (10)$$

where W_1, b_1, W_2, and b_2 are model parameters.

Optimization Strategies: In the training set, for each source tree, we have one positive gold Chinese tree and many corrupted trees, which are made by replacing one node or several nodes of gold Chinese tree with random Chinese word senses or entities.

Just like the pairwise ranking model, the candidate trees of each pair are compared: the candidate tree with more correctly Chinese word senses or entities is ranked higher than the other one in the pair. Here, we take RankNet [16] with Adam stochastic optimizer [17] as our implementation.

$$Loss = \sum_{i=1}^{N}\max(0, 1 - S(X,E_1) + S(X,E_2)), \quad (11)$$

where N is the number of training instances and E_1 is ranked higher than E_2.

IV. EXPERIMENT

A. EXPERIMENTAL SETUP

1) PRE-TRAINED WORD EMBEDDINGS

We use Jul. 2019 dump of English[5] and Chinese[6] Wikipedia as text corpora to train the word embedding, which contains 5,346,897 English articles and 1,232,543 Chinese articles, respectively. In order to obtain both entity embedding and common word embedding at the same time, all entities in the anchored text are treated as special words. For example, the anchor text "Carl Linnaeus" in the sentence "The bald eagle was one of the many species originally described by Carl Linnaeus ..." is replaced as an English entity by the special word "[[Carl_Linnaeus]]". Chinese text also has the same process and we use Jieba tool[7] for Chinese text segmentation. We use Word2Vec [18] to train English and Chinese Wiki corpora for word embedding respectively.

2) PRE-TRAINED SPACE TRANSLATION MATRIX

In order to pre-train the translation matrix used in the mention feature and context feature to convert mention embedding from English space to Chinese space, we use the Youdao translator to collect a bilingual lexicon with 200,000 translation pairs. We choose those English words that are polysemous, a named entity, or a single word pair. In total, 10,422 translation pairs are picked as our pre-training dataset.

3) DATASET

We collect some triples with 4 relations and their Chinese translation triples, and split them into train, dev, and test datasets with the ratio of 7:1:2 (2103:287:590). These 4 relations are "IsA", "MadeOf", "AtLocation", and "RelatedTo".

4) HYPERPARAMETER SETTINGS

As hyperparameters, all these models use the same word embedding dimension $D_m = 300$ and the size of each final feature vector is set to $D_h = 100$. Since our training dataset is not large, we use dropout [19] layer behind each feature layer to prevent the model from overfitting, with a keep probability of 0.8. For each epoch, we iterate over the whole training dataset and evaluate the model performance on the dev dataset every epoch. After training, we pick up the best model on the dev dataset as our final model and report the performance on the test dataset. Our model implementation is done in Python using the Pytorch[8] machine learning library.

5) EXPERIMENTAL SETTINGS

In this section, we describe some baseline experiments, including the state-of-the-art model, for comparison with our model.

First, we design some following baseline models.

- Surface Matching: Given an English triple (Arg_1, rel, Arg_2), we first get the Chinese word senses and entities for Arg_1 and Arg_2. Then, we select the highest-ranked Chinese candidate for each argument and combine them as the best translation.
- Hints Similarity: Given an English triple, we get all Chinese candidate triples by doing the Cartesian product on the Chinese word senses and entities of these two arguments, and then use the Web search to measure the correlation between these two Chinese arguments in each candidate triple. Specifically, we concatenate these two Chinese arguments as a query and put it into

[5]https://dumps.wikimedia.org/enwiki/
[6]https://dumps.wikimedia.org/zhwiki/
[7]https://github.com/fxsjy/jieba
[8] https://pytorch.org/

the Chinese search engine. We count the co-occurrence frequency in snippets and select the highest-ranked candidate triple as the translation result of this English triple.

Then, some following experiments are carried out on the basis of direct translation.

- Direct Triple Translation: Given a triple, we translate it through the Google translator by feeding the Google translator with the contextualized sentence instead of the word or triple. For example, given a triple (plant, IsA, banana), we contextualize it to the sentence "banana is a plant", which is trying to make the translator to capture more semantics in the sentence than the triple itself, then we feed it into the translator and design a few patterns to extract the translation result to get the corresponding translated Chinese triple.
- Direct Tree Translation: This baseline experiment adopts a similar way as the Direct Triple Translation experiment, except that we construct the sentence from all the nodes of each source tree. For example, we convert the source tree, shown in Fig. 1, to the sentence "Movie, Old school, Onward, Accepted, Outbreak", and also design a few patterns to extract the translation result to get the corresponding translated Chinese triples.

We also design some following baselines based on neural network.

- CNN: We concatenate all the nodes and the relation in the source tree as a token sequence with a maximal length of 6 (a root node, 4 child nodes, and one relation), then feed this sequence of embeddings into the convolution kernel and max-pooling layer to get the feature vector, and we also do the same to the candidate tree. Then, we concatenate the two feature vectors of the source tree and candidate tree, and feed it to the fully connected layer to get the final score. We use the same training strategy and optimization method as our framework.
- LSTM: The LSTM-based baseline model is similar to the above CNN-based baseline model, except that we feed each sequence of embeddings into the LSTM cell instead of CNN and use the last hidden state as the final feature vector of the source tree and candidate tree. We use the same training strategy and optimization method as our framework.
- BERT [20]: The Bert-based baseline model is also similar to the above CNN-based baseline model, except that we feed these token embeddings into the Bert encoder instead of CNN and use the [CLS] token embedding as the feature vectors of the source tree and candidate tree. We use the same training strategy and optimization method as our framework.

We exploit some out-of-the-box machine translation systems to do translation as comparisons. These out-of-the-box systems are ModernMT [21], Marian NMT [22] and OpenNMT [23]. We follow the same experiment steps as the Direct Tree Translation experiment, except that we replace the Google translator with these out-of-the-box translation systems.

We also implement the work based on an adaptive neural network in [13], which is the state-of-the-art model for knowledge base translation.

- Adaptive Neural Network (ANN): The state-of-the-art model proposed for knowledge base translation in the work [13] tries to translate each English triple and Chinese triple into a similar space, and chooses the Chinese candidate triple with the shortest distance to the English triple as the translation result.

Our model also has the following variations, converting the four feature vectors into the final score in different ways.

- Ours + Cosine: We concatenate these four feature vectors and use another vector with dimension 400 ($4{*}D_h$) to cosine the concatenated vector to get the final score. We do the same training and optimization for this variant model according to our model.
- Ours + Average: We first average these four feature vectors into one vector and then feed it into a fully connected layer to get the final score. We also do the same training and optimization for this variant model according to our model.

B. EXPERIMENTAL RESULT AND ANALYSIS

We preform experiments described in Section IV-A5 and the results are shown in Table 1. We report Micro accuracy as the evaluation metric, which is the percentage of correct triples over all triples in the test dataset. We do not use Macro accuracy like [14] as an evaluation metric, since almost all trees are of the same size. We do not use BLEU metric, because, in our case, the end-to-end translation is from an English word or phrase to a Chinese word or phrase for each argument of the triples, not like natural sentence translation.

As shown in Table 1, *Surface Matching* performs the worst, because the method cannot consider both two arguments and their relation as context to choose the proper Chinese candidate, which is crucial for disambiguation.

As shown in Table 1, *Hints Similarity* is slightly better than *Surface Matching*, because Web search helps find the correlation between two arguments by counting their co-occurrence. However, it can not explicitly capture the semantic of the relation. For example, when translating the triple (can, AtLocation, shelf), the wrong candidate pair ("能/could", "架子/shelf") occurs more frequently than the correct pair ("罐头/a kind of container", "架子/shelf") when searching on the Internet, because "能" (could), as an auxiliary word, appears more often than the word "罐头" (a kind of container).

Direct Triple Translation and *Direct Tree Translation* both have better results than *Surface Matching* and *Hints Similarity*, because the translator can better understand word sense ambiguity, but the translation effect of named entities needs to be further improved. The accuracies of the experiments

TABLE 1. Comparison of accuracy of the different models for knowledge base translation.

Model type	Model	IsA	MadeOf	AtLocation	RelatedTo	Avg
Common baselines	Surface Matching	62.3	62.9	64.4	63.2	63.2
	Hints Similarity	64.7	65.2	66.2	65.5	65.4
	Direct Triple Translation	66.1	69.3	73.4	72	70.2
	Direct Tree Translation	68.7	70.9	74.3	73.3	71.8
NN baselines	CNN	74.1	75.7	76.2	75.6	75.4
	LSTM	75.6	76.1	76.8	75.9	76.1
	Bert	76.5	77.9	78.9	77.9	77.8
Out-of-the-box	ModernMT	73.5	73.6	74.7	72.6	73.6
	Marian NMT	73.3	72.9	74.5	72.1	73.2
	OpenNMT	72.7	72.9	73.9	71.7	72.8
SOTA	ANN	77.4	78.6	79.5	78.5	78.5
Our Variants	Ours+Cosine	79.4	79.7	81.2	80.5	80.2
	Ours+Average	80.9	81.1	81.6	80.8	81.1
	Ours	81	81.5	82.2	81.7	81.6

from these out-of-the-box machine translation systems, such as ModernMT, Marian NMT and OpenNMT, are similar to Direct Triple Translation result, since Google translator can also be seen as an out-of-the-box machine translation system.

Table 1 also shows the neural model baselines outperform other non-neural model baselines, demonstrating the powerfulness of the neural network's feature representation ability. The greater correlation between the source tree and the candidate tree the neural network can capture, the higher the accuracy will be. CNN-based model, LSTM-based model, and Bert-based model try to capture the correlation between source tree and candidate tree, but they cannot capture the inner structure feature of the candidate tree. The Bert-based model performs better than the other two models, which shows the effectiveness of the pre-training language model even based on a small amount of training data. However, the improvement of the Bert-based model is not significant enough, compared with other two models, because the input sequence is not a natural sentence, which makes it less well represented than a natural sentence.

Models with different feature fusion methods have different accuracies. *Ours+Cosine* has the worst effect, since different features cannot be fully fused, whereas *OURS+Average* is close to OURS, which benefits from the good fusion of all features. All these three different feature fusion models outperform the state-of-the-art ANN-based model, suggesting that it is important to consider all these structure features and combine them appropriately.

Generally, all these methods perform better at the relation "AtLocation" and worse at the relations "IsA" and "RelatedTo", because the triples with the relation "AtLocation" are usually less ambiguous than triples of other relations.

We further investigate the effect of the number of child nodes (tree size) on the final translation result. Fig. 4 shows that the accuracy is the highest when the number of child nodes reaches 4 and 5, because our model best captures the structural semantics of the tree. The green and purple lines indicate that the accuracy will also peak in the absence of relation feature or coherence feature when the child nodes reach 4 and 5. However, in the dark line, there is no peak without relation feature and coherence feature, which means

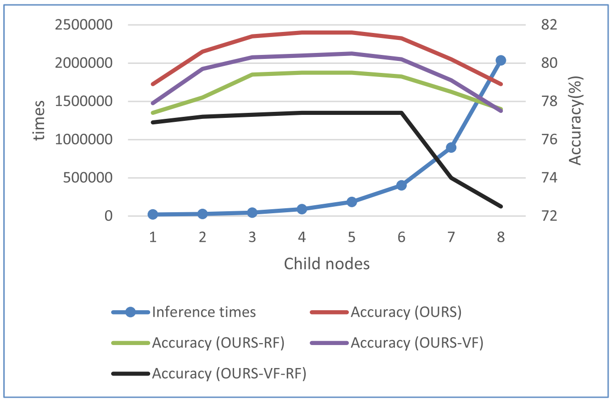

FIGURE 4. Times and accuracy of inference with different number of child nodes.

that the relation feature and coherence feature best capture the tree structure when the child nodes reach 4 or 5. The dark line also shows that when there are no relation feature and coherence feature, and the child nodes are greater than 6, the accuracy drops rapidly, because if the root node of the source tree has too many child nodes, the root node may have more than one meaning, which means some triples in the tree must be translated incorrectly. The blue line shows that the times of inference increase exponentially with the increase of child nodes, because the number of candidate trees increases exponentially with the increase of child nodes. Therefore, we choose $C = 4$ as the maximum number of the child nodes, which has both better accuracy and less computational cost.

C. ABLATION STUDY

In this section, we further explore the contributions of the various components in our framework.

Table 2 evaluates the results of different feature combinations by displaying the accuracy of the top 3 candidate trees with the highest score. As shown in Table 2, all features contribute to better results. Single mention feature or context feature performs better than a single coherence feature or relation feature, and the feature combination performs better than a single feature. Coherence feature, relation feature, or combination of them usually perform worse than others,

TABLE 2. Ablation test for our model.

Model	Acc_top1 (δ)	Acc_top2 (δ)	Acc_top3 (δ)
MF	72.1 (-9.5)	77.7 (-9.6)	80.3 (-13.2)
CF	72.4 (-9.2)	79.2 (-8.1)	81.1 (-12.4)
RF	73.6 (-8)	80.4 (-6.9)	84.5 (-9)
MF+CF	77.4 (-4.2)	83.4 (-3.9)	85.6 (-7.9)
VF+RF	78.1 (-3.5)	84.3 (-3)	87.2 (-6.3)
MF+CF+VF	79.5 (-2.1)	85.3 (-2)	88.4 (-5.1)
MF+CF+RF	80.4 (-1.2)	86.4 (-0.9)	90.4 (-3.1)
Full	81.6 (0)	87.3 (0)	93.5 (0)

TABLE 3. Accuracies on the test dataset with different model training methods.

Optimization strategy	Acc
Ranking-based optimization	81.6
Hinge loss optimization	76.3

because they fail to attend to information from the source tree, which cuts off the correlation between the source tree and the candidate tree. The coherence feature and relation feature are used to capture the inner candidate structure semantic, which complements the model to choose a better candidate tree.

For a specific example, one English tree (“song”: “alright”, “baby”, “black & white”, IsA) is correctly translated into (“歌曲/song”: “[[我心所愿]]/[[my desire]]”, “[[宝贝 贾斯汀·比伯歌曲)]]/[[baby_(Justin bieber song)]]”, “[[痞子英雄]]/[[rascal hero]]”, IsA) in *Full* model, whereas in *MF+CF* model, it is translated into (“歌曲/ song”: “[[我心所愿]]/[[my desire]]”, “宝宝/baby”, “[[痞子英雄]]/[[rascal hero]]”, IsA). The *Full* model can consider the structure information inside the candidate tree, which can correctly translate the entity “baby” into a song name“[[宝贝_(贾斯汀·比伯歌曲)]]/ [[baby_(Justin bieber song)]]”, which is a Chinese entity, because in this candidate tree, all of the child nodes are named entities, and coherence feature can help find this rule.

To compare with ranking-based optimization strategy, we also try the hinge loss optimization strategy during training. We use the hinge loss by maximizing the difference between correct English-Chinese tree pairs and the corrupted ones. For the corrupted tree, we also randomly replace some nodes in the candidate Chinese tree with others. In Table 3, it shows that hinge loss is less effective than RankNet in our model, probably because that hinge loss treats all negative candidate trees the same, thus other negative candidate trees with more corruptions become less effective in the training.

V. APPLICATION

To make further use of the framework we designed, we translate the entire Probase and ConceptNet (English part) into Chinese Zh-Probase and Zh-ConceptNet.

A. PROBASE TRANSLATION

1) TRAINING AND TRANSLATING

Probase only has an “IsA” relation, so we can use the model we trained to translate it into Zh-Probase without adding more training data.

TABLE 4. Size of existing taxonomic knowledge bases.

	Zh-Probase	CN-Probase	zhishi.me
Concepts	2,094,825	270,000	17,936
Instances	4,532,110	17,000,000	511,667
Concept-instance pairs	7,054,382	-	959,581
Concept-subconcept pairs	4,238,111	-	2,003
IsA pairs	11,292,493	33,000,000	961,587

TABLE 5. Accuracies of existing Chinese taxonomic bases. The Kappa coefficients [25] of two annotators suggest the substantial agreement.

Knowledge Base	Accuracy (Kappa)
Probase	93.0% (0.75)
Zh-Probase	86.6% (0.88)
CN-Probase	95.0%
zhishi.me	100%

2) RESULT

After translation, we get Zh-Probase and compare it with two well-known Chinese taxonomic knowledge bases CN-Probase [9] and zhishi.me [10] in terms of coverage and accuracy.

a: COVERAGE

As shown in Table 4, Zh-Probase is of the same order of magnitude as CN-Probase and **11.74** times larger than zhishi.me. We further evaluate the overlap between Zh-Probase and these Chinese taxonomic knowledge bases. Since CN-Probase is not open-source, we use sampling to calculate the overlap ratio. First, we sample 500 “IsA” triples from CN-Probase through its public API, where **1%** of them are in Zh-Probase. Then, we sample 500 “IsA” triples from Zh-Probase, where **6%** of them are in CN-Probase. zhishi.me is open-source and there are only **5,243** overlapping triples between Zh-Probase and zhishi.me. The reasons for the slight overlap between Zh-Probase and CN-Probase are as follows: First, as shown in Table 4, CN-Probase has more instances and fewer concepts whereas Zh-Probase has more concepts and fewer instances. Second, CN-probase has more named entities than Zh-Probase since its data source is the Chinese encyclopedias. Third, many of the most recent entities are collected in CN-Probase rather than Zh-Probase, because CN-Probase is released later. Since Zh-Probase has a small overlap with existing Chinese taxonomic knowledge bases, Zh-Probase can greatly enrich them. In addition, due to its large concept space and broader topics, it will exhibit a stronger ability in capturing the implied semantics, as demonstrated in [24].

Therefore, we can conclude that although the “IsA” triples in CN-Probase is three times as large as in Zh-Probase, Zh-Probase can still greatly enrich the existing Chinese taxonomic knowledge bases.

b: ACCURACY

To evaluate the quality of Zh-Probase, we sample 500 triples from Probase and Zh-Probase, respectively. We ask two annotators, who are familiar with knowledge bases, to check

TABLE 6. Some "IsA" samples in Zh-Probase. Correct translations are in blue, and incorrect ones are in red.

English	OURS	DTT
(bank, natural feature)	岸边(sloping land), 自然特征(natural feature)	银行(a financial institution), 自然特征(natural feature)
(bank, man-made boundary)	岸边(sloping land), 人造边界(man-made boundary)	银行(a financial institution), 人造边界(man-made boundary)
(grand Arab capital, capital)	阿拉伯首都(grand Arab capital), 首都(a seat of government)	阿拉伯首都(grand Arab capital), 资本(money)
(spring, natural water)	泉水(ground water), 天然水(natural water)	春天(a season), 天然水(natural water)
(spring, surface water)	泉水(ground water), 地表水(surface water)	春天(a season), 地表水(surface water)
(bark, close range vocalization)	吠(noise from the dog), 近距离发声(close range vocalization)	树皮(covering of the tree), 近距离发声(close range vocalization)
(bark, vocalization)	吠(noise from the dog),发声(vocalization)	树皮(covering of the tree),发声(vocalization)
(scale, graphic learning material)	比例尺(a ratio), 图形学习材料(graphic learning material)	规模(size), 图形学习材料(graphic learning material)
(scale, voice exercise)	音阶(a series of notes), 语音练习(voice exercise)	规模(size), 语音练习(voice exercise)
(scale, animal covering)	鳞(the epidermis shed), 动物覆盖(animal covering)	规模(size), 动物覆盖(animal covering)
(scale, musicianship skill)	音阶(a series of notes), 音乐技巧(musicianship skill)	规模(size), 音乐技巧(musicianship skill)
(ball, social event)	舞会(the lavish dance), 社交活动(social event)	球(a game), 社交活动(social event)
(ball, physical activity)	舞会(the lavish dance), 体力活动(physical activity)	球(a game), 身体活动(physical activity)
(ball, celebration)	舞会(the lavish dance), 庆祝(celebration)	球(a game), 庆祝(celebration)
(florist, outlet)	花商(florist), 出口(a business place)	花店(florist), 出口(a business place
(go move shift, song)	去移动(go move shift), 鸣声(a sound)	转移(go move shift), 歌曲(song)
(Banks of the Ohio, song)	俄亥俄州的银行(Banks of the Ohio), 曲子(song)	俄亥俄州的银行(Banks of the Ohio), 歌(song)
(chin check, song)	下巴检查(chin check), 鸣声(a sound)	下巴检查(chin check), 歌(song)

TABLE 7. Size of Zh-ConceptNet.

Dataset	Size
1. Original Chinese Part of ConceptNet	438,307 (x1.00)
2. Translated English Part of ConceptNet	1,716,327 (x3.91)
Zh-ConceptNet (Merge 1 and 2)	**2,085,681 (x4.76)**

TABLE 8. Accuracy of different datasets. The Kappa coefficients of two annotators suggest a substantial agreement.

Approach	Accuracy (Kappa)
1. Original Chinese Part of ConceptNet	98.3% (0.58)
2. Translated English Part of ConceptNet	87.3% (0.85)
Zh-ConceptNet (Merge 1 and 2)	**89.6%** (0.79)

the correction of these triples. The results are shown in Table 5. The accuracy of our designed model on the created test dataset is lower than the full translation of Probase, because the test dataset is carefully constructed, which has more word ambiguities and named entities, whereas there are still a lot of less ambiguous triples in Probase, which can be easily translated. In addition to the inherent error around 7% (accuracy 93.0%) in Probase, our translation approach introduces only an additional error of 6.4% (accuracy 86.6%).

c: QUALITATIVE RESULTS

We compare some typical translation results based on our method and the Direct Triple Translation method. Our approach can handle some intractable word sense disambiguations better, such as "bank", "bark", "scale", as shown in Table 6. However, our method also fails to handle some triples, as shown in the red part in Table 6, because both arguments in these triples are very ambiguous.

B. CONCEPTNET TRANSLATION

1) TRAINING AND TRANSLATING

For translating ConceptNet, we also annotate many more triples of other relations from these 44 relations, except for those relations, which are meaningless or useless for Chinese after translation, such as "formof", "etymologicallyrelatedto", "derivedfrom", "Idbpedia/knownfor". For example, "applauds" has the "formof" relation with "applaud", however, the two arguments are the same after translation. Then, we retrain the model with the new annotated training dataset and then translate ConceptNet into Zh-ConceptNet.

2) RESULT

After translation, we get Zh-ConceptNet, a Chinese commonsense knowledge base, which comes from two sources: the original Chinese part of ConceptNet and the translation result of the English part of ConceptNet.

a: COVERAGE

As shown in Table 7, Zh-ConceptNet is **4.76** times the size of the original Chinese part of ConceptNet, which is a significant increase. The size of Zh-ConceptNet is not 4.91 times as large as we expected due to the overlap between the original Chinese part and the translated Chinese part.

To our best knowledge, there are no other Chinese knowledge bases dedicated to common sense, and Zh-ConceptNet will be the first large one with approximately 2 million triples, which we hope will be a valuable asset for Chinese commonsense research.

b: ACCURACY

To evaluate the quality of Zh-ConceptNet, we sample 500 triples from the original Chinese part in ConceptNet, the translation result of the English part of ConceptNet based on our model, and Zh-ConceptNet, respectively. We ask two annotators, who are familiar with knowledge bases, to check the correction of these triples. As shown in Table 8, there exists an error of 1% in the original Chinese part of ConceptNet, which comes from the crowdsourcing error. The accuracy of Zh-ConceptNet has also been improved to **89.6%** due to the high quality of the merged original Chinese part from ConceptNet.

TABLE 9. Some examples in Zh-ConceptNet. Correct translations are in blue, and incorrect ones are in red.

English	OURS	DT
(date, fruit)/IsA	枣(a kind of fruit), 水果(fruit)	时间(time), 水果(fruit)
(ball, ballroom)/AtLocation	舞会(a lavish dance), 舞厅(ballroom)	球(a game), 舞厅(ballroom)
(can, shelf)/AtLocation	罐头(a kind of container), 货架(shelf)	可以(could), 货架(shelf)
(court, gymnasium)/AtLocation	球场(an area for sports), 体育馆(gymnasium)	法院(lawcourt), 体育馆(gymnasium)
(munition, arm)/MannerOf	军火(munition), 武装(any instrument for fighting)	军火(munition), 手臂(a human limb)
(spinach, can)/RelatedTo	菠菜(spinach), 罐头(the container)	菠菜(spinach), 可以(could)
(fan, blow)/RelatedTo	扇子(a device for creating a current of air), 吹(blow)	粉丝(an ardent follower), 吹(blow)
(fan, peacock)/RelatedTo	扇子(a device for creating a current of air), 孔雀(peacock)	粉丝(an ardent follower), 孔雀(peacock)
(fan, sector)/RelatedTo	扇(a device for creating a current of air), 扇形(sector)	粉丝(an ardent follower), 部门(department)
(brook, tolerate)/RelatedTo	容忍(tolerate), 姑息(tolerate)	小溪(a natural stream), 容忍(tolerate)
(blunt, money)/RelatedTo	生硬(hard), 钱(money)	生硬(hard), 钱(money)
(fouta, thin)/RelatedTo	伏塔加(V tower), 瘦(thin)	富塔(Futa), 瘦(thin)
(major ninth, interval)/RelatedTo	主要的第九(the major ninth), 间隙(interval)	主要的第九(the major ninth), 间隔(interval)
(melo, music)/RelatedTo	甜瓜(melon), 音乐(music)	melo(melo), 音乐(music)

c: QUALITATIVE RESULTS

Our approach can handle some intractable word sense disambiguations better, such as ''date'', ''ball'', ''court'', ''capital'', ''fan'', as shown in Table 9. However, our method also fails to handle some ''RelatedTo'' triples, as shown in the red part in Table 9. Since the ''RelatedTo'' relation is weak such as the triple (''blunt'', RelatedTo, ''money''), it becomes weaker after translation, and both direct translation and our model cannot handle it well.

VI. RELATED WORK

A. KNOWLEDGE BASE CONSTRUCTION

Previous work has looked at constructing knowledge bases as relational schemas using expert knowledge [12], [26] or crowdsourcing efforts [1], [27] or text extraction [2], [2], [9], [28]. In our work, we focus on building the knowledge base quickly and well, without requiring a lot of source datasets, or a lot of human efforts and time.

B. REPRESENTATION LEARNING FROM STRUCTURES

Recently, various neural networks have been proposed to capture the structural information of various forms of structural data, such as triples, trees, and graphs.

Many existing works try to learn representation in triples, such as TransE [29], TransH [15], TransR [30]. TransE learns vector embeddings for both entities and relations, based on the idea that the relation between two entities corresponds to a translation between the embeddings of entities in triples [29]. Since TransE has problems when modeling 1-to-N, N-to-1, and N-to-N relations, TransH is proposed to enable an entity to have different representations when involved in various relations [15]. Furthermore, TransR [30] models entities and relations in distinct spaces, i.e., entity space and multiple relation spaces (i.e., relation-specific entity spaces), and performs translation in the corresponding relation space.

Many existing works also try to learn the representation of trees. In general, different tree-structured encoders are proposed to embed the input data and different tree-structured decoders are proposed to predict the output trees. In [31], [32], they develop tree-structured autoencoders to learn vector representations of trees, and show better performance on tree reconstruction and other tasks such as sentiment analysis. Another work [33] also proposes to use a tree-structured encoder-decoder architecture for natural language translation.

Graph neural networks try to learn tree representation and this topic has received more and more attention in recent years [34]. Many authors generalize well-established neural network models like CNN that apply to regular grid structure (2-d mesh or 1-d sequence) to work on arbitrarily structured graphs [35], [36].

However, these structure representation methods are all based on one language, they cannot be applied directly to our case. Therefore, we propose our own representation learning method to capture the structure semantic of both the source tree and the candidate tree at the same time.

C. WORD SENSE DISAMBIGUATION AND NAMED ENTITY DISAMBIGUATION

Word Sense Disambiguation(WSD) is a long-standing challenge in natural language processing, where the mention of an open-class word is linked to a concept in a knowledge base, typically WordNet [26]. Cross-lingual WSD is that where the word senses of a word in a source language come from a separate target translation language. SemEval-2010 [37] and SemEval-2013 [38] are both about cross-lingual WSD. These tasks feature English nouns as the source words and word senses as translation in other languages, such as Chinese.

Named Entity Disambiguation (NED) is the task of linking a named-entity mention to an instance in a knowledge base, typically Wikipedia [39], Freebase [12], YAGO [28]. It is a crucial procedure of many complex natural language processing applications, such as information retrieval and question answering.

Recently neural network-based named entity disambiguation methods have established the most advanced results, such as in this work [40], where context, entity, and mention, together with neural similarity functions, are essential components. Different from general entity linking, some works [14], [41]–[43] focus on the entries in structural table.

Unlike the structural information that naturally exists in the table structure, we first try to construct trees from triples and then capture the semantics from the tree structure by our proposed neural network. We can also capture the relation feature, which is not contained in [14].

VII. CONCLUSION AND FUTURE

This paper proposes a neural network-based tree translation method for Chinese knowledge base construction. We construct trees based on triples, design a neural network to capture different features from the tree structure, and try to score each candidate tree to get the best translation. Experiments show that our approach achieves better performance than baseline methods, including the state-of-the-art method. In addition, based on our approach, we translate Probase and ConceptNet to Zh-Probase and Zh-ConceptNet respectively, which are valuable for the Chinese research community. Possible future works include designing a neural network to capture the semantic of the graph in knowledge bases or to translate knowledge bases with complex structures such as Framenet [44].

REFERENCES

[1] H. Liu and P. Singh, "ConceptNet—A practical commonsense reasoning tool-kit," *BT Technol. J.*, vol. 22, no. 4, pp. 211–226, Oct. 2004.
[2] W. Wu, H. Li, H. Wang, and K. Q. Zhu, "Probase: A probabilistic taxonomy for text understanding," in *Proc. Int. Conf. Manage. Data (SIGMOD)*, 2012, pp. 481–492.
[3] M.-H. Hsu, M.-F. Tsai, and H.-H. Chen, "Query expansion with ConceptNet and WordNet: An intrinsic comparison," in *Proc. Asia Inf. Retr. Symp.* Singapore: Springer, 2006, pp. 1–13.
[4] R. Speer and J. Lowry-Duda, "ConceptNet at SemEval-2017 task 2: Extending word embeddings with multilingual relational knowledge," in *Proc. 11th Int. Workshop Semantic Eval. (SemEval)*. Vancouver, BC, Canada: Association Computational Linguistics, Aug. 2017, pp. 85–89. [Online]. Available: https://www.aclweb.org/anthology/S17-2008
[5] S. Mukherjee and S. Joshi, "Sentiment aggregation using ConceptNet ontology," in *Proc. 6th Int. Joint Conf. Natural Lang. Process.*, 2013, pp. 570–578.
[6] B. Ding, H. Wang, R. Jin, J. Han, and Z. Wang, "Optimizing index for taxonomy keyword search," in *Proc. Int. Conf. Manage. Data (SIGMOD)*, New York, NY, USA: Association Computing Machinery, 2012, pp. 493–504, doi: 10.1145/2213836.2213892.
[7] J. Wang, H. Wang, Z. Wang, and K. Q. Zhu, "Understanding tables on the Web," in *Proc. Int. Conf. Conceptual Modeling*. Berlin, Germany: Springer, 2012, pp. 141–155.
[8] Y. Song, H. Wang, Z. Wang, H. Li, and W. Chen, "Short text conceptualization using a probabilistic knowledgebase," in *Proc. 22nd Int. Joint Conf. Artif. Intell. (IJCAI)*, 2011, pp. 2330–2336.
[9] J. Chen, A. Wang, J. Chen, Y. Xiao, Z. Chu, J. Liu, J. Liang, and W. Wang, "CN-probase: A data-driven approach for large-scale Chinese taxonomy construction," in *Proc. IEEE 35th Int. Conf. Data Eng. (ICDE)*, Apr. 2019, pp. 1706–1709.
[10] X. Niu, X. Sun, H. Wang, S. Rong, G. Qi, and Y. Yu, "Zhishi. Meweaving chinese linking open data," in *The Semantic Web—ISWC*. Berlin, Germany: Springer, 2011.
[11] W. Cui, X. Zhou, H. Lin, Y. Xiao, H. Wang, and W. Wang, "Verb pattern: A probabilistic semantic representation on verbs," in *Proc. 30th Conf. Artif. Intell. (AAAI)*, 2016, pp. 2587–2593.
[12] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, 2008, pp. 1247–1250, doi: 10.1145/1376616.1376746.
[13] X. Feng, D. Tang, B. Qin, and T. Liu, "English-Chinese knowledge base translation with neural network," in *Proc. 26th Int. Conf. Comput. Linguistics, Tech. Papers (COLING)*, 2016, pp. 2935–2944.
[14] X. Luo, K. Luo, X. Chen, and K. Q. Zhu, "Cross-lingual entity linking for Web tables," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 362–369. [Online]. Available: https://www.aaai.org
[15] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proc. AAAI Conf. Artif. Intell.*, 2014, pp. 1112–1119. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=3BD702B75032E3D2814951AE36786825?doi=10.1.1.486.2800&rep=rep1&type=pdf
[16] C. J. Burges, "From RankNet to LambdaRank to LambdaMART: An overview," *Learning*, vol. 11, p. 81, Jun. 2010.
[17] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980
[18] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 2013, pp. 3111–3119.
[19] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1. Minneapolis, MI, USA: Association Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://www.aclweb.org/anthology/N19-1423
[21] U. Germann, N. Bogoychev, E. Barbu, L. Bentivogli, N. Bertoldi, D. Madl, C. Buck, V. Sudharshan, D. Caroselli, L. Carvalho, A. Cattelan, M. Federico, B. Haddow, L. Mastrostefano, P. Mathur, A. Ruopp, A. Samiotou, M. Trombetti, R. Cettolo, V. Sudharshan, and J. van der Meer, "Modern MT: A new open-source machine translation platform for the translation industry," in *Proc. 19th Annu. Conf. Eur. Assoc. Mach. Transl., Projects/Products*, Riga, Latvia, 2016, p. 397. [Online]. Available: https://www.aclweb.org/anthology/2016.eamt-2.21
[22] M. Junczys-Dowmunt, R. Grundkiewicz, T. Dwojak, H. Hoang, K. Heafield, T. Neckermann, F. Seide, U. Germann, A. Fikri Aji, N. Bogoychev, A. F. T. Martins, and A. Birch, "Marian: Fast neural machine translation in C++," in *Proc. ACL*, Melbourne, VIC, Australia: Association Computational Linguistics, Jul. 2018, pp. 116–121. [Online]. Available: http://www.aclweb.org/anthology/P18-4020
[23] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. Rush, "OpenNMT: Open-source toolkit for neural machine translation," in *Proc. ACL, Syst. Demonstrations*. Vancouver, BC, Canada: Association Computational Linguistics, Jul. 2017, pp. 67–72. [Online]. Available: https://www.aclweb.org/anthology/P17-4012
[24] Y. Wang, H. Li, H. Wang, and K. Q. Zhu, "Toward topic search on the Web," Microsoft, Beijing, China, Tech. Rep. MSR-TR-2011-28, 2010.
[25] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977.
[26] G. A. Miller, "WordNet: A lexical database for English," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.
[27] M. Sap, R. Le Bras, E. Allaway, C. Bhagavatula, N. Lourie, H. Rashkin, B. Roof, N. A. Smith, and Y. Choi, "Atomic: An atlas of machine commonsense for if-then reasoning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 3027–3035.
[28] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A core of semantic knowledge," in *Proc. 16th Int. Conf. World Wide Web (WWW)*, 2007, pp. 697–706.
[29] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2787–2795.
[30] H. Lin, Y. Liu, W. Wang, Y. Yue, and Z. Lin, "Learning entity and relation embeddings for knowledge resolution," *Procedia Comput. Sci.*, vol. 108, pp. 345–354, Jun. 2017.
[31] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2011, pp. 151–161.
[32] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato, "Grammar variational autoencoder," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 1945–1954.

[33] J. Bradbury and R. Socher, "Towards neural machine translation with latent tree attention," in *Proc. 2nd Workshop Structured Predict. Natural Lang. Process.* Copenhagen, Denmark: Association Computational Linguistics, Sep. 2017, pp. 12–16. [Online]. Available: https://www.aclweb.org/anthology/W17-4303

[34] H. Cai, V. W. Zheng, and K. C.-C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 9, pp. 1616–1637, Sep. 2018.

[35] M. Henaff, J. Bruna, and Y. LeCun, "Deep convolutional networks on graph-structured data," *CoRR*, vol. abs/1506.05163, pp. 370–381, Jun. 2015. [Online]. Available: http://arxiv.org/abs/1506.05163

[36] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, Toulon, France, 2017, pp. 1–13. [Online]. Available: https://openreview.net/forum?id=SJU4ayYgl

[37] E. Lefever and V. Hoste, "SemEval-2010 task 3: Cross-lingual word sense disambiguation," in *Proc. 5th Int. Workshop Semantic Eval.*, Jun. 2010, pp. 82–87.

[38] E. Lefever and V. Hoste, "SemEval-2013 task 10: Cross-lingual word sense disambiguation," in *Proc. 2nd Joint Conf. Lexical Comput. Semantics (SEM), Proc. 7th Int. Workshop Semantic Eval.*, vol. 2, 2013, pp. 158–166. [Online]. Available: http://www.aclweb.org/anthology/S13-2029%5Cnhttp://www.aclweb.org/anthology/S13-2#page=194

[39] Z. Cai, K. Zhao, K. Q. Zhu, and H. Wang, "Wikification via link co-occurrence," in *Proc. 22nd ACM Int. Conf. Conf. Inf. Knowl. Manage. (CIKM)*, 2013, pp. 1087–1096, doi: 10.1145/2505515.2505521.

[40] O.-E. Ganea and T. Hofmann, "Deep joint entity disambiguation with local neural attention," in *Proc. Conf. Empirical Methods Natural Lang. Process.* Copenhagen, Denmark: Association Computational Linguistics, Sep. 2017, pp. 2619–2629. [Online]. Available: https://www.aclweb.org/anthology/D17-1277

[41] M. J. Cafarella, A. Halevy, D. Z. Wang, E. Wu, and Y. Zhang, "WebTables: Exploring the power of tables on the Web," *Proc. VLDB Endowment*, vol. 1, no. 1, pp. 538–549, 2008.

[42] Y. Ibrahim, M. Riedewald, and G. Weikum, "Making sense of entities and quantities in Web tables," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2016, pp. 1703–1712. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2983323.2983772

[43] Y. A. Sekhavat, F. D. Paolo, D. Barbosa, and P. Merialdo, "Knowledge base augmentation using tabular data," in *Proc. CEUR Workshop*, vol. 1184, 2014, pp. 1–5.

[44] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The Berkeley FrameNet project," in *Proc. 36th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 1998, p. 86. [Online]. Available: http://portal.acm.org/citation.cfm?doid=980845.980860

HAIJUN ZHANG received the B.S. degree in computer science from Northeastern University, Shengyang, China, in 2017. He is currently pursuing the M.S. degree in computer science with Shanghai Jiao Tong University, Shanghai, China.

His research interests include recommendation systems, natural language processing, knowledge base construction, event extraction, causal reasoning, knowledge-based reasoning, and expert system-based application.

Mr. Zhang's awards and honors include National Encouragement Scholarship, National Scholarship, Merit Student, and one national invention patent of China.