IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 26, 2020, accepted February 4, 2020, date of publication February 7, 2020, date of current version February 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2972379

English and Chinese Neural Metonymy Recognition Based on Semantic Priority Interruption Theory

CHUANDONG SU[1], XIAOXI HUANG[1], FUMIYO FUKUMOTO[2], JIYI LI[2], RONGBO WANG[1], AND ZHIQUN CHEN[1]

[1]College of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China
[2]Department of Computer Science and Engineering, University of Yamanashi, Yamanashi 400-8510, Japan

Corresponding author: Xiaoxi Huang (huangxx@hdu.edu.cn)

This work was supported in part by the Humanities and Social Sciences Research Program Funds from Ministry of Education of China under Grant 18YJA740016 and in part by the Major Projects of the National Social Science Foundation of China under Grant 18ZDA290.

ABSTRACT Metonymy is one of the types of common figurative languages and often used in human conversation without any difficulties. However, metonymy recognition in NLP requires a deep semantic/contextual processing to interpretation because it is highly related to the discourse of the contexts. Moreover, the fact that few available datasets of figurative languages make it more problematic. Motivated by the shortcomings of metonymy recognition, we develop several new data sets, including the Chinese version of the data, and design an end-to-end neural network metonymy recognizer. Our framework is based on the semantic priority interrupt theory and additional knowledge is introduced which makes to learn contexts effectively. Through a series of experiments, we show that our method is comparable to the state-of-the-art metonymy recognition method, especially we verified that metonymy trigger words information contributes to performance improvement in our model.

INDEX TERMS Metonymy recognition, neural network, semantic priority interrupt theory.

I. INTRODUCTION

Metonymy is one of the types of figurative languages which indicates the substitution of the concept, phrase or word being meant with a semantically related one [1]. Metonymy recognition has been intensively studied since neural network methods have attracted much attention. For example, the sentence "*England lost the semifinals.*" is a metonymy of location for people. Here, "England" does not refer to the country, but the team. In another sentence, "*Panasonic is of good quality.*" is a metonymy of organization for the product. "Panasonic" is not the company but its product. Humans can easily understand another concept of meaning beyond the literal expression of language. However, it is a serious bottleneck in the automatic machine understanding of language because it is highly related to the discourse of the contexts.

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

There are many forms of metonymy (e.g., "thing for people" and "thing for organization") [2], and it often hampers the development of NLP applications such as geographical analysis [3], [4] machine translation [5], question answering [6], anaphora analysis [7], [8], and geographic information retrieval [9]. For example, we note that the user asks an intelligent navigation system: "I am at Tokyo station now, where is the Beijing restaurant?". If the answer from the system is "first you take a car to Narita Airport, then you take a plane to Beijing.". We can see that the system does not understand that "Beijing" is not a geographical location of Beijing but a metonymic usage. This shows that the system does not infer the user's intention which is to find a restaurant with Beijing flavor food near to the Tokyo station. Another example is that we often use metonymic sentences in Google search engines, which often return the results we do not prefer. Therefore, improving the metonymy understanding ability of the machine can optimize and improve the performance of NLP applications.

In this paper, we focus on metonymy about a geographical noun which is related to the geographic analysis. There are several issues in metonymy recognition: (1) At present, the available metonymy datasets are very few, only SemEval 2007 task 8 (SemEval) [10] and ReLocaR [1], and there is no other language version of the data set available because manual tagging of the data is extremely costly. (2) The method based on rules and knowledge base depends on the construction of appropriate handcrafted features, which is also a very hard task, and the performance of the system depends on the quality of construction handcrafted features. (3) The method of metonymy recognition based on deep learning lacks the guidance of linguistic theory and has poor interpretability. To deal with the first problem mentioned in the above, we build a large-scale data set by merging the existing English metonymy data. Besides, we use machine translation technology to build the corresponding Chinese data sets. Our solution for the second and third problems is that we build an end-to-end neural metonymy recognizer based on the semantic priority interruption theory. It uses linguistic features such as metonymy trigger words, additional knowledge, location information, part of speech (POS), and uses the improved pre-training language model to extract metonymy features. Our model based on deep learning contributes to improve the interpretability of the model.

The main contributions of our work can be summarized: (1) We propose an enhanced metonymy recognition data set (EMR) and three Chinese version metonymy data sets. As far as we know, these Chinese data sets are the first trials we put forward in the field of metonymy recognition. (2) We propose a metonymy recognition method that introduces the theory of semantic priority interruption into the neural language model, using additional knowledge such as metonymy trigger words, location information, POS as additional features. (3) The experimental results on six data sets show that our model is comparable to the state-of-the-art approach in both the English and Chinese metonymy recognition tasks.

II. RELATED WORK

Metonymy is not only a figure of speech and figurative language but also a cognitive phenomenon [11], [12]. The processing of metonymy by the computer includes recognition, understanding, interpretation, and generation. Among them, metonymy recognition is the important foundation of computer metonymy understanding [13]. In early related research on metonymy recognition, Nissim and Markert [14] attempted to use syntactic relationships, grammatical roles, and knowledge base to overcome data sparseness and generalization problems. They also proposed the grammatical role of the potentially metonymic word (PMW). However, the method is still limited to classifying unknown data. Farkas *et al.* [15] utilized PMW and maximum entropy classifiers (ME) to achieve the accuracy of 85.2% in the SemEval. However, their method needs to do feature engineering and rely on external tools. Caroline *et al.* [16] achieved 85.1% accuracy in the SemEval by using a local grammar and global distribution features. Nastase and Strube [17] applied a support vector machine (SVM) with handcrafted features (provided by Markert and Nissim [10]) to achieve an accuracy of 86.1% in the SemEval. A similar research by Nastase and Strube [18] extends the research of Nastase *et al.* [19]. Their work makes use of SVM and a powerful knowledge base based on Wikipedia to achieve the accuracy of 86.2% in SemEval, which is by far the highest performance and still maintains state of the art (SOTA) status. Recently, Gritta *et al.* [1] demonstrated how the minimalist neural network approach combined with a predicate window can achieve competitive results (84.8% and 84.8%) in the SemEval and ReLocaR, respectively. The metonymy feature extraction ability of these traditional machine learning methods is weak, so using a more powerful deep learning model is required to improve the overall performance.

More recently, the most popular NLP method based on deep learning techniques has been intensively studied. Deep learning and word representation in vector space [20], [21] are fundamental technologies of NLP. Hochreiter and Schmidhuber [22] proposed long short-term memory (LSTM), that performs well on multiple NLP tasks. Kim [23] used a convolutional neural network (CNN) and pooling techniques to classify sentences and achieve good results. Bahdanau *et al.* [24] proposed the attention mechanism on machine translation tasks and obtained the results of SOTA. Raffel and Ellis [25] applied attention mechanism to the recurrent neural network (RNN) and achieved advanced results on multiple NLP tasks. Zhou *et al.* [26] and Hu [27] introduced the attention mechanism in the bi-directional LSTM (Bi-LSTM) to capture important semantic information and achieved the best results on the SemEval 2010 relationship classification task. Wu *et al.* [28] propose a metaphor recognition model based on CNN and Bi-LSTM, which achieve advanced results of metaphor recognition in the VU Amsterdam Metaphor Corpus (VUA). We will also introduce this model into baseline to test its effect on metonymy recognition. The combination of deep learning and word embedding can achieve better results than methods that rely on handcrafted features and rules. Do Dinh and Gurevych [29] show that relying solely on word embedding trained on large corpora can achieve results similar to other systems with additional resources. Mykowiecka *et al.* [30] found that only word embedding based solutions can achieve results comparable to complex solutions that require additional linguistic features. However, the feature extraction ability of a simple neural network is limited, and static word embedding can not solve the problem of polysemy. With the development of NLP technology, the pre-training language model has a good effect on each NLP task. ELMo used Bi-LSTM to generate dynamic word embedding solves the problem of polysemy [31]. OpenAI GPT [32] used Transformer [33] with strong feature extraction ability to construct a pre-training language model, which has a good performance in the task of natural language generation (NLG). BERT [34] changed the

one-way model of GPT into the bi-directional model, which further improved the ability of the pre-training language model. However, the serious problem of these deep learning models is like a black box, which has poor interpretability and lack of guidance of linguistic theory.

In the context of Chinese figurative languages, the attempts started late. It mainly focused on the rule-based approach and traditional machine learning. Wang *et al.* [35] used the maximum entropy model to identify Chinese nominal metaphors. Li *et al.* [36] identified Chinese similes by combining maximum entropy and conditional random field (CRF). Huang [37] uses dependency syntax to model different metaphor patterns and proposes a pattern matching algorithm to identify Chinese metaphors. The application of deep learning to Chinese figurative languages is very few. Metonymy is similar to metaphor, while Chinese metonymy does not even have relevant data sets and good solutions.

In summary, the main work of metonymy recognition is focused on traditional machine learning and simple neural network architecture. Moreover, most of the research focuses on English metonymy. It can be seen that the development of Chinese information processing urgently needs a breakthrough in Chinese metonymy recognition technology. Aiming at the problem of English and Chinese metonymy recognition, we use an end-to-end neural network metonymy recognizer based on the semantic priority interrupt theory to further expand and improve the methods of previous researchers.

III. METHODS

A. SEMANTIC PRIORITY INTERRUPTION THEORY OF METONYMY

Wilks [38] proposed the concept of semantic interruption and the model of preferential choice. The non-literal expression of metaphor would lead to semantic interruption. For example, in "*Your eyes are stars.*", the "eyes" are body organ entities, and the word "stars" are celestial entities. In the semantic priority interruption theory, the phrase will be semantically interrupted, thus judging the existence of metaphor language phenomena in this phrase. Because metonymy is similar to metaphor, we think that metonymy is also triggered by semantic priority interruption. For example, in the sentence "*Tokyo welcomes you!*", "Tokyo" is a geographical noun entity, and "welcome" means very happy to accept, usually the behavior of people. When a geographical noun entity is collocated with a human action verb, it will produce semantic priority interruption, thus triggering a metonymy. Therefore, the theory of semantic priority interruption can be used to distinguish metonymy and literal meaning. Three parts of speech of words can trigger semantic priority interrupt, as shown in Table 1. In the sentence "*Japan was a good experience for me.*", the collocation of the noun "experience" and the geographical noun "Japan" means that "Japan" is a tourist experience. In the sentence "*I really enjoyed that delicious Yamanashi.*", the adjective "delicious" and the geographical noun "Yamanashi" indicate that "Yamanashi" is a delicacy of Yamanashi. These special triggers will produce semantic priority interruption in sentences, thus trigger metonymy. The semantic priority interruption theory of metonymy can help us identify metonymy very well. In subsection III-B, we will model the trigger words and additional knowledge of trigger words to build an end-to-end neural metonymy recognizer.

TABLE 1. Examples of typical metonymy.

text	trigger	POS
Tokyo welcomes you!	welcomes	verb
Japan was a good experience for me.	experience	noun
I really enjoyed that delicious Yamanashi.	delicious	adjective

B. METNET: A END-TO-END NEURAL METONYMY RECOGNIZER

According to the semantic priority interruption theory of metonymy, we design an end-to-end metonymy recognizer based on the Transformer encoder. The model architecture is shown in Figure 1. We express the token sequence of the original text data as $I_1^t, I_2^t, \ldots, I_n^t$, where n is the sequence length. The token level of English is word level, and the token level of Chinese is character level. All tokens are from a vocabulary $\mathbb{V}$, $I_i^t \in \mathbb{V}$. We represent the metonymy trigger words sequence as $I_1^w, I_2^w, \ldots, I_m^w$ and the locations sequence as $I_1^l, I_2^l, \ldots, I_k^l$. M and k are the length of metonymy trigger words sequence and locations sequence respectively. The metonymy trigger words and the locations sequence can be found in the original text data tokens. So we annotate these sequences from the text data token. We use the metonymy trigger words sequence to query the WordNet (knowledge graph (KG)) $\mathbb{K}$, obtain the knowledge I_i^k $(0 \leq i \leq m)$ corresponding to I_i^w, and then get the knowledge sequence $I_1^k, I_2^k, \ldots, I_m^k$, as shown in Formula (1). Function $\mathbb{K}$ is used to acquire related external knowledge for the metonymy trigger words.

$$I_i^k = \mathbb{K}(I_i^w) \tag{1}$$

We obtain the POS I_i^p $(0 \leq i \leq m)$ corresponding to I_i^w by tagging the parts of speech of the trigger words sequence, so as to obtain the POS sequence $I_1^p, I_2^p, \ldots, I_m^p$, as shown in Formula (2), function POS means to acquire the part of speech of the metonymy trigger word.

$$I_i^p = POS(I_w^t) \tag{2}$$

We make these sequences into input sequences, with metonymy recognition special token $[CLS]$ at the beginning and segment separation special token $[SEP]$ between different feature sequences. We sum up these token sequences and corresponding segment embedding and position embedding to get the final input embedding sequence. The final input embedding construction method is shown in Formula (3).

$$E_j^F = I_j^F + position(I_j^F) + segment(I_j^F) \tag{3}$$

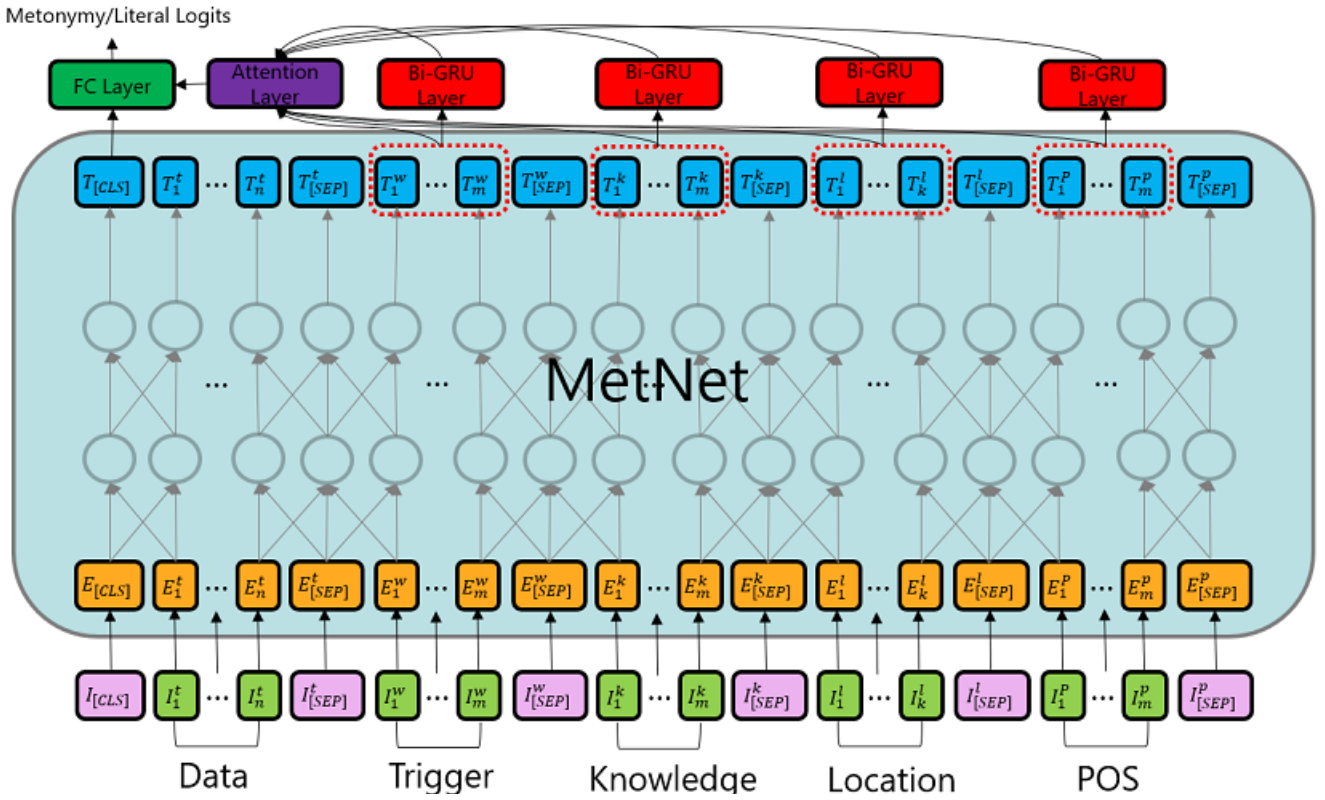

FIGURE 1. The overall architecture of our model (MetNet).

where, F represents final input embedding sets, $0 \leq j \leq n + 3m + k + 6$. Position embedding represents the position information of the token relative to the first token, and segment embedding represents the different segments of the token, which are divided by [*SEP*]. The *position* function will get the distance of the current token from the position of the first token to get the relative position information of the token. The *segment* function will assign different segment embeddings to different segments separated by [*SEP*] token to distinguish different segments. We have seven different sequence features: data sequence, trigger words sequence, knowledge sequence, locations sequence, POS sequence, and two special tags. We transform them into corresponding input embeddings E_j^F.

We use the Transformer encoder to encode these features. The number of layers, hidden neurons and self-attention head of the encoder are consistent with the BERT-base [34], and the specific super parameters are summarized in the subsection IV-C. Transformer is based on multiple self attention heads, which has stronger feature extraction ability than CNN, RNN and other deep neural networks. It has achieved good results in many NLP fields [33]. MetNet obtains the terminal deep semantic feature T_j^F ($T_j^F \in \mathbb{R}^H$) corresponding to E_j^F through the multi-layer Transformer encoder, and H is the size of hidden state. We use bi-directional gated recurrent unit (Bi-GRU) [39] to extract feature information of deep-layer order structure, including trigger words deep-layer feature sequence $T_1^w, T_2^w, \ldots, T_m^w$, knowledge deep-layer feature sequence $T_1^k, T_2^k, \ldots, T_m^t$, location deep-layer feature sequence $T_1^l, T_2^l, \ldots, T_k^l$, and POS deep-layer feature sequence $T_1^p, T_2^p, \ldots, T_m^p$.

Then these deep feature sequences are passed to Bi-GRU respectively. Bi-GRU is given by Formula (4)–(7).

$$z_t = \sigma(W_z x_z + U_z h_{t-1}) \tag{4}$$
$$r_t = \sigma(W_t x_z + U_t h_{t-1}) \tag{5}$$
$$\tilde{h}_t = tanh(W x_t + U(r_t \odot h_{t-1})) \tag{6}$$
$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{7}$$

Among them, z_t is the update gate, the function is the logic gate to update the neuron state, r_t is the reset gate, the function is to reset the neuron state, W and U are the Bi-GRU neuron parameter matrix, h_t and $\tilde{h}_t$ is the hidden state, σ and *tanh* is the activation function, $\odot$ is the Hadamard product. Then we combine these deep feature sequences with Bi-GRU's output and input them into the attention layer. This residual structure design can better integrate multiple deep-layer features. The attention layer is given by Formula (8)–(10).

$$s(x_i, q) = v^T tanh(Ax_i + Bq) \tag{8}$$
$$a_i = softmax(s(X_i, q)) \tag{9}$$
$$attention(q, X) = \sum_{i=1}^{N} a_i X_i \tag{10}$$

Among them, q is the query vector, v is the value vector, X is the input information, A and B are the attention layer parameter matrix, $s(x_i, q)$ is the sum of the attention scoring function, metonymy related features will give higher scores, and metonymy unrelated features will give lower scores, and the attention probability distribution a_i will be obtained through softmax, and finally the attention probability distribution value will be assigned to the input information X_i.

Then the metonymy recognition vector $T_{[CLS]}$ and the high-level metonymy feature vector of attention layer are passed to the full connection (FC) layer to obtain the two-dimensional metonymy discrimination vector, and use softmax to get the probability distribution of metonymy and literal meaning. As shown in Formula (11).

$$y_\tau = softmax(V^T x + b) \tag{11}$$

y_τ is a two-dimensional real value vector representing metonymy and literal probability, V and b are the FC layer neuron parameter matrix. In the process of training the model, we use the parameter weight of pre-training language model published by Google [34] to fine-tune the Transformer encoder layer. The Bi-GRU layer, attention layer and FC layer will use the training method to train the model through the Adam optimizer with adaptive learning rate. The training target is the cross entropy loss function, as shown in Formula (12).

$$\mathcal{L} = -\sum_{i=1}^{M} (\hat{y} \log y_{\tau 0} + (1 - \hat{y}) \log y_{\tau 1}) \tag{12}$$

where M is the number of training data samples, $\hat{y}$ is the real label of the data, 1 represents metonymy, 0 represents literal, $y_{\tau 0}$ and $y_{\tau 1}$ represent the prediction probability of whether the data belongs to metonymy and literal respectively, and $y_{\tau 0}, y_{\tau 1} \in [0, 1], y_{\tau 0} + y_{\tau 1} = 1$.

IV. EXPERIMENTS AND ANALYSIS

A. DATA SETS

1) SemEval

The earliest data set of metonymy recognition is SemEval 2007 shared task 8 proposed by Market *et al.* [10], which was annotated textcoloryellowfrom the data of British National Corpus (BNC). The corpus contains two types of metonymy: locations, and organizations. We mainly focus on location-related metonymy. The data set consists of two types of data, including literal (geographical territories and political entities) and metonymy (place for people, the place for product, the place for an event, capital for government or place for organization). SemEval consists of 340 metonymy data and 1458 literal data. This is probably the natural distribution of location-related metonymy in the text.

2) ReLocaR

Gritta *et al.* [1] annotated a new metonymy data set, ReLocaR, which is different from annotation criteria of SemEval. Annotation criterion of SemEval regards political entities as literal, while Gritta *et al.* considers political entities as metonymy. Metonymy and literal data distribution of ReLocaR are more balanced. ReLocaR consists of 1013 metonymy data and 995 literal data.

3) EMR

SemEval and ReLocaR have small data volume and only the English version. A large data set is necessary for metonymy recognition, while it is very difficult to label manually. Therefore, we merge several existing metonymy recognition data sets. Gritta *et al.* [1] also annotated additional metonymy data[1] to assist in the research of the metonymy recognition task, which came from CoNLL 2003 named entity recognition (NER) data sets. We combine SemEval, ReLocaR and this additional data set into an enhanced metonymy recognition data set (EMR). EMR has 3479 metonymic data and 6542 literal data. we checked the dataset manually and used the annotation criteria of ReLocaR to modify partial annotation of SemEval to make the data tend to be the same distribution.

4) CHINESE VERSION METONYMY DATA SETS

Chinese metonymy research is also a very important topic of Chinese natural language understanding, but there is no relevant data set. Therefore, we use the Baidu machine translation system[2] to transform the three data sets SemEval, ReLocaR and EMR into Chinese data textcoloryellowset SemEval-CN, ReLocaR-CN, and EMR-CN. We check the Chinese data sets and correct the wrong translation to ensure translation quality.

B. BASELINES

To compare our metonymy recognizer, we choose the following baselines: (1) A metonymy recognizer proposed by Farkas Gritta *et al.* [15] and based on potential metonymic features and maximum entropy classifier. (2) A metonymy recognizer based on SVM and knowledge base by Nastase and Strube [18]. (3) A metonymy recognizer based on Bi-LSTM and predicate window proposed by Gritta *et al.* [1]. (4) Kim [23] proposed TextCNN, a kind of CNN integrated with multiple channel filters. (5) A NLP model based on Bi-LSTM and attention mechanism, proposed by Zhou *et al.* [26]. (6) A metonymy recognition model based on CNN and Bi-LSTM, which is presented by Wu *et al.* [28]. (7) Peters *et al.* [31] proposed ELMo, and it is a pre-training language model based on Bi-LSTM. (8) Devlin *et al.* [34] proposed the BERT, and BERT is a bidirectional language model based on the Transformer encoder. Among them, the first three models all use the features related to metonymy trigger words and additional knowledge. The latter five models are pure deep learning models, which only encode the original text without using additional features.

[1] https://github.com/milangritta/Minimalist-LocationMetonymy-Resolution

[2] https://fanyi.baidu.com

TABLE 2. Specific setting of experimental hyperparameters.

Hyperparameter		Setting
Model	number of Transformer encoder layers	12
	hidden size per Transformer encoder layer	768
	self-attention heads per Transformer encoder layer	12
	number of Bi-GRU layers	4
	neurons per Bi-GRU layer	16
	neurons of FC layer	2
Training	static English word embedding	Glove
	static Chinese word embedding	W2V
	sequence length	256
	batches	64
	initial learning rate	1e-5
	epochs	20
	cross-validation folds	10

C. DATA PREPROCESSING AND HYPERPARAMETERS SETTING

The evaluation metrics of our metonymy recognition task are accuracy and F1 sorce, and accuracy is the most commonly used evaluation metric of the metonymy recognition task. Metonymy recognition related baselines' accuracy is reported according to the accuracy given in relevant papers. The accuracy of baselines for general NLP tasks is given by our experiments, and the hyperparameters are set according to the recommendations of corresponding papers. The hyperparameters of our model adopt the default hyperparameters of BERT-base [34]. The number of neurons in Bi-GRU layer, attention layer, and the best training hyerparameters are obtained by grid search in a reasonable range. Specific setting of experimental best hyperparameters are shown in Table 2. The sequence length of the most data in the metonymy data sets is less than 256. We thus set the sequence length to 256. Data larger than 256 will be truncated, and data smaller than 256 will be filled with zeros. Glove [40] will be used as static word embedding in English data sets of deep learning baseline, and Chinese word embedding pre-trained by Skip-Gram algorithm of Word2Vec (W2V) [20] with Baidu encyclopedia corpus will be used in Chinese data sets. English data sets use SpaCy[3] framework for data cleaning such as case-to-case conversion, stemming extraction and punctuation removal to improve the probability of word mapping to word vectors and alleviate the problem of out of vocabulary (OOV). Due to the special features of the Sino-Tibetan language family, the data processing of the Chinese words segmentation is needed and then the word vectors mapping is carried out. We use the Jieba[4] framework to segment Chinese words. The baselines of the pre-training language model directly encodes the data into dynamic word embedding. The hyperparameter settings of MetNet include 12 layers of Transformer encoder, 768 hidden layer neurons and 12 self-attention heads in each layer, 16 neurons in each Bi-GRU and 2 neurons in full connection layer. The training parameters of all models are 64 batches, the initial learning rate is 1e-5, and the number of learning epochs is 20.

[3]https://spacy.io

[4]https://github.com/fxsjy/jieba

TABLE 3. Experimental results of English metonymy recognition. Each data set has two evaluation metrics, the left is the accuracy (%), the right is the F1 score (%), and bold represents the best result.

Model	SemEval		ReLocaR		EMR	
ME+knowledge [15]	85.2	-	-	-	-	-
SVM+knowledge [18]	86.2	-	-	-	-	-
BiLSTM+predicate window [1]	84.6	-	84.8	-	-	-
TextCNN [23]	78.5	72.5	83.6	84.1	77.8	71.5
BiLSTM-Attention [26]	70.6	70.9	74.1	76.6	77.7	70.6
CNN-BiLSTM [28]	67.8	65.7	75.6	76.3	81.0	74.4
ELMo [31]	76.8	75.9	80.1	81.5	76.7	79.1
BERT [34]	77.4	78.3	77.7	79.8	77.3	78.8
MetNet (Our model)	**88.3**	**86.1**	**88.1**	**87.0**	**89.3**	**88.8**

TABLE 4. Experimental results of Chinese metonymy recognition. Each data set has two evaluation metrics, the left is the accuracy (%), the right is the F1 score (%), and bold represents the best result.

Model	SemEval-CN		ReLocaR-CN		EMR-CN	
TextCNN [23]	56.2	54.0	79.1	80.8	79.6	71.8
BiLSTM+Attention [26]	80.0	71.6	80.2	80.0	77.8	69.5
CNN-BiLSTM [28]	51.5	53.8	82.4	81.8	78.9	71.6
BERT [34]	86.4	84.0	87.1	85.1	83.6	87.0
MetNet (Our model)	**90.0**	**89.9**	**90.6**	**89.5**	**90.7**	**90.3**

The best model parameter weight saved in the verification set as the final model parameter weight. The training set and test set of all experimental data sets are divided into 90% and 10%. And ten-fold cross-validation was used in all experiments to make full use of the training set and alleviate the over-fitting. The experimental results were got by run ten times and take the average to reduce the interference of random factors.

D. EXPERIMENTAL RESULTS

Our experimental results on the English metonymy recognition dataset are shown in Table 3. MetNet is the best in three English metonymy recognition datasets. The experimental results show that metonymy trigger words and additional knowledge can significantly improve the neural network's ability of metonymy recognition, and use deep learning to directly model text can not achieve good results in the task of English metonymy recognition. Our experimental results on the Chinese metonymy recognition datasets are shown in Table 4. The experimental results are similar to those of English metonymy recognition. Our model performs best in three Chinese metonymy recognition datasets, and the performance of the pre-training language model is significantly better than that of the general deep learning model.

To verify which factor can affect performance of MetNet, we conducted ablation experiments of the model. The experimental results are shown in Table 5, and experimental results show that metonymy trigger words have the greatest impact on the model, additional knowledge and location information have a certain improvement on the model performance, and POS features have the least impact. At the level of model structure, we also designed corresponding ablation experiments. When the model does not use Bi-GRU layer or attention layer, the accuracy and F1 score of metonymy

TABLE 5. Experimental results of ablation experiments. Each data set has two evaluation metrics, the left is the accuracy (%), the right is the F1 score (%), and bold represents the best result, w/o represents not using a feature or model structure, + represents adding a neural network structure.

Model	English data sets						Chinese data sets					
	SemEval		ReLocaR		EMR		SemEval-CN		ReLocaR-CN		EMR-CN	
MetNet	**88.3**	86.1	**88.1**	87.0	**89.3**	**88.8**	90.0	**89.9**	**90.6**	**89.5**	**90.7**	**90.3**
w/o trigger	79.6	80.2	79.0	78.6	80.9	80.0	87.2	87.6	87.7	86.9	84.4	85.3
w/o knowledge	86.4	**87.0**	87.6	86.3	86.8	86.9	88.0	88.5	89.0	89.6	89.2	88.7
w/o POS	87.0	85.9	87.3	**87.1**	88.7	88.1	88.9	87.5	90.0	90.3	89.9	89.5
w/o location	84.2	85.0	83.0	83.8	84.2	82.6	88.6	89.2	88.2	**89.5**	85.6	85.7
w/o Bi-GRU layer	88.2	85.9	87.8	87.0	87.8	87.6	**90.1**	89.8	88.9	89.4	89.7	90.1
w/o Attention layer	88.1	85.5	86.7	**87.1**	88.4	**88.8**	89.5	88.4	90.5	**89.5**	89.3	90.0
+ Bi-GRU layer (64 neurons)	87.1	85.0	86.9	86.2	87.4	86.8	89.1	88.5	89.3	89.1	88.4	88.7
+ FC layer (64 neurons)	85.7	84.8	86.5	84.3	86.3	85.9	87.0	86.3	88.1	87.4	88.3	86.4

TABLE 6. Examples of metonymy recognition results of typical cases by several models. The Golds column is the gold label, and the other columns are the judgment of the corresponding model on whether the text is metonymy or literal meaning.

ID	Text	Location	Golds	TextCNN	BERT	MetNet
1	The match was part of a tournament in Austria, comprising Japan and Austria as well as Chile and Switzerland.	Austria	literal	literal	literal	literal
2	The match was part of a tournament in Austria, comprising Japan and Austria as well as Chile and Switzerland.	Japan	metonymy	literal	literal	metonymy
3	Ptolemy's geography describes commercial relations between western India and Alexandria, the chief eastern emporium of the Roman Empire.	India	metonymy	literal	metonymy	metonymy
4	UK lowers noise limits for three London airports.	UK	metonymy	literal	literal	metonymy
5	Bangladesh June M2 up 3.8 pct m / m, up 8.2 pct y / y.	Bangladesh	literal	metonymy	metonymy	literal
6	The finance minister of France met him.	France	literal	literal	metonymy	metonymy

recognition have declined, which shows that Bi-GRU layer and attention layer are helpful to capture high-level metonymy information. We try to increase the complexity of the model to explore whether we can continue to improve the performance of metonymy recognition. Before Bi-GRU layer and FC layer, we add the corresponding type of neuron network layer with 64 neurons, and find that the ability of the metonymy recognizer has declined significantly, because the complexity of the transformer encoder layer is high enough. Adding a neural network layer with high complexity in the outer layer will produce a violent shock to the weight of the pre-training model, thus affecting the performance of the metonymy recognizer.

E. ANALYSIS OF EXPERIMENTS AND CASES

From the experimental results of English and Chinese metonymy recognition, we can see that metonymy trigger words and external knowledge guidance can improve the metonymy recognition performance of the neural network metonymy recognizer. Location information and POS features can also be regarded as additional knowledge because location information contains the subject entity of metonymy. POS features are related to the parts of speech of metonymy trigger words. This knowledge belongs to implicit knowledge, which is different from our explicit knowledge introduced from WordNet. Our model and baselines using metonymy trigger words and knowledge as additional features show good results. However, only using a deep learning model can not play a good role. This shows that the semantic priority interruption theory and additional knowledge guidance of metonymy trigger words can help metonymy and literal discrimination, and these features are effective in English and Chinese metonymy recognition tasks. The reason why MetNet is better than the traditional machine model and knowledge-based metonymy recognition baselines is that the ability of metonymy feature extraction of pre-training language models based on Transformer encoders is stronger than the traditional machine learning model. The reason why our model is better than the metonymy recognition baselines is that the semantic priority interruption theory and additional knowledge guidance of metonymy trigger words are very helpful to the task of metonymy recognition rather than using general NLP model and general pre-training language model, and only encoding from text can't achieve a good result of metonymy recognition, because metonymy is a high-level semantic phenomenon, and metonymy features are difficult to extract from the text directly.

In Table 6, we show the metonymy recognition effect of several models in several typical sentences. In the following cases: (1) a sentence contains multiple geographical nouns, and there are two situations: metonymy and literal meaning (ID 1,2,4), (2) the sentence is long and the sentence structure is complex (ID 3), (3) the sentence structure is incomplete and with more noise (ID 5), our model can distinguish metonymy and literal meaning correctly. However, TextCNN and BERT have a higher misjudgment rate in these cases. We have also analyzed the case (ID 6) of MetNet's misidentification, which is mainly caused by the ambiguous referential relationship between the metonymy trigger words and the subject. In the sentence "*The finance minister of France met him.*", the "met" is human behavior. When it is collocated with a geographic information entity, it will trigger

the semantic priority interruption, but the subject associated with this sentence is ''The finance minister of France'' rather than ''France'', this can cause recognition errors. The complex syntactic structure of the subject will lead to errors in Metonymy recognition.

We draw four important conclusions from our experiments in the field of metonymy recognition: (1) The theory of semantic priority interruption (which we implement by introducing the features of metonymy trigger words and additional knowledge into the model) can improve the accuracy of the model. (2) The method of using deep learning directly to model text is not effective. (3) The effect of the pre-training language model combined with the linguistic theory is better than that of the traditional machine learning model. (4) The complex syntactic structure of the subject and the collocation of trigger words will affect the overall performance of metonymy recognition.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed an enhanced metonymy recognition data set (EMR) and obtained the Chinese metonymy recognition data sets through machine translation technology. We also introduced the semantic priority interruption theory of metaphor into metonymy. According to this theory, we designed an end-to-end neural metonymy recognizer and introduce metonymy trigger words features and external knowledge guidance. We demonstrated the validity of our model through a large number of experiments on English and Chinese metonymy recognition data sets. We also designed ablation experiments to verify that the semantic priority interruption theory of metonymy and found that external knowledge guidance contribute the overall performance of the metonymy recognition ability of the pre-training language model.

There are a number of interesting directions for future work. We should explore in several aspects: (1) metonymy is a special figurative language, and we will extend our approach to metaphor, simile, sarcasm, pun, and other figurative languages. (2) we only use the knowledge of WordNet, and we will use knowledge graph (KG) such as VerbNet,[5] ConceptNet[6] [41], FrameNet[7] [42] and Chinese HowNet[8] to obtain more rich external knowledge guidance. (3) Our work only uses the linguistic theory of semantic priority interruption. We will introduce more linguistic theories to our work to make the deep learning model more explanatory. (4) In error analysis, we showed that the complex syntactic structure of the subject affects the performance of the model. We will consider to incorporate syntactic structure information into the metonymy recognizer for the robustness of the model.

[5]https://verbs.colorado.edu/verbnet/
[6]http://conceptnet.io/
[7]https://framenet.icsi.berkeley.edu/fndrupal/
[8]http://www.keenage.com/

REFERENCES

[1] M. Gritta, M. T. Pilehvar, N. Limsopatham, and N. Collier, ''Vancouver welcomes you! Minimalist location metonymy resolution,'' in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 1248–1259.

[2] E. Shutova, J. Kaplan, S. Teufel, and A. Korhonen, ''A computational model of logical metonymy,'' *ACM Trans. Speech Language Process.*, vol. 10, no. 3, p. 11, 2013.

[3] B. R. Monteiro, C. A. Davis, and F. Fonseca, ''A survey on the geographic scope of textual documents,'' *Comput. Geosci.*, vol. 96, pp. 23–34, Nov. 2016.

[4] M. Gritta, M. T. Pilehvar, N. Limsopatham, and N. Collier, ''What's missing in geographical parsing?'' *Lang. Resour. Eval.*, vol. 52, no. 2, pp. 603–623, Jun. 2018.

[5] S.-I. Kamei and T. Wakao, ''Metonymy: Reassessment, survey of acceptability, and its treatment in a machine translation system,'' in *Proc. 30th Annu. Meeting Assoc. Comput. Linguistics*, 1992, pp. 309–311.

[6] D. Stallard, ''Two kinds of metonymy,'' in *Proc. 31st Annu. Meeting Assoc. Comput. Linguistics*, 1993, pp. 87–94.

[7] S. M. Harabagiu, ''Deriving metonymic coercions from wordnet,'' in *Proc. Usage WordNet Natural Lang. Process. Syst.*, 1998, pp. 1–7.

[8] K. Markert and U. Hahn, ''Understanding metonymies in discourse,'' *Artif. Intell.*, vol. 135, nos. 1–2, pp. 145–198, Feb. 2002.

[9] J. Leveling and S. Hartrumpf, ''On metonymy recognition for geographic information retrieval,'' *Int. J. Geograph. Inf. Sci.*, vol. 22, no. 3, pp. 289–299, Mar. 2008.

[10] K. Markert and M. Nissim, ''Semeval-2007 task 08: Metonymy resolution at semeval-2007,'' in *Proc. 4th Int. Workshop Semantic Eval. (SemEval)*, 2007, pp. 36–41.

[11] G. Radden and Z. Kövecses, ''Towards a theory of metonymy,'' in *Metonymy in Language and Thought*, vol. 4. Amsterdam, The Netherlands: John Benjamins Publishing Company, 1999, pp. 17–60.

[12] K. U. Panther and L. Thornburg, ''Metonymy,'' in *The Oxford Handbook of Cognitive Linguistics*. Oxford, U.K.: Oxford Univ. Press, 2007.

[13] Y. Peirsman, ''Example-based metonymy recognition for proper nouns,'' in *Proc. 11th Conf. Eur. Chapter Assoc. Comput. Linguistics, Student Res. Workshop*, 2006, pp. 71–78.

[14] M. Nissim and K. Markert, ''Syntactic features and word similarity for supervised metonymy resolution,'' in *Proc. 41st Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2003, pp. 56–63.

[15] R. Farkas, E. Simon, G. Szarvas, and D. Varga, ''GYDER: Maxent metonymy resolution,'' in *Proc. 4th Int. Workshop Semantic Eval. (SemEval)*, 2007, pp. 161–164.

[16] B. Caroline, E. Maud, and J. Guillaume, ''XRCE-M: A hybrid system for named entity metonymy resolution,'' in *Proc. 4th Int. Workshop Semantic Eval. (SemEval)*, 2007, pp. 488–491.

[17] V. Nastase and M. Strube, ''Combining collocations, lexical and encyclopedic knowledge for metonymy resolution,'' in *Proc. Conf. Empirical Methods Natural Lang. Process.*, vol. 2, 2009, pp. 910–918.

[18] V. Nastase and M. Strube, ''Transforming Wikipedia into a large scale multilingual concept network,'' *Artif. Intell.*, vol. 194, pp. 62–85, Jan. 2013.

[19] V. Nastase, A. Judea, K. Markert, and M. Strube, ''Local and global context for supervised and unsupervised metonymy resolution,'' in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn.*, 2012, pp. 183–193.

[20] T. Mikolov, K. Chen, G. Corrado, and J. Dean, ''Efficient estimation of word representations in vector space,'' 2013, *arXiv:1301.3781*. [Online]. Available: https://arxiv.org/abs/1301.3781

[21] 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.*, 2013, pp. 3111–3119.

[22] S. Hochreiter and J. Schmidhuber, ''Long short-term memory,'' *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] Y. Kim, ''Convolutional neural networks for sentence classification,'' 2014, *arXiv:1408.5882*. [Online]. Available: https://arxiv.org/abs/1408.5882

[24] D. Bahdanau, K. Cho, and Y. Bengio, ''Neural machine translation by jointly learning to align and translate,'' 2014, *arXiv:1409.0473*. [Online]. Available: https://arxiv.org/abs/1409.0473

[25] C. Raffel and D. P. Ellis, ''Feed-forward networks with attention can solve some long-term memory problems,'' 2015, *arXiv:1512.08756*. [Online]. Available: https://arxiv.org/abs/1512.08756

[26] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, and B. Xu, "Attention-based bidirectional long short-term memory networks for relation classification," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, 2016, pp. 207–212.
[27] D. Hu, "An introductory survey on attention mechanisms in NLP problems," 2018, *arXiv:1811.05544*. [Online]. Available: https://arxiv.org/abs/1811.05544
[28] C. Wu, F. Wu, Y. Chen, S. Wu, Z. Yuan, and Y. Huang, "Neural metaphor detecting with CNN-LSTM model," in *Proc. Workshop Figurative Lang. Process.*, 2018, pp. 110–114.
[29] E.-L. Do Dinh and I. Gurevych, "Token-level metaphor detection using neural networks," in *Proc. 4th Workshop Metaphor NLP*, 2016, pp. 28–33.
[30] A. Mykowiecka, M. Marciniak, and A. Wawer, "Literal, metphorical or both? Detecting metaphoricity in isolated adjective-noun phrases," in *Proc. Workshop Figurative Lang. Process.*, 2018, pp. 27–33.
[31] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," 2018, *arXiv:1802.05365*. [Online]. Available: https://arxiv.org/abs/1802.05365
[32] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. (2018). *Improving Language Understanding by Generative Pre-Training*. [Online]. Available: https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/languageUnderstand.paper.pdf
[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[34] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: https://arxiv.org/abs/1810.04805
[35] Z. Wang, H. Wang, and S. Yu, "Chinese nominal metaphor recognition based on machine learning," *High Technol. Lett*, vol. 17, no. 6, pp. 575–580, 2007.
[36] B. Li, L.-L. Yu, M. Shi, and W.-G. Qu, "Computation of Chinese simile with 'xiang," *J. Chin. Inf. Process.*, vol. 22, no. 6, pp. 27–32, 2008.
[37] X. Huang, "Research on some key issues of metaphor computation," (in Chinese), Ph.D. dissertation, Zhejiang Univ., Hangzhou, China, 2009.
[38] Y. Wilks, "A preferential, pattern-seeking, semantics for natural language inference," *Artif. Intell.*, vol. 6, no. 1, pp. 53–74, 1975.
[39] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*. [Online]. Available: https://arxiv.org/abs/1412.3555
[40] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
[41] H. Liu and P. Singh, "ConceptNet—A practical commonsense reasoning tool-kit," *BT Technol. J.*, vol. 22, no. 4, pp. 211–226, 2004.
[42] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The Berkeley framenet project," in *Proc. 17th Int. Conf. Comput. Linguistics*, vol. 1, 1998, pp. 86–90.

CHUANDONG SU received the B.S. degree in computer science and technology from the Henan University of Science and Technology (HAUST), Luoyang, China, in 2018. He is currently pursuing the dual M.S. degrees in computer science and technology with Hangzhou Dianzi University (HDU), Hangzhou, China, and the University of Yamanashi, Yamanashi, Japan. His research interests include natural language processing, metaphor computation, and deep learning.

XIAOXI HUANG was born in Wenzhou, Zhejiang, China, in 1979. He received the B.S. degree in computer science and technology from Zhejiang University (ZJU), Hangzhou, in 2001, and the Ph.D. degree in computer science and technology from Zhejiang University, Hangzhou, China, in 2009. From 2010 to 2011, he was a Postdoctoral Researcher with the Center of Study of Language and Cognition, Zhejiang University. Since 2012, he has been an Associate Professor with the College of Computer Science and Technology, Hangzhou Dianzi University (HDU), Hangzhou, China. His research interests include natural language processing, metaphor computation, and machine learning.

FUMIYO FUKUMOTO received the M.S. degree from the Centre for Computational Linguistics, University of Manchester Institute of Science and Technology (UMIST), Manchester, U.K., in 1993, and the Ph.D. degree from the Department of Information Science, The University of Tokyo, Tokyo, Japan, in 1997. She was a Research Assistant with the University of Yamanashi, until 1999 and an Associate Professor, until 2010, where she was appointed as a Professor of the Interdisciplinary Graduate School of Medicine and Engineering. Her current research interest is computational linguistics in particular lexical semantics.

JIYI LI received the B.S. and M.S. degrees from Nankai University (NKU), Tianjin, China, in 2005 and 2008, respectively, and the Ph.D. degree from the Department of Social Informatics, Graduate School of Informatics, Kyoto University (KyotoU), Kyoto, Japan, in 2013. He is currently an Assistant Professor with the Department of Computer Science and Engineering, University of Yamanashi, Yamanashi, Japan. His current interests include natural language processing and data mining.

RONGBO WANG was born in Yiwu, Zhejiang, China, in 1978. He received the B.S. degree in computer and application from the Zhejiang University of Technology (ZJUT), Hangzhou, in 1999, the M.S. degree in computer science and technology from Zhejiang University (ZJU), Hangzhou, in 2002, and the Ph.D. degree in electronic and information engineering from The Hong Kong Polytechnic University (PolyU), Hong Kong, in 2005. Since 2005, he has been an Associate Professor with the College of Computer Science and Technology, Hangzhou Dianzi University (HDU), Hangzhou, China. His research interests include natural language processing, question and answering systems, and machine learning.

ZHIQUN CHEN was born in Nanchang, Jiangxi, China, in 1973. He received the B.S. degree in computer software and theory from Jiangxi Normal University (JNU), Jiangxi, China, in 1996, and the M.S. degree in computer application from Zhejiang University (ZJU), Zhejiang, China, in 1999. Since 2006, he has been an Associate Professor with the College of Computer Science and Technology, Hangzhou Dianzi University (HDU). His research interests include natural language processing, text mining, and artificial intelligence.