IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

SPECIAL SECTION ON DATA-ENABLED INTELLIGENCE FOR DIGITAL HEALTH

Received March 25, 2019, accepted April 13, 2019, date of publication May 10, 2019, date of current version June 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2912584

Context Embedding Based on Bi-LSTM in Semi-Supervised Biomedical Word Sense Disambiguation

ZHI LI[1,2], FAN YANG[3], AND YAORU LUO[1]
[1]College of Electronics and Information Engineering, University of Sichuan, Chengdu 610065, China
[2]Key Laboratory of Wireless Power Transmission, Ministry of Education, University of Sichuan, Chengdu 610065, China
[3]Key Laboratory of Obstetric and Gynecologic and Pediatric Diseases and Birth Defects, Ministry of Education, Department of Gynecology and Obstetrics, West China Second Hospital, University of Sichuan, Chengdu 610041, China

Corresponding author: Fan Yang (sharry48@163.com)

This work was supported by the Key Research and Development Project of the Science and Technology Department of Sichuan Province under Grant 2019YFG0192.

ABSTRACT Word sense disambiguation (WSD) is a basic task of natural language processing (NLP) and its purpose to choose the correct sense of an ambiguous word according to its context. In biomedical WSD, recent research has used context embeddings built by concatenating or averaging word embeddings to represent the sense of a context. These simple linear operations on neighbor words ignore the information about the sequence and may cause their models to be flawed in semantic representation. In this paper, we present a novel language model based on Bi-LSTM to embed an entire sentential context in continuous space by taking account of word order. We demonstrate that our language model can generate high-quality context representations in an unsupervised manner. Unlike the previous work that directly predicts the word senses, our model classifies a word in a context by building sense embeddings and this helps us set a new state-of-the-art result (macro/micro average) on both MSH and NLM datasets. In addition, with the same language model, we propose semi-supervised learning based on label propagation (LP) to reduce the dependence on biomedical data. The results show that this method can nearly approach the state-of-the-art results produced by our Bi-LSTM when reducing the labeled training data.

INDEX TERMS Word sense disambiguation, semi-supervised learning, context embedding, biomedical domain.

I. INTRODUCTION

In the field of biomedicine, large amounts of domain-specific knowledge are embedded in biomedical texts (such as symptoms, treatments, diseases) [5]. Extracting information from these texts can improve the applications such as patient-centered care, clinical medicine research. With natural language processing (NLP) of biomedical text, a difficult challenge concerns lexical ambiguity. Some words in a text may have two or more senses, for instance, the word *cold* can refer to both a disease and a temperature sensation, but there is no ambiguity in the sentence "*I am taking aspirin for my cold*". Therefore, disambiguation of ambiguous words is a very important step in semantic understanding.

Word sense disambiguation (WSD) attempts to map words with multiple meanings to the most likely semantics according to their context. Commonly used approaches include supervised [1], [2], unsupervised [3]–[5], and knowledge-based approaches [6]–[8]. In biomedical WSD, supervised learning generally performs better than unsupervised and knowledge-based algorithms. However, this approach requires large amounts of high-quality data, and annotation of these data requires medical expertise. So reducing the cost of data processing is a challenging task in biomedical WSD.

Context representation is a critical step in WSD task. Commonly used methods of representing texts are based on word embeddings. Word embeddings (such as those generated by the popular software package, word2vec) represent words in a low-dimensional continuous vector space [9]–[11]. These vectors have been proved to be able to capture

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

semantic information. Recent studies have shown that context embeddings generated by word embeddings can represent the sense of a sentence [12]–[14]. These methods include concatenating the vectors of the words around the target word or averaging the neighboring words vectors and more. However, most of these approaches are based on a bag of words model which without considering the word order or the sense of the entire sentence.

In this paper, we explore the use of context embeddings as features for WSD problem. We compare context embeddings generated by different strategies and find out that representation based on Bidirectional Long Short Term Memory (Bi-LSTM) can perform better than others. Furthermore, according to the characteristics of biomedical texts, we propose a semi-supervised algorithm based on label propagation (LP). We demonstrate this method helps our model learn a classifier from a smaller size of labeled data and nearly approach the state-of-the-art results produced by our Bi-LSTM.

The rest of this paper is organized as follows:

Related work is reviewed in Section II. We introduce our language model and semi-supervised algorithm in Section III. Experimental results are shown in Section IV. Discussion in Section V and the conclusions and future work in Section VI.

II. RELATED WORK

WSD is a fundamental challenge to semantic understanding, which aims at selecting a proper meaning of polysemous words according to their context [15]. Commonly used WSD algorithms can be divided into three categories: supervised learning, unsupervised learning, and knowledge-based algorithms. Supervised algorithms use labeled data to learn the underlying classification mechanism, and then classify the ambiguous words in accordance with the contextually appropriate word sense [16], [46]. Since this method requires a large amount of labeled data, it is not the best choice in some cases especially when labeled training data are limited. Knowledge-based algorithms use the external knowledge information as the training data [32], [33], [35]. These data have high confidence and are normalized by experts. Unsupervised algorithms do not need labeled data, contexts concerning similar word senses are clustered in an unsupervised manner [18]–[20]. All these methods are widely used in biomedical WSD tasks. For instance, many studies have been proposed for disambiguation of clinical texts(Xu et al., 2012; Wu, Denny, et al., 2012; Wu et al., 2015), including methods based on topic-modeling(Chasin et al., 2014), traditional machine learning and deep learning based on optimized features(Moon et al., 2013; S. Moon et al., 2012; Antonio Jimeno Yepes, 2017).

In this work, we focus on semi-supervised learning which uses label propagation (Talukdar et al., 2009) to automatically label unlabeled data. We demonstrate that our method not only can reduce the dependence on labeled data, but also maintain the good performance of the model.

Word embedding is a commonly used method to encode words into a low dimensional space. Early approaches to word embedding used the One-Hot encoder which based on bag-of-words representation. This method represents each word as a vector with dimensionality equal to the number of unique terms in the vocabulary. Only one of these dimensions could take on the value 1 and the others are 0. The problem is that when using the one-hot encoder to represent the documents, the vector will be mostly empty. Therefore, this representation is too sparse to provide the intrinsic relationship between words especially when the corpus is large. In order to solve this problem, Mikolov et al. (2013) presented a neural language model to train word embeddings, known as Word2vec [9], [10]. Pennington et al. (2014) present another model, known as GloVe [11]. These two methods transform the bag-of-words representation to a continuous vector space representation. Similar words have similar vectors and geometric distances between words can reflect semantic similarity. In most NLP tasks, these vectors can be used to initialize the input layer of a neural network.

Generic word embeddings are widely used in WSD tasks. Context embeddings, a combination of word embeddings is usually used to represent the sense of a context. In order to make the context embeddings better contain semantic information, several researchers have proposed some linear methods to improve performance. These include Zhong and Ng (2010), Taghipour and Ng [36], Chen et al. [40], Rothe and Schütze [17] and more. These methods can be summarized as follows:

· ***Concatenation embeddings***. Given a window size t as the number of words on a single side, w_i is the target word and $(w_{i-t}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+t})$ is the context of word w_i. With this method, context embedding involves concatenating the vector of the context words into a larger word vector:

$$e(i) = (e(w_{i-t}), ..., e(w_{i-1}), e(w_{i+1}), ..., e(w_{i+t}))$$

· ***Average embeddings***. This method involves calculating the average of surrounding word embeddings. The sense of the contexts is expressed by averaging:

$$e(i) = \sum_{\substack{j=i-t \\ j \neq i}}^{j=i+t} \frac{e(j)}{2t}$$

· ***Weighted sum of embeddings***. This method sets a weight for each word, and the context embeddings are computed by weighted sum of the embeddings of the surrounding words. The common setting method is based on the distance from the target word:

$$e(i) = \sum_{\substack{j=i-t \\ j \neq i}}^{j=i+t} e(j)\frac{t-|i-j|}{t}$$

These methods only have simple linear operations on neighbor words of the target word. In order to represent the entire sentential context as a whole, Melamud *et al.* [21] used a Bidirectional LSTM to build context embeddings. This work showed that a language model can generate high-quality context representations and surpass or nearly reach state-of-the-art results on many NLP tasks. Yepes [22] showed approaches using support vector machine (SVM) with a combination of features (unigrams, bigrams, etc) can get better performance in biomedical WSD, they also explored word embeddings with LSTM to represent contexts. Both of these models generated context vectors during the course of using a LSTM to predict word senses directly when it is trained on a large number of labeled examples.

In our work, we train a language model to learn context embeddings by predicting a word. Then we use LP algorithm to label unlabeled data based on our context representations. Finally, we build sense embeddings to classify a word into the correct sense. Compared with traditional supervised learning, our semi-supervised method can reduce the dependence on labeled training data and get a better performance in biomedical WSD. We describe our methods in Section III.

III. MATERIALS AND METHODS

We use Bi-LSTM to predict the target word in a sentence to generate our context embeddings. Then we build sense embeddings by averaging context embeddings of the same labels. Finally, we assign a word in a context by calculating the maximal cosine similarity with the sense embeddings.

A. DATA SETS

We evaluated our semi-supervised model on MSH WSD data set [8] and NLM WSD data set [23]. These data sets can be found from http://wsd.nlm.nih.gov

1) MSH WSD DATA SET

MSH WSD data set has 203 ambiguous entities, including 106 ambiguous abbreviations, 88 ambiguous terms and 9 which are combination of both. Each instance containing the ambiguous word was assigned a CUI (Concept Unique Identifier) [24] from the 2009AB version of the UMLS®(Unified Medical Language System). For each ambiguous entity, up to 100 instances per sense can be found from MEDLINE baseline. There are 37,888 ambiguity cases in 37,090 MEDLINE citations in total [8].

2) NLM WSD DATA SET

NLM WSD data set has 50 ambiguous terms which are highly frequent. 552,153 cases are represented by these 50 terms. Each term is annotated with a sense number and be mapped to UMLS semantic types. There are 100 manually disambiguated samples for each term [23].

B. WORD EMBEDDINGS

We used *Word2Vec*'s *CBOW* to train our word embeddings (Mikolov et al., 2013a). In this approach, context words are used as inputs of a neural network and try to predict the target word. After training, the input weights are the final word embeddings (Fig 1). According to the ambiguous words, we collect the latest 4,000 retrieval texts from MEDLINE as our training corpus to generate the word embeddings. The hyperparameters are as follows: 200 dimensions, window-size 10, 10 negative samples.

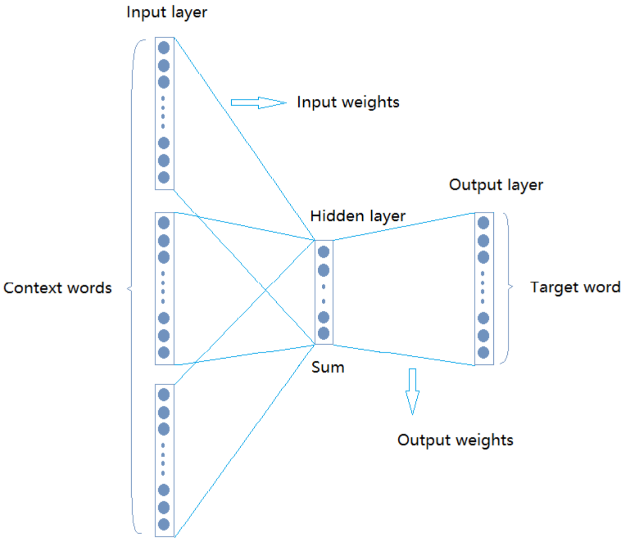

FIGURE 1. word2vec's CBOW.

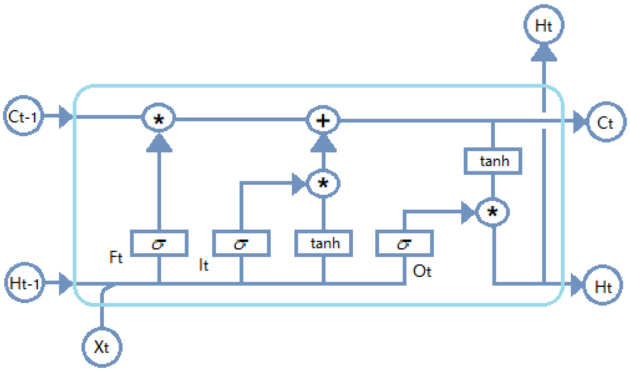

FIGURE 2. An unit of LSTM.

C. BIDIRECTIONAL LSTM

Recurrent neural network(RNN) is widely used in natural language processing. However, with the number of hidden layers increasing, there is a well-known problem of gradient vanishing or explosion. In order to solve the problem of long-term dependence, Hochreiter and Schmidbuber (1997) presented a variant structure of RNNs which named *Long Short Term Memory*(LSTM). Figure 2 gives a basic structure of an LSTM unit (Fig 2) [38], [39].

For a given time t, LSTM has an input gate i_t, an output gate o_t, a forget gate f_t and a memory cell c_t. Each gate is composed out of a sigmoid neural net layer and has the ability to remove or add information from the cell c_{t-1}. The gates output numbers between zero and one. A value of one means ‘‘all information pass’’, while a value of zero means ‘‘all information prohibited ‘‘. f_t decides what information

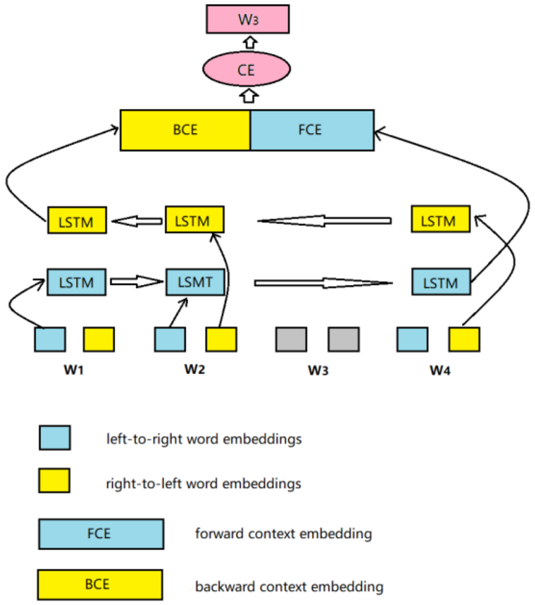
W3
CE
BCE
FCE
LSTM
LSTM
LSTM
LSTM
LSMT
LSTM
W1
W2
W3
W4
left-to-right word embeddings
right-to-left word embeddings
FCE
forward context embedding
BCE
backward context embedding

FIGURE 3. **Context embedding based on Bi-LSTM.**

can throw away from the old cell c_{t-1}. i_t decides to add the new information $\tilde{c}$ which from the input x_t and the old hidden output h_{t-1} into the cell c_{t-1}. Both of f_t and i_t are going to update the new cell c_t. Finally, the output h_t is decided by o_t and c_t. To update an LSTM unit at each time, the following formulas are used:

$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

$$\tilde{c} = \tanh(w_c \cdot [h_{t-1}, x_t] + b_c) \tag{3}$$

$$c_t = f_t * c_{t-1} + i_t * \tilde{c} \tag{4}$$

$$o_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = {o_t}^{*} \tanh(c_t) \tag{6}$$

Bidirectional LSTM (Bi-LSTM) is an update structure of LSTM (Graves and Schmidhuber, 2005). It consists of two LSTMs which one going left and another going right. The advantage is that the network will have messages about preceding words and succeeding words at the same time.

D. CONTEXT EMBEDDINGS BASED ON Bi-LSTM

In WSD, building context embeddings for target words is a necessary step. Traditional methods just use a simple linear operation on neighbor words embeddings, and these methods do not take account of word order. More recently, Antonio Jimeno Yepes (2017) have shown that using LSTM model to generate context embeddings can improve the performance of WSD in biomedical domain. Our model is different from that of Yepes. We train our Bi-LSTM to predict an ambiguous word given the surrounding context. Unlike their direct prediction of word senses, our language model just generates the context representations. Our proposed language model is illustrated in Figure 3.

As shown in Figure 3, w_3 is the ambiguous word. w_1, w_2 and w_4 are the context of word w_3. We first feed the forward LSTM network with the sentence words from left to right. At the same time, we feed the backward LSTM network with the sentence words from right to left. It is important to note that both networks are fed the whole sentence(without target word) and this is different from the Melamud's model. The parameters of these two networks are completely separate and our model will output two context embeddings, one is from the forward network and another is from the backward network. Let $FCE(w_3)$ represent the forward context embedding which from the last output of forward network and $BCE(w_3)$ represent the backward context embedding which from the last output of backward network. Next we concatenate $FCE(w_3)$ and $BCE(w_3)$ to generate a new vector $\overline{CE}(w_3)$:

$$\overline{CE}(w_3) = [FCE(w_3), BCE(w_3)] \tag{7}$$

After that, the new vector $\overline{CE}(w_3)$ will serve as input for a multi-layer perceptron $MLP(x)$ to generate the final context vector $CE(w_3)$:

$$CE(w_3) = MLP(\overline{CE}(w_3)) \tag{8}$$

$$MLP(x) = F_2(Re\mathrm{LU}(F_1(x))) \tag{9}$$

where $F_i(x)$ is a fully connected linear operation:

$$F_i(x) = w_i \cdot x + b_i \tag{10}$$

The Bi-LSTM model has 256 hidden units, the MLP hidden units are 400. The dimensions of output $CE(w_3)$ are equal to the size of word vectors. We train our model by minimizing sampled softmax loss with Adagrad. The learning rate has been set to 0.01 and learning rate decay is 0.01. Because $CE(w_3)$ contains entire sentential around the target word, so we use it directly as a representation of the context. In the next experimental part, it will be seen that this method is more effective than other methods when generating context embeddings.

E. SEMI-SUPERVISED WSD BASED ON LABEL PROPAGATION

Based on the method of Yuan et al.(2016), our semi-supervised method uses context embeddings to labeled unlabeled sentences based on the similarity of their context vectors. Note that our context embeddings built by Bi-LSTM while Yuan's built by LSTM. Like the word embeddings, similar contexts have similar embeddings in vector space. Then we build sense embeddings s_j by averaging context embeddings of all sentences of the same label.

Figure 4 shows a label-propagation graph. We use the context embeddings of the labeled sentences as our seed nodes (filled nodes), then the seed nodes will propagate their labels to unlabeled sentences(unfilled nodes). When the propagation is finished, nodes with the same color represent the sentences which have the similar senses. To classify a word w_i in a context, we calculate the maximum cosine similarity

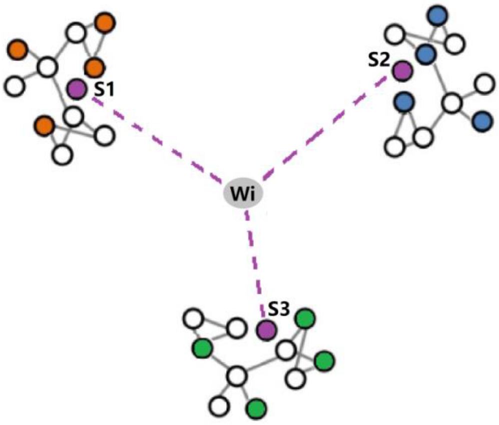

FIGURE 4. Semi-supervised classifier. Filled nodes represent the labeled data. Unfilled nodes represent the unlabeled data. s_j represents the sense embeddings.

between its context vector $CE(w_i)$ and sense embeddings s_j (blue nodes).

In our experiments, we randomly select 92% of the labeled sentences as our seed nodes for each lemma in both data sets (MSH and NLM). Then we remove the labels of other sentences and run our LP to label them. ith seed node and jth unlabeled node are connected by an edge whose weight is w_{ij}:

$$w_{ij} = \frac{\vec{c}_i \cdot \vec{c}_j}{\|c_i\| \cdot \|c_j\|} \tag{11}$$

c_i is the context embedding of ith seed node, c_j is the context embedding of jth unlabeled node. The weight w_{ij} reflects the similarity between node i and node j. Then the seed nodes propagate sense labels to other unlabeled examples depending on the probability p_{ij}:

$$p_{ij} = \frac{w_{ij}}{\sum\limits_{k=1}^{n} w_{ik}} \tag{12}$$

n represents the number of unlabeled nodes. If the number of nodes is too large, the propagation graph will be large, and if the number is too small, labeled data do not propagate sufficiently. So we connect two nodes if their similarity is above 90%. To avoid the emergence of isolated nodes, we force each node to connect to at least 5 neighbors.

All of our experiments are using 10-fold cross-validation. Macro and micro accuracy are used to evaluate our model.

IV. RESULTS

In this section, we use the MSH WSD data set and NLM WSD data set to assess the performance of our model. We mainly compare our model with other methods in two aspects: (1) Compare different ways to generate the context embeddings. (2) Explore the performance of our semi-supervised algorithm in biomedical WSD tasks.

TABLE 1. Macro and micro average WSD results in MSH data set.

Configuration	Macro average	Micro average
SVM+WE S100(Con)	94.46	94.31
SVM+WE S200(Con)	94.48	94.30
SVM+WE S100(Avg)	94.53	94.37
SVM+WE S200(Avg)	94.59	94.40
SVM+WE S100(Wsum)	95.13	94.98
SVM+WE S200(Wsum)	95.35	95.22
SVM+CE S100	96.11	95.96
SVM+CE S200	**96.28**	**96.09**

Numbers in bold indicate the best accuracy for a group of results

TABLE 2. Macro and micro average WSD results in NLM data set.

Configuration	Macro average	Micro average
SVM+WE S100(Con)	90.33	90.25
SVM+WE S200(Con)	90.38	90.30
SVM+WE S100(Avg)	90.54	90.41
SVM+WE S200(Avg)	90.57	90.46
SVM+WE S100(Wsum)	91.11	90.96
SVM+WE S200(Wsum)	91.25	91.08
SVM+CE S100	91.71	91.57
SVM+CE S200	**91.82**	**91.70**

Numbers in bold indicate the best accuracy for a group of results

A. CONTEXT EMBEDDINGS

In the method section, we present a neural network method to generate context embeddings. We use SVM as our basic classifier and compare our context embeddings (*CE*) with other three combination strategies which include *Concatenation (Con), Averaging (Avg), Weighted sum of words (Wsum)*. Embedding size 100(S100) and 200(S200) are used in our experiments. Table 1 shows the results in MSH WSD set and Table 2 shows the result in NLM WSD set.

As we can see in Table 1, strategies of concatenation and averaging don't have obvious improvement when we increased the dimensions of word vectors. However, *Wsum* gets a better performance with the same situation. In addition, under the same size of word vectors, context embeddings help our classifier get the best performance (96.28/96.09) and *Wsum* can perform better than other two traditional methods(*Con* and *Avg*).

Table 2 shows the similar results on NLM WSD data set. *SVM+CE(S200)* gets the highest macro average and micro average (91.82/91.70).

TABLE 3. Semi-supervised algorithm based on LP in MSH data set.

Configuration	Macro average	Micro average
Jimeno Yepes(2017)	95.97	95.80
LSTM	96.14	95.94
Bi-LSTM	**96.71**	**96.48**
LSTM + LP	96.01	95.75
Bi-LSTM + LP	96.53	96.30

Numbers in bold indicate the best accuracy for a group of results

TABLE 4. Semi-supervised algorithm based on LP in NLM data set.

Configuration	Macro average	Micro average
Jimeno Yepes(2017)	90.64	90.42
LSTM	91.60	91.47
Bi-LSTM	**92.25**	**92.07**
LSTM +LP	91.39	91.28
Bi-LSTM +LP	92.01	91.85

Numbers in bold indicate the best accuracy for a group of results

B. SEMI-SUPERVISED WSD RESULTS

We assess our semi-supervised algorithm by comparing with supervised learning (without LP) in this section. Two strategies are used to build context embeddings: *LSTM* and *Bi-LSTM*. Both of these models classify a word by calculating the maximal cosine similarity with their sense embeddings. At the same time, we use Jimeno Yepes (2017) as our baseline.

Table 3 lists the performance of our experiments on MSH WSD data set. As we can see, the system developed by Jimeno Yepes incorporated other features (such as part-of-speech, unigrams) in the word vectors. However, both of LSTM and Bi-LSTM outperform the results produced in this work and Bi-LSTM gets the best score in macro and micro average (96.71/96.48). Note that the score of *LSTM+LP* and *Bi-LSTM+LP* nearly reach the results when without using LP.

We use the same settings and run all models on NLM WSD data set. Similar results are shown in Table 4. *Bi-LSTM* gets the state-of-the-art results in both macro and micro average (92.25/92.07).

C. THE EFFECT OF LANGUAGE MODEL STRUCTURE

We have verified that under the same settings, our language model can perform better than other models on biomedical WSD tasks. In order to better explore the algorithm we present, we set another experiment to explore the effect of different parameters in our language model.

In this experiment, we use the different number of hidden units h and dimensions of context embeddings d to explore the influence of these parameters on the final results.

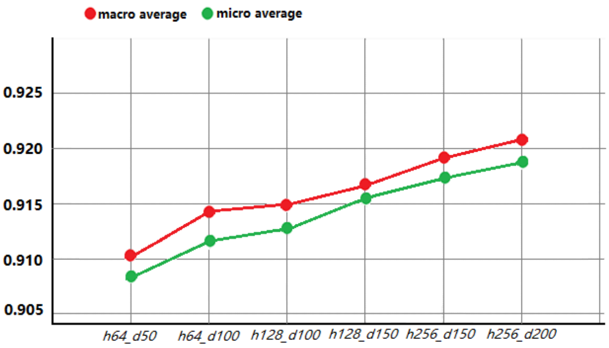

FIGURE 5. Macro and micro accuracy with different parameters: h is the number of hidden units, d is the context embedding dimension.

We select *Bi-LSTM+LP* as our basic model. In order to avoid overfitting due to the small amount of data, we train our model just on NLM WSD data set. The results are illustrated in Figure 5.

As we can see in Figure 5, there is a positive correlation between results and structure of our model. The promotion of the hidden layers can help improve performance, but it is not as obvious as the promotion of the context embedding dimension.

V. DISCUSSION

We did a series of experiments to evaluate the performance of our language model in biomedical WSD. And we also explored how the parameters affect the final results in our semi-supervised learning. We will analyze from three aspects.

A. COMPARISON OF CONTEXT EMBEDDINGS

We compared four types of context embeddings with different vector dimensions in our WSD tasks. As we can see in Table 1 and Table 2, the performance of *WE (Con)* or *WE (Avg)* is not as good as *WE(Wsum)* and our context embeddings. These two strategies (*Con* and *Avg*) just use a simple linear operation for the word embeddings, and this may not help context embeddings contain more messages about neighbor words. At the same time, the promotion of dimensions did not bring a significant increase in these two methods. *WE (Wsum)* sets a weight for each word and this can assign a fixed level of importance to different locations when generating context embeddings. As we can see it performed better than other two methods. The problem is that some words far from the target word may be important but this method cannot dynamically assign importance to each contextual unit. However, our context embeddings which generated by Bi-LSTM solved the problem of long-term dependence. This language model helped our system achieve the highest score in both data sets. In addition, *WE (Wsum)* and *CE* are benefited from embeddings of higher dimension.

B. COMPARISON OF LANGUAGE MODELS

Table 3 and Table 4 showed the performance of different language models in biomedical WSD tasks. Jimeno Yepes (2017) used SVM and LSTM to get the highest macro and micro average in the recent study. He also demonstrated other feature combinations (unigrams, bigrams, etc) can improve the performance of context representations. However, our *LSTM* and *Bi-LSTM* get a significant improvement without using any other external resources or handcrafted features and *Bi-LSTM* achieved the state-of-the-art results in both data sets. This may be because the model shared most of the parameters and these parameters are able to learn more relationships between words.

In our semi-supervised methods, we randomly removed 8% labels of the training data and our models nearly reach the score which made by our supervised learning. We proved our semi-supervised model can get a good level in both data sets.

C. THE EFFECT OF PARAMETERS

Fig 5 shows with increasing the hidden units, the difference in performance is not significant. This may be due to the size of data and our context embeddings might be seen as a pre-training which had already learned information of the context.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a novel language model to solve the problem of biomedical WSD. We demonstrated our context embeddings generated by Bi-LSTM can perform better than other traditional word embeddings. With the high quality context representations, the best performance was achieved by our supervised model (*Bi-LSTM*). In addition, after combined with label propagation, our semi-supervised method approximates the results of supervised learning while reducing the labeled data. Considering the minimal difference in performance, the reduced need for labeled data offers advantages for biomedical WSD tasks.

Our semi-supervised learning can carry more useful messages from a global objective, but it still needs enough data to ensure the performance of the model. Meanwhile, our unlabeled data were obtained by removing labels from the original data sets. We would like to see whether our LP can leverage additional unlabeled data (instead of eliminating labels) to improve our results. Finally, further developments in our language model may increase performance and we will explore unsupervised learning to deal with the serious loss of biomedical data in the future work.

REFERENCES

[1] Z. Zhong and H. T. Ng, "It makes sense: A wide-coverage word sense disambiguation system for free text," in *Proc. ACL Syst. Demonstrations, Assoc. Comput. Linguistics*, 2010, pp. 78–83.

[2] M. Stevenson, Y. Guo, R. Gaizauskas, and D. Martinez, "Disambiguation of biomedical text using diverse sources of information," *BMC Bioinf.*, vol. 9, no. 11, p. S7, 2008.

[3] T. Pedersen, "The effect of different context representations on word sense discrimination in biomedical texts," in *Proc. 1st ACM Int. Health Inform. Symp.*, 2010, pp. 56–65.

[4] S. Brody and M. Lapata, "Bayesian word sense induction," in *Proc. 12th Conf. Eur. Chapter Assoc. Comput. Linguistics, Assoc. Comput. Linguistics*, 2009, pp. 103–111.

[5] R. Chasin, A. Rumshisky, O. Uzuner, and P. Szolovits, "Word sense disambiguation in the clinical domain: A comparison of knowledge-rich and knowledge-poor unsupervised methods," *J. Amer. Med. Inform. Assoc.*, vol. 21, no. 2, pp. 842–849, 2014.

[6] R. Navigli, S. Faralli, A. Soroa, O. de Lacalle, and E. Agirre, "Two birds with one stone: Learning semantic models for text categorization and word sense disambiguation," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage.*, 2011, pp. 2317–2320.

[7] B. T. McInnes, T. Pedersen, Y. Liu, G. B. Melton, and S. V. Pakhomov, "Knowledge-based method for determining the meaning of ambiguous biomedical terms using information content measures of similarity," in *Proc. AMIA Annu. Symp.*, vol. 895, 2011, pp. 895–904.

[8] A. J. Jimeno-Yepes, B. T. McInnes, and A. R. Aronson, "Exploiting mesh indexing in medline to generate a data set for word sense disambiguation," *BMC Bioinform.*, vol. 12, no. 1, p. 223, 2011.

[9] 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

[10] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. NIPS*, 2013, pp. 3111–3119.

[11] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proc. EMNLP*, 2014, pp. 1532–1543.

[12] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng, "Improving word representations via global context and multiple word prototypes," in *Proc. ACL*, 2012, pp. 873–882.

[13] M. Kågebäck, F. Johansson, R. Johansson, and D. Dubhashi, "Neural context embeddings for automatic discovery of word senses," in *Proc. NAACL*, 2015, pp. 25–32.

[14] O. Melamud, I. Dagan, and J. Goldberger, "Modeling word meaning in context with substitute vectors," in *Proc. ACL*, 2015, pp. 472–482.

[15] R. Navigli, "Word sense disambiguation: A survey," *ACM Comput. Surv.*, vol. 41, no. 2, 2009, Art. no. 10.

[16] J. Camacho-Collados, M. T. Pilehvar, and R. Navigli, "A unified multilingual semantic representation of concepts," in *Proc. 53rd ACL*, Beijing, China, vol. 1, 2015, pp. 741–751.

[17] S. Rothe and H. Schütze, "Autoextend: Extending word embeddings to embeddings for synsets and lexemes," in *Proc. 53rd ACL*, Beijing, China, vol. 1, 2015, pp. 1793–1803.

[18] S. Manandhar, I. P. Klapaftis, D. Dligach, and S. S. Pradhan, "SemEval-2010 task 14: Word sense induction & disambiguation," in *Proc. SemEval*, Uppsala, Sweden, 2010, pp. 63–68.

[19] T. Van de Cruys and M. Apidianaki, "Latent semantic word sense induction and disambiguation," in *Proc. 49th ACL*, Portland, OR, USA, vol. 1, 2011, pp. 1476–1485.

[20] A. Di Marco and R. Navigli, "Clustering and diversifying Web search results with graph-based word sense induction," *Comput. Linguistics*, vol. 39, no. 2, pp. 709–754, 2013.

[21] O. Melamud, J. Goldberger, and I. Dagan, "*Context2vec*: Learning generic context embedding with bidirectional LSTM," in *Proc.20th SIGNLL Conf. Comput. Natural Lang. Learn. (CoNll)*, 2016, pp. 51–61.

[22] A. J. Yepes, "Word embeddings and recurrent neural networks based on long-short term memory nodes in supervised biomedical word sense disambiguation," *J. Biomed. Inform.*, vol. 73, pp. 137–147, Sep. 2017.

[23] M. Weeber, J. G. Mork, and A. R. Aronson, "Developing a test collection for biomedical word sense disambiguation," in *Proc. AMIA Symp., Amer. Med. Informat. Assoc.*, 2001, p. 746.

[24] B. T. Mclnnes, T. Pedersen, and J. Carlis, "Using UMLS concept unique identifiers (CUIs) for word sense disambiguation in the biomedical domain," in *Proc. AMIA Annu. Symp.*, 2007, pp. 533–537.

[25] M. Akbari, K. Relia, A. Elghafari, and R. Chunara, "From the user to the medium: Neural profiling across Web communities," in *Proc. ICWSM*, 2018, p. 118.

[26] J. Chen and H. Yu, "Unsupervised ensemble ranking of terms in electronic health record notes based on their importance to patients," *J. Biomed. Inform.*, vol. 68, pp. 121–131, Apr. 2017.

[27] S. Moon, H. Xu, T. Cohen, and B. Berster, "Word sense disambiguation of clinical abbreviations with hyperdimensional computing," in *Proc. AMIA Annu. Symp.*, 2013, p. 1007.

[28] S. Moon, S. Pakhomov, and G. Melton, (2012). Clinical Abbreviation Sense Inventory. Retrieved from the University of Minnesota Digital Conservancy. [Online]. Available: http://hdl.handle.net/11299/137703

[29] Y. Wu, J. C. Denny, S. T. Rosenbloom, R. A. Miller, D. A. Giuse, and H. Xu, "A comparative study of current clinical natural language processing systems on handling abbreviations in discharge summaries," in *Proc. AMIA Annu. Symp.*, 2012, pp. 997–1003, 2012.

[30] H. Xu, P. D. Stetson, and C. Friedman, "Combining corpus-derived sense profiles with estimated frequency information to disambiguate clinical abbreviations," in *Proc. AMIA Annu. Symp.*, 2012, pp. 1004–1013.

[31] J. Xu, Y. Zhang, and H. Xu, "Clinical abbreviation disambiguation using neural word embeddings," in *Proc. BioNLP*, 2015, pp. 171–176.

[32] S. P. Ponzetto and R. Navigli, "Knowledge-rich word sense disambiguation rivaling supervised systems," in *Proc. 48th ACL*, Uppsala, Sweden, 2010, pp. 1522–1531.

[33] T. Miller, C. Biemann, T. Zesch, and I. Gurevych, "Using distributional similarity for lexical expansion in knowledge-based word sense disambiguation," in *Proc. COLING*, Mumbai, India, 2012, pp. 1781–1796.

[34] E. Agirre, O. L. de Lacalle, and A. Soroa, "Random Walks for Knowledge-based Word Sense Disambiguation," *Comput. Linguistics*, vol. 40, no. 1, pp. 57–84, 2014.

[35] A. Moro, A. Raganato, and R. Navigli, "Entity linking meets word sense disambiguation: A unified approach," *Trans. ACL*, vol. 2, pp. 231–244, Dec. 2014.

[36] K. Taghipour and H. T. Ng, "Semi-supervised word sense disambiguation using word embeddings in general and specific domains," in *Proc. Annu. Conf. NAACL*, Denver, CO, USA, 2015, pp. 314–323.

[37] S. Rothe and H. Schütze, "AutoExtend: Extending word embeddings to embeddings for synsets and lexemes," in *Proc. 53rd ACL*, Beijing, China, vol. 1, 2015, pp. 1793–1803.

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

[39] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[40] X. Chen, Z. Liu, and M. Sun, "A unified model for word sense representation and disambiguation," in *Proc. EMNLP*, 2014, pp. 1025–1035.

[41] D. Yuan, J. Richardson, R. Doherty, C. Evans, and E. Altendorf, "Semi-supervised word sense disambiguation with neural models," in *Proc. COLING*, 2016, pp. 1374–1385.

ZHI LI was born in Chengdu, Sichuan, China, in 1975. He received the B.S. degree in electronics engineering from Shenyang Aerospace University, in 1997, and the M.S. degree in pattern recognition and intelligent system and the Ph.D. degree in applied mathematics from Sichuan University, in 2000 and 2004, respectively, where he has been a Researcher and a Teacher with the College of Electronics and Information Engineering.

In 2013, he was a Visiting Scholar with the Autonomous Computing Laboratory, The University of Arizona, USA. Since 2017, he has been a Professor with the Communication and Information Systems Engineering Department, Sichuan University. He has authored three books, more than 100 articles, and more than ten inventions. His research interests include wireless sensor networks, the smart IoT, compressive sensing, big data analysis, cognitive computing, and natural language process in medical education.

FAN YANG was born in Chengdu, Sichuan, China, in 1982. She received the master's degree in clinical medicine from the West China Clinical Medical College, Sichuan University, in 2007, and the Ph.D. degree from the Department of Obstetrics and Gynecology, West China Second Hospital, Sichuan University, in 2009.

She began her gynecologic oncology research in the State Key Laboratory of Biotherapy and Cancer Center, in 2007, and after that, she has been doing research in the Key Laboratory of Obstetric and Gynecologic and Pediatric Diseases and Birth Defects of Ministry of Education, West China Second Hospital, Sichuan University till now. Her main research interests include gene therapy for gynecologic oncology, immunotherapy for ovarian cancer, brain imaging changes in gynecological diseases, and medical data mining and processing.

Dr. Yang is the member of the New York Academy of Science, Chinese National Health and Family Planning Association, Chinese Maternal and Child Health Association, Maternal and Child Minimally Invasive Professional Committee, and Youth Committee of Hypos-copy Group.

YAORU LUO received the B.S. degree from the Chengdu College, University of Electronic Science and Technology, Chengdu, China, in 2015. His research interests include name entity recognition, data mining, and machine learning.