Word Sense Disambiguation Method Based on Graph Model and Word Vector

Tongtong Sun¹, Yongxing Du¹, Liyong Zhou¹*, Lijing Niu¹, Baoshan LI¹, Yanzhang Gong², Jianmin Dong³

¹School of Information Engineering, Inner Mongolia University of Science and Technology, Baotou, Inner Mongolia, China
²Inner Mongolia Autonomous Region Discipline Inspection and Supervision Big Data Laboratory Baotou Big Data R&D Application Center, Baotou, Inner Mongolia, China
³Inner Mongolia Autonomous Region Discipline Inspection and Supervision Big Data Laboratory Baotou Big Data R&D Application Center, Hohhot, Inner Mongolia, China

*Corresponding Liyong Zhou r’s e-mail: glzly@imust.edu.cn

Abstract. As a basic research of natural language processing, word sense disambiguation (WSD) has a very important influence on machine translation, classification tasks, retrieval tasks, etc. In order to solve the problem that existing disambiguation methods rely too much on knowledge base, a disambiguation method combining graph model and word vector is proposed in this paper. Firstly, in this method, the text data are preprocessed by removing punctuation marks and segmenting words. Secondly, the dependency relation is extracted by using the tool of PYLTP for dependency parsing, the words of dependency parent node are matched and the undirected graph is built, and the context knowledge of ambiguous words is selected according to the minimum path length set by the graph model. Finally, Word2Vec model is used to train Wikipedia corpus to obtain word vectors containing ambiguous words and contextual knowledge, and calculate the cross similarity of the word vector, the high mean similarity is regarded as the correct meaning of the ambiguous word. The effectiveness of the proposed method is verified by comparative experiments on the SEVAL-2007 Task# 5 dataset.

1. Introduction

Word sense disambiguation mainly solves the problem of ambiguity in the expression of polysemous words in sentences. It judges the specific meanings of ambiguous words based on contextual knowledge, so that the computer can better understand the intention expressed by the user. Word sense disambiguation technology has a direct impact on information retrieval, machine translation, text classification, speech processing and question answering system, so it plays a very critical role in the field of natural language processing [1].

The research of word sense disambiguation technology can be traced back to the 1950s and 1960s. Its task is to let the computer automatically determine the meaning of polysemous words in the corresponding language environment. However, due to the limitations of computer performance and data sources, part of the research only stayed at the theoretical level. In the 1990s, in order to solve the problem of word sense disambiguation, researchers manually sorted out a large number of data sets, such as Wikipedia, babelnet, Gigaword, Sogou corpus and HowNet [2]. At present, word sense
disambiguation technology is mainly divided into two ways: supervised word sense disambiguation model (Bayesian model, decision tree model, maximum entropy model, dictionary based leks algorithm and linguistic model, etc.) and unsupervised word sense disambiguation model (graph based model embedding method, K-means clustering method and LDA topic model, etc.)[3, 4]. In order to automatically build a high-quality knowledge base and accurately select the relevant words of ambiguous word, Lu W [5] proposed word sense disambiguation based on dependency constraint knowledge. The method has achieved the best performance among unsupervised and knowledge-based methods in SemEval dataset. Meng F [6] proposed a graph and word similarity method for word sense disambiguation, this method can effectively use English knowledge resources and obtain a disambiguation accuracy rate of 0.451 on the SemEval-2007 task#5 dataset, which can improve the accuracy of Chinese word sense disambiguation. Kwon S et al. [7] introduced a novel knowledge-based word sense disambiguation (WSD) system, the experimental results demonstrate that the suggested methods significantly enhance the baseline WSD performance in all corpora. Li W [8] proposed a hybrid context based topic model with an adaptive context window length for word sense disambiguation in document representation, not only can divide various senses for each polysemous word, but also preserve the differences between synonyms. Although the above research has achieved good results, it has made a breakthrough in word sense disambiguation technology. However, word sense disambiguation is limited by domain knowledge and dictionary size, and most of the methods are over dependent on the external knowledge base.

2. Data Source and Method

2.1. Data Source
The dataset used in this paper is from the international semantic evaluation dataset SEMEVAL-2007, and the task#5 of this dataset is the Chinese and English word sense disambiguation task [9]. The evaluation method used is accuracy rate. The calculation formula is shown in (1)

\[
\text{Accuracy} = \frac{\text{Forecast the number of correct samples}}{\text{The total number of sentences in the sample of the ambiguous word}}
\]  

(1)

2.2. Method

2.2.1. Dependency parsing
Dependency parsing is one of the key techniques in the category of natural language processing. It was first proposed by French linguist L. Thesniere [10]. Its task is to analyze the sentence structure of the ambiguous sentence according to the grammar rules, and then determine the role and dependency of the part of speech and phrase in the sentence. In this paper, with the help of the natural language toolkit PYLTP developed by the research center of social computing and information retrieval of Harbin Institute of technology, we use dependency parsing to analyze sentence structure. In this paper, the dependency parser is used to analyze the sentence structure with the help of the natural language toolkit PYLTP developed by the Social Computing and Information Retrieval Research Center of HIT Institute of Technology. It includes CWS. Model, POS. Model, NER. Model, Parser. Model and STR_DATA. This paper uses Parser. Model to analyze sentence structure and extract dependency relations, mainly completing the following two aspects. On the one hand, we use the parser. Model to give a formal definition of the grammatical structure of legitimate sentences in the language and determine the grammatical system of the language. On the other hand, according to the given grammatical system, analyzes the syntactic units contained in the sentence and the relationship between these syntactic units, and obtains the contextual dependency diagram of the ambiguous sentence. The partial dependency descriptors are shown in Table 1.
Table 1. Partial dependency specifier

| symbol | Dependencies |
|--------|--------------|
| ADV    | adverbial    |
| HED    | head         |
| VOB    | verb-object  |
| DOB    | double object|
| CNJ    | conjunction  |
| POB    | preposition-object|
| SBV    | subject-verb |
| COO    | coordinate   |

2.2.2. Construction of graph model

At present, graph model is widely used in disambiguation. The relationship among root node, part of speech and form obtained from the above dependency syntax analysis is represented in the form of triple to generate the network graph [11]. The details are as follows: first, extract the dependent parent node ID, extract the dependency relationship, and match the dependent parent node words. Then, based on the obtained dependencies, build a graph model.

2.2.3. Word2Vec trains word vectors

In natural language processing, it is necessary to convert text data into digital data for calculation. The conversion methods include one-hot, space vector model VSM and so on. However, the above methods will cause the data latitude is too large, and does not contain semantic information. In 2013, the Google team proposed word vectors to represent textual data [12]. The model is based on the hypothesis that if you want to know the meaning of a word, you can determine what it means by looking at the words it is used with. According to this hypothesis, a distributed model is defined. First, enter a central word $w$, $w$ represents the position of a word. A single layer neural network is used to predict several words $w_{t-2}$, $w_{t-1}$ around it. The goal is to train word vectors with semantics. The objective function of training is shown in formula (2).

$$y = \frac{1}{T} \sum_{t=1}^{T} \sum_{i=0}^{T} \log(p(w_{t+j}) / w_t)$$  \hspace{1cm} (2)$$

The probability density is obtained by softmax function, as shown in formula (3).

$$p(o/c) = \frac{\exp(u_0^T v_c)}{\sum_{w=1}^{w} \exp(u_w^T v_c)}$$  \hspace{1cm} (3)$$

Where $u_0$ represents the output vector, $v_c$ represents the input vector, the higher the dot product of $u_0$ and $v_c$, the higher the probability. Due to the large number of words, the calculation speed is slow, so the negative sampling method is used to speed up the training and calculation of the model. Specific formulas are shown in (4) and (5).

$$P(D=1/w,c,\theta) = \frac{1}{1+e(-v_c^T v_w)}$$ \hspace{1cm} (4)$$

$$y = \log \sigma(u_{c-w+j}^T \cdot v_c) + \sum_{k=1}^{i} \log \sigma(-u_k^T \cdot v_c) \hspace{1cm} (u_k / k = 1,...,k)$$ \hspace{1cm} (5)$$

This paper selects the Chinese corpus of Wikipedia to train the word2vec model, which has a large quantity, high quality and wide fields. The training process is shown in Figure 2.
Obtain the Chinese corpus
Convert the XML Wiki data to Text format
Conversion of simplified and traditional Chinese
Remove English and spaces
Chinese word segmentation (Jieba word segmentation)

Figure 1. Process of word vector training model

2.2.4. Space vector model and similarity calculation
The word2vec model is used to train Wikipedia database to get the word vector with context knowledge and ambiguous words. Vector space model (VSM) [13] can calculate the cross similarity of the word vector, take the mean value, and judge the high score is the correct meaning of ambiguous words. In this way, the computability and operability of text content are realized, and the model is also one of the most mature and widely used models.

Suppose that an example sentence containing an ambiguous word is represented by D (document). Firstly, the existing natural language processing technology is used to segment the user's description, remove the stop words and set the shortest path to extract the context knowledge. Feature item refers to the basic language unit that can be described by the ambiguous sentence in the document. It is represented by T (term), and is generally composed of the background knowledge of the context. Ambitious sentences and contextual background knowledge are represented by the set D (T1, T2, T3...Tn) where Tk(1 ≤ k ≤ n) is the feature word of contextual knowledge. The process of generating vector space model is shown in Figure2.

Figure 2. Process of generating vector space model

Using the space vector model to transform text content into vector can be expressed as follows: for each word in context knowledge m, Use \( W_{im} \) to represent the weight of the i-th word in m, namely \( \bar{m} = (W_{1,m}, W_{2,m}, \cdots, W_{i,m}) \), in the same way, use \( W_{in} \) to represent the weight of the i-th word in the meaning text \( n \) of the ambiguous word is \( \bar{n} = (W_{1,n}, W_{2,n}, \cdots, W_{n,n}) \). Then the similarity between \( \bar{m} \) and \( \bar{n} \) is calculated by cosine theorem as the similarity between the two texts. The weight W of context knowledge is trained according to the Word2Vec model. In this paper, \( \bar{m} \) represents the context knowledge of ambiguous sentences, and \( \bar{n} \) represents the meaning of ambiguous words. The similarity calculation is shown in formula (6).

\[
\sin(m,n) = \cos \theta = \frac{\bar{m} \cdot \bar{n}}{||\bar{m}|| ||\bar{n}||} = \frac{\sum_{i=1}^{l} W_{im} \times W_{in}}{\sqrt{\sum_{i=1}^{l} W_{im}^2 \times \sum_{i=1}^{l} W_{in}^2}}
\] (6)
2.3. **Overall design idea**

This system is mainly divided into three modules: data preprocessing, graph model construction and word vector training. The data preprocessing module is mainly to process the data of SemEval-2007 task#5. Firstly, the HTML data is converted into text data, and then the Jieba word segmentation tool is used for word segmentation, removal of punctuation marks, removal of stop words and other preprocessing, so as to obtain ambiguous words in ambiguous sentences. In the construction of graph model, firstly, PYLTP tool is used to analyze dependency syntax, extract dependency, construct graph model for dependency parent node [14], determine the distance between context background knowledge and ambiguous words, determine the threshold of minimum path, and then obtain context background knowledge. The training word vector module uses word2vec model to train more than 910000 Wikipedia data, and gets the word vector of each ambiguous word and context. Some ambiguous words and contextual knowledge do not appear in the Wikipedia data. In this paper, the word vector of this type is set to UNK. If the initial value is 0, UNK does not participate in the training, so that UNK shares the same semantic information. Finally, the cross-weighted similarity of the three modules is calculated to take the mean value, and the highest score is determined as the correct meaning of the ambiguous word. The overall block diagram of the system is shown in Figure 3.

![Diagram of System Design](image)

**Figure 3. Overall design idea of the system**

3. **Results & Discussion**

3.1. **Comparative approach**

(1) **Window-based approach (WIN)**

Based on the method of window movement, the context knowledge of ambiguous words is selected automatically according to the size of the set window.

(2) **HowGraph approach**

HowGraph uses examples of ambiguous words in HowNet dictionary to construct graph model for disambiguation [15], and other parameter settings are the same as those in this paper.

(3) **Our algorithm**

This paper proposes a word sense disambiguation method based on graph model and word2vec.

3.2. **Analysis of experimental results**

This paper made experimental comparison based on the above three methods, and the comparative results are shown in Table 2.
Table 2. Comparison of three disambiguation methods

|               | Win  | HowGraph | Our algorithm |
|---------------|------|----------|---------------|
| accuracy %    | 48.2 | 51.8     | 56.2          |
| percentage increase % | 8    | 4.4      |               |

The average disambiguation accuracy of the algorithm in this paper is 0.562, which is 4.4% higher than the HowGraph method and 8% higher than the Win method. The window-based method selects contextual knowledge according to the size of the setting window. Obviously, the words on both sides of the window can’t represent ambiguous words well, which will have adverse effects on the disambiguation results. This paper uses graph model to solve the problem by setting the minimum path threshold. HowGraph method relies too much on the example sentences of ambiguous words in HowNet knowledge base, and the limitation of HowNet dictionary has a negative effect on the disambiguation result. The word vectors trained by Word2vec model in this paper have good spatial semantic information, so the accuracy of word disambiguation is improved obviously in word sense disambiguation. The results of disambiguation of some words are shown in Figure 4. Experiments show that the proposed method has a significant improvement in the accuracy of disambiguation.

3.3. Parameter selection

The influence of the threshold length of the shortest path on the average disambiguation accuracy is shown in Figure 5. When Length is 2, both methods can get the best disambiguation result, when the length is other, the disambiguation effect is not ideal. Because when the length is too small, it can’t express the meaning of ambiguous words well. When the length value is too large, irrelevant contextual background knowledge will be introduced to form interference factors, resulting in the decline of disambiguation accuracy. Therefore, the threshold length of the shortest path adopted in this algorithm is 2.

4. Conclusion

This paper proposes a word sense disambiguation method which combines graph model with word vector. Experimental results show that this method can improve the accuracy of disambiguation obviously. The quality of context knowledge extraction affects the accuracy of disambiguation. Therefore, the next step of research focuses on how to better extract context knowledge to effectively represent the meaning of ambiguous words.
Acknowledgments

This work was supported in part by the Inner Mongolia Autonomous Region's major science and technology: AI application technology and product research and development; application research and demonstration in modern pastures, Fund Number 2019ZD025. It was also supported by the Inner Mongolia Natural Science Foundation on livestock big data, livestock grazing trajectory mining, and optimized production decision-making research Fund Number 2019MS06021. It was supported by Inner Mongolia Discipline Inspection and Supervision Big Data Laboratory Open Project Fund: Research on Inner-Party Laws and Regulations Information Retrieval Model and Technology Based on Word Sense Disambiguation Technology, Fund Number IMDBD2020019.

References

[1] Rahman N, Borah B. Improvement of query-based text summarization using word sense disambiguation[J]. Complex & Intelligent Systems, 2020, 6(1): 75-85.
[2] Chen Y, Luo Z. A Word Representation Method Based on Hownet[J]. Beijing Da Xue Xue Bao, 2019, 55(1): 22-28.
[3] Wang S, Zong C. A Dual-LDA Method on Chinese Word Sense Representation and Induction[J]. Chinese Journal of Computers. 2016, 39(8): 1652-1666.
[4] Yang Y, Xiao L I, Yating Y. Unsupervised word sense disambiguation for Uyghur based on LDA topic model[J]. Journal of Xiamen University, Natural Science, 2020, 59(2): 198-205.
[5] Lu W. Word sense disambiguation based on dependency constraint knowledge[J]. Cluster Computing, 2019, 22(3): 7549-7557.
[6] Meng F. Graph and Word Similarity for Word Sense Disambiguation[C]//2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI). IEEE, 2020: 1114-1118.
[7] Kwon S, Oh D, Ko Y. Word sense disambiguation based on context selection using knowledge-based word similarity[J]. Information Processing & Management, 2021, 58(4): 102511.
[8] Li W, Suzuki E. Adaptive and hybrid context-aware fine-grained word sense disambiguation in topic modeling based document representation[J]. Information Processing & Management, 2021, 58(4): 102592.
[9] Mazzoleni M, Maroni G, Previdi F. Unsupervised Learning of Fundamental Emotional States via Word Embeddings[M]. NEW YORK: IEEE, 2017, 31-36.
[10] Jana A, Halder R, Kalahasti A, et al. Extending Abstract Interpretation to Dependency Analysis of Database Applications[J]. IEEE TRANSACTIONS ON SOFTWARE ENGINEERING. 2020, 46(5): 463-494.
[11] Lu W, Meng F, Wang S, et al. Graph-Based Chinese Word Sense Disambiguation with Multi-Knowledge Integration[J]. Computers, Materials & Continua. 2019, 61(1): 197-212.
[12] Zhou S, Jiang L, Lin S, et al. Characteristic representation method of document based on Word2vector[J]. Journal of Chongqing University of Posts and Telecommunications. Natural Science Edition. 2018,30(2): 272-279.
[13] Sang J, Pang S, Zha Y, et al. Design and analysis of a general vector space model for data classification in Internet of Things[J]. EURASIP JOURNAL ON WIRELESS COMMUNICATIONS AND NETWORKING. 2019(1): 263.
[14] Liu D, Zhang L, Shao Y, et al. Leverage External Knowledge and Self-attention for Chinese Semantic Dependency Graph Parsing[J]. INTELLIGENT AUTOMATION AND SOFT COMPUTING, 2021, 28(2): 447-458.
[15] Nie H, Zhou J, Wang H, et al. Word Similarity Computing Based on HowNet and Synonymy Thesaurus[C]//Proceedings of SAI Intelligent Systems Conference. Springer, Cham, 2019: 292-305.