Journal of Information Hiding and Multimedia Signal Processing ©2018 ISSN 2073-4212
Ubiquitous International Volume **9**, Number **2**, March **2018**

A Distributional Semantics Based Syntagmatic Association Measuring Method

Xiao-Xia Liu

School of Computer Science and Technology
Dalian University of Technology
No.2, Linggong Road, Ganjingzi Zone, Dalian, 116024, China
liuxxivy@mail.dlut.edu.cn

De-Gen Huang

School of Computer Science and Technology
Dalian University of Technology
No.2, Linggong Road, Ganjingzi Zone, Dalian, 116024, China
*:Corresponding author,huangdg@dlut.edu.cn

Shun-Miao Zhang

School of Information Science and Engineering
Fujian University of Technology
No.3, Xueyuan Road, University Town, Minhou, Fuzhou, 350118, China
zshunmiao@mail.dlut.edu.cn

Received May, 2017; revised October, 2017

Abstract. *Two kinds of relations exist between words, syntagmatic and paradigmatic. Word embedding as a state-of-the-art model of distributional semantics has been used to discover the paradigmatic relations between words and has been widely used in natural language processing tasks. Based on a hypothesis that at sentence level, except for words in paradigmatic relations, two words in certain syntagmatic relation are more similar than those not in any syntagmatic relations, we propose to discover words in syntagmatic relations in a sentence using word embedding based similarity computation. The experiments prove that word embedding based similarity between words in syntagmatic relations is higher than that between words not in any syntagmatic relations. And word embedding based method is competitive to the best measures in literature and can be a good complement to those measures. This discover can be conducive to many syntagmatic related natural language processing tasks such as parsing, text generation, machine translation, collocation extraction and multi-word expression recognition. Further experiment in collocation extraction shows that the proposed word embedding based association measure is effective in filtering the noisy collocation candidates at sentence level and it outperforms the existing well-known association measures in all precision, recall and F-measure.*
Keywords: Syntagmatic relations, Association measure, Word embedding.

1. **Introduction.** There are two fundamental types of relations between words according to Ferdinand de Saussure who is the father of modern linguistics: syntagmatic and paradigmatic relations [1]. Two words are in a syntagmatic relation if they co-occur more frequently than expected from chance and if they have different grammatical roles in the sentences in which they occur. Typical examples are word pairs like hat-wear and

rain-heavy. There is a paradigmatic relation between two words if they can substitute for one another in a sentence without affecting the grammaticality or acceptability of the sentence. Typical examples are synonyms or antonyms such as heavy-strong or strong-weak.

The ways to find these two relations are called association computation. Paradigmatic associations are words with high semantic similarity. Traditionally, the semantic similarities between words can be computed by simple vector comparisons based on the distributional hypothesis of distributional semantics. Syntagmatic associations are words that frequently co-occur. Traditionally, way to extract them from texts is to look for word pairs whose co-occurrence is significantly larger than chance. To test this significance, many association measures can be used, and more than 80 such association measures are mentioned in literature [2]-[3]. But we think that words in syntagmatic relations also always occur together in similar contexts. In a sentence, except those words in paradigmatic relations, words in syntagmatic relations should be closely related to each other. If this is the case, then distributional semantic model can also be used to predict words in syntagmatic relations.

In the past, distributional semantic models (DSMs for short) build semantic representations dynamically with high dimensional vector spaces after a statistical analysis of the contexts in which words occur and DSMs are thus regarded as a promising technique for solving the lexical acquisition bottleneck by unsupervised learning, and it is said that this kind of distributed representation provides a plausible, robust and flexible architecture for processing semantic information. With the advent of word embedding, this representation has been changed from high dimensional to comparatively low dimensional vector spaces, which makes it even more quick and easy to process semantic information on big data.

The most popular word embedding model is introduced in [4] and available as the word2vec toolkit. The word2vec method is the most frequently cited method for capturing meaningful semantic relations between words from corpus. It has the advantage of not requiring any tagging while training. The prevailing view is, however, that it is able to capture semantic similarity but incapable of capturing complex compositional semantics and so is virtually useless for most purposes. But we believe that since syntagmatic relations are words co-occurring more often than by chance in the same contexts, word embedding can also capture the syntagmatic relations between words. The literature has already proved that word embedding is effective in discovering the paradigmatic relations by similarity computation. The aim of this study is thus to prove that word embedding is effective in discovering the syntagmatic relations between words, too. In this context, our primary research questions are: (1) Can word embedding be used to discover the syntagmatic relations between words in a sentence? (2) Can different parameter values in training word embedding influence the discovering performance? (3) Is word embedding based method superior to conventional association measures? (4) Is this method language independent? (5) Is the method useful in practical applications?

In Part 2, we briefly review the best association measures used in syntagmatic relation detecting and the application of word embedding in word relation detecting. In Part 3, we mainly answer the first four research questions mentioned above by experiments and we explore the usefulness of the proposed method in practical task of collocation extraction in Part 4. Part 5 concludes the whole work.

2. **Literature Review.** Traditional ways to measure the syntagmatic relations between words are the so called association measures. There are more than eighty different association measures reported [2]-[3]. It is said none of them can beat the others completely

but each can be useful in measuring certain types of syntagmatic relations. So, recently, researchers, especially those who are interested in machine learning methods to do the research [5]-[10], tend to combine several different association measures in the discovering of the syntagmatic relations. Among all the association measures, the following several are reported the best and the most commonly used:

First, mutual information (MI for short). The formula is as the following:

$$MI(x, y) = \log_2 \frac{p(x, y)}{p(x)p(y)}. \tag{1}$$

In the formula, $p(x)$, $p(y)$, $p(x, y)$ are probabilities of occurrence of words x, y, and bigram x, y respectively. [11] uses point-wise mutual information to measure the correlation between words. But MI favours rare events [12], in fact, [11] suggests using a frequency threshold of five. But on large frequency-filtered data, MI was shown to lead to competitive results [2]. Many studies use mutual information to evaluate the syntagmatic relations [13]-[17].

Secondly, Dice coefficient [18], which is defined as:

$$DICE(x, y) = \frac{2f(x, y)}{f(x) + f(y)}, \tag{2}$$

where $f(x)$, $f(y)$, and $f(x, y)$ are frequencies of words x, y, and bigram x, y respectively. Dice coefficient is a method that can provide optimal results in a given setting, for instance, in discovering A-N (adjective and noun pattern) in German [5] and in the collocation extraction tool Xtract [19].

Thirdly, the chi-square test (χ^2), which is defined as:

$$\chi^2 = \sum_{i,j} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}, \tag{3}$$

where O_{ij} and E_{ij} are the observed and expected frequencies in a contingency table. The chi-square test overcomes the normal distribution problem as it makes no assumptions about the data, but it still over-emphasizes rare events and also over-emphasizes common events [20]. Moreover, it is inaccurate when the sample size is small [21]. It has been used in [5].

Fourthly, Log-likelihood Ratio (LLR for short) is an often used association measure for it is argued to be appropriate to both rare and common phenomena, to both large and small text samples [22]. The formula is:

$$LLR = 2 \sum_{i,j} O_{ij} \log \frac{O_{ij}}{E_{ij}}, \tag{4}$$

where O_{ij} and E_{ij} are the observed and expected frequencies in a contingency table. Log-likelihood Ratio is generally considered as the most appropriate measure for collocation extraction [23], so it is used in many works such as [5], [22], [24]-[26].

The state-of-the-art distributional semantic model, in the form of word embedding, has demonstrate their utility in a wide range of NLP tasks, including identifying various morphosyntactic and semantic relations [27], dependency parsing [28], sentiment analysis [29], named-entity recognition [30]-[31] , and machine translation [32].

Word embedding based similarity computation is the most commonly used DSM in which it is believed that the vectors of two semantically related words are related by linear transformation. [27] applies it into the word-based translation. [33] applies this to the learning of hyponym-hypernym relations in Chinese. [34] models the similarity between standard language and non-standard language. [35] uses a simple word embedding based model to discover lexical substitutions for certain word. [36] uses word embedding to compute the compositionality of multi-word expressions. [37] uses a transformation matrix learned from a few collocation examples to discover collocates for untrained head words. [38] uses word embedding to discover similar words and to recognize noun compounds by using the non-substitution of collocations.

The association measures in literature only consider the co-occurrence of the words but cannot capture the semantic and syntactic relations between words. With the advent of distributional semantics, the semantic and syntactic information of a word can be captured in vector representation. But as the literature shows, the use of distributional semantics, in the form of word embedding, only focuses on the computation of paradigmatic association, and it is seldom used in the computation of syntagmatic association. But theoretically, in a sentence, except for the paradigmatic relations, the distance between words in syntagmatic relations must be shorter than the distance between words without any relation. At the same time, at sentence level, few words are in paradigmatic relations, and many of them are in syntagmatic relations. So it is possible to use word embedding based similarity computation to discover the syntagmatic relation at sentence level.

3. **Word Embedding Based Method to Discover Syntagmatic Relations.** Word embedding is low-dimensional vector representations of word types. In literature, word embedding has been used to discover the paradigmatic relation by computing the similarity between words but so far not to discover the other relation between words, syntagmatic relation. Intuitively, the distance between words in syntagmatic relations in a sentence is shorter than words not in any syntagmatic relations in the sentence. For example, In EXAMPLE 1, we compute word embedding based similarity between words in the sentence and rank the similarities. For space limitation, we only put the most similar ten pairs in TABLE 1. The most similar word pair is "海军(the navy)" and "陆军(the army)", which is a pair of synonyms. Those ranking after from number two to nine are either parts of a syntagmatic relation or syntagmatic relation, but those after rank number nine are seldom in any relations. From this example, we can see that it is possible to discover the syntagmatic relations in the sentence by using word embedding based similarity computing.

EXAMPLE 1: 海军被作为陆军配属军种，不具备独立执行作战任务的能力。(As a subsidiary force to the army, the navy cannot shoulder independent combat mission.)

So we believe word embedding based similarity computing can be used to discover the syntagmatic relations in a sentence.

3.1. **The Proposed Method.** To use word embedding based similarity computing to discover the syntagmatic relations in a sentence, we use similarity rankings. Given a head word H, and a sentence $S(S = w_1, w_2, \cdots, w_n)$ that includes the head word, the aim of our method is to discover the words in the sentence that are in syntagmatic relations with the head word H. For each candidate word W ($W \in S$ and $W \neq H$) , we have to compute the similarities between words in three lists:

List 1: all word pairs in the sentence;

List 2: word pairs in the sentence that include the headword H;

List 3: word pairs in the sentence that include the candidate word W.

From these three lists, we will get three ranks for the candidate word pair (H, W)

TABLE 1. The top ten similar pairs

Word A	Word B	Sim	Relations
海军(the navy)	陆军(the army)	0.7801	paradigmatic
海军(the navy)	作战(combat)	0.6562	syntagmatic
陆军(the army)	作战(combat)	0.5645	syntagmatic
执行(carry out)	作战(combat)	0.4105	part of syntagmatic
独立(independent)	作战(combat)	0.4079	part of syntagmatic
作战(combat)	任务(mission)	0.3788	syntagmatic
执行(carry out)	任务(mission)	0.3712	syntagmatic
独立(independent)	执行(carry out)	0.3541	syntagmatic
具备(possess)	能力(ability)	0.3207	syntagmatic
具备(possess)	执行(carry out)	0.2946	none

respectively, which are designated as Rank 1, Rank 2 and Rank 3 and act as the criteria to decide whether the candidate word pair is in syntagmatic relation or not. Specifically, in our experiments, only if one of the rankings of the candidate word pair (H, W) ranks before (and including) 5, the candidate word pair is regarded as in certain syntagmatic relation. That is:

$$Syntagmatic(H,W) = \begin{cases} True & \text{if Rank } 1 \leqslant 5 \text{ or Rank } 2 \leqslant 5 \text{ or Rank } 3 \leqslant 5 \\ False & \text{else} \end{cases}$$

To choose these three rankings is based on the following hypotheses: the high rank in first ranking reveals that the candidate and the head word are more closely related compared with all the other candidates in the sentence; the high rank of the second ranking tells that the head word has closer association with the candidate word than the other words in the sentence and the high rank of the third ranking tells that the candidate word is more closely associated with the head word compared with other words in the sentence.

3.2. **Experimental Setup.** In order to test whether word embedding based method is useful in discovering the syntagmatic relations between words in a sentence, the training of word embedding is necessary. To train word embedding, we use the Python version gensim word2vec [39].

In order to test the language independence of the proposed method, a self-compiled parallel corpus including Chinese and English serves as the training corpus. The detailed information is shown in TABLE 2. The Chinese corpus was first segmented by using an available tool [40]. The result word embedding includes 57,979 Chinese words and 141,966 English words. The testing corpus is a mini parallel corpus of 100 sentences. All the syntagmatic relations are manually annotated. In annotating, we ask two linguistic experts to annotate the syntagmatic relations in the testing sentences independently, only those relations agreed by the two are accepted. In our experiments, we only choose several representative words to test. The representatives are chosen according to two criteria. On the one hand, we choose noun and verb as representatives since they are strong in creating relations, on the other hand, we choose the most frequent word in the test corpus. The more frequent the word is, the more comprehensive relations can be studied. So for Chinese part, we choose “能力 (ability)” representing nouns and “提高 (improve)” representing verbs. For English part, we only choose the most frequent noun “ability” as the representative. The evaluation work is conducted through three well-known indicators including precision, recall and F_1 measure.

Table 2. Training corpus

Languages	Sentences	Tokens	Types
Chinese	457,899	8,388,455	57,979
English	457,899	7,971,065	158,998

3.3. **Setting the Best Threshold for Each Key Parameter in Word Embedding Training.** Word embedding in this study is trained by using the python version of word2vec package gensim word2vec. There are several key parameters in this package: the algorithm, the size of the trained vectors, the window size for training the vectors, and the iteration times in the training. For the parameter algorithm, there are two choices: CBOW and skip-gram. The default values for the size of vector, window size and iteration times are 100, 5, and 5 respectively, and we can set different values for each parameter. In order to test the influence of different settings of each parameter, we change only one parameter threshold every time and set default values for the other parameters. We evaluate the influence by using the discovering F_1 measure of the syntagmatically related words.

We first decide a better algorithm to train task-oriented word embedding. With other parameters set as the default value, we change the algorithm to train different word embedding and use the trained different word embedding in the syntagmatic relation discovering. The result comparison (FIGURE 1) tells that in this task, skip-gram performs a little better.

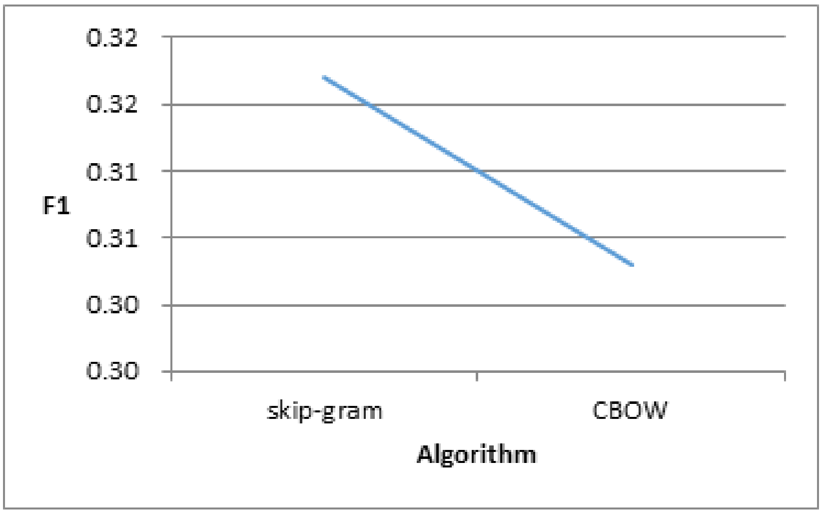

Figure 1. Influence of different algorithms

After choosing a better algorithm, it's time to decide the best values for other parameters. This experiment is to decide the best window size. The default window size is 5. That means when training word embedding five words before and after the word in study are chosen as the window. We choose 2, 8, 10 respectively as the window size. According to the experimental results (FIGURE 2), the optimal window size is 10, and the word embedding trained when the window size is 2 is better than when the window size is enlarged but does not surpass 10. That is because most of the syntagmatically related words are adjacent to each other, so the window size 2 can achieve comparatively good results. Since the average sentence length of the corpus is 18, that means when the window size is 10, the training of word embedding almost considers the whole sentence, so the optimal result is achieved.

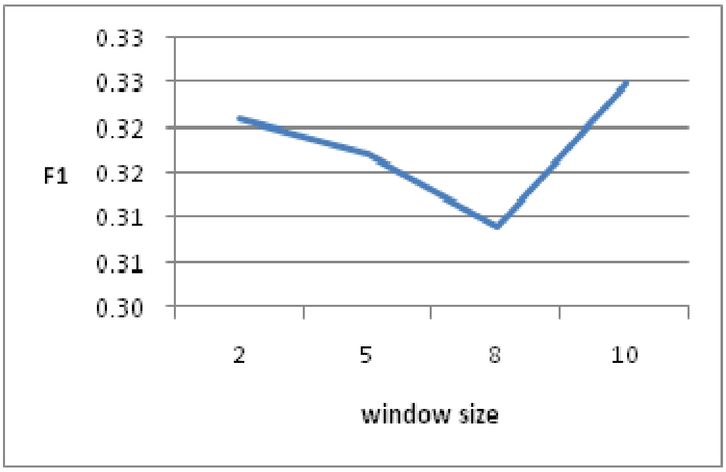

FIGURE 2. Influence of window size

Another key parameter in the training of word embedding is the size of the vectors. In the experiment, we choose 100, 200, 300, 400, 500, 600, 700, 800, 900 and 1000 to train the word embedding respectively and find that when the vector size is smaller than 600 (including 600), the larger the vector size, the better the results. But when the vector size is larger than 600, the influence is not always positive, and there are ups and downs in the influence curve (FIGURE 3).

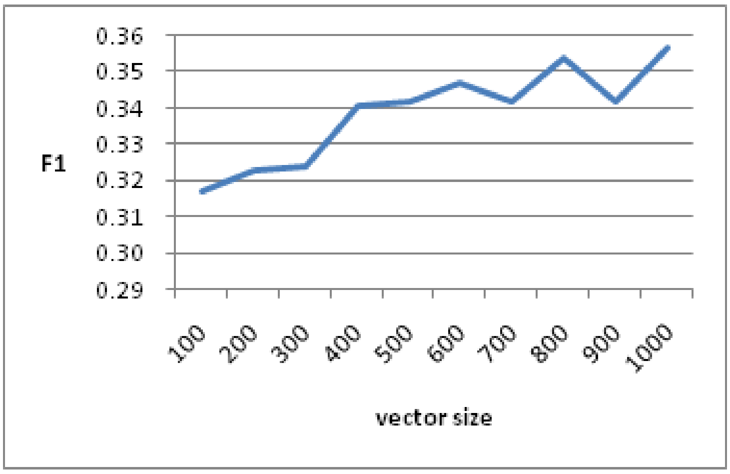

FIGURE 3. Influence of vector size

The last key parameter is the training iteration time. Since the word embedding training corpus is comparatively small, the iteration times are set to enlarge by ten. The experimental results (FIGURE 4) show that from 10 to 20 the result changes dramatically but after 20, although enlarged vector size brings better results, the improvement is not significant.

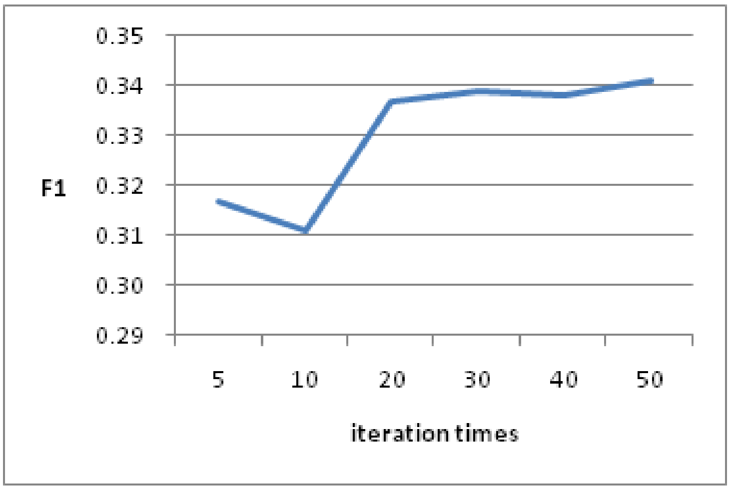

FIGURE 4. Influence of iteration times

Combining the above experimental results, we get the optimal parameter value set for the task of measuring the syntagmatically related words. They include the skip-gram model as the algorithm, window size 10, vector size 600 and iteration time 30.

3.4. **Comparison with Other Association Measures.** Is word embedding based similarity computing competitive to the state-of-the-art methods? What's the difference of the proposed method compared to other methods in the literature? In literature, the syntagmatically related words are discovered by using different association measures. The most commonly used and also universally acknowledged as the most suitable association measures for recognizing syntagmatically related words are Dice, χ^2, mutual information and log likelihood ratio. These methods are often used to extract syntagmatic relations in the large text corpus. In order to compare the results at sentence level, we rank the word pairs in the sentence according to different association measures and in order to be comparative with the proposed method, we choose those ranking before (and including) five as correctly discovered syntagmatic relations. The evaluation results of different methods including our proposed method are presented in TABLE 3.

Word embedding in the proposed method is trained by using the best parameter

TABLE 3. Comparison of several association measures in Chinese

Methods	Correct/All	P	R	F	C
Our Method	171/745	0.230	0.679	0.343	/
DICE	116/635	0.183	0.459	0.261	72
χ^2	179/635	0.282	0.708	0.403	33
MI	180/635	0.284	0.712	0.405	36
LLR	168/635	0.265	0.664	0.378	36

values mentioned in 3.3. We evaluate the proposed method in two perspectives: first, we compare the proposed method with the other commonly used measures by precision, recall and F_1 measure. Secondly, we evaluate the proposed method from the perspective of how much it can complement the other measures. In TABLE 3 the second column includes all the extracted syntagmatic relations (Correct in the table) and all the correctly extracted ones (All in the table) of different methods. The third, fourth and fifth columns are precision, recall and F_1 measure of different methods. And the last column shows the complement of our method to other methods. Experimental results show that compared with the commonly used association measures, the proposed method is not the best, but it can complement the others. For example, in the correctly discovered syntagmatic relations, the proposed method can discover 72 syntagmatic relations that Dice cannot, 33, χ^2 cannot, 36, MI and LLR cannot. All these experimental results tell us that the word embedding based method is effective in measuring the syntagmatic relations and can be a good complement to the existing methods.

Further study of results got by the proposed method show that almost all types of syntagmatic relations can be extracted such as “具备能力 (have the ability)” in the form of “verb + noun”, “能力降低 (ability to reduce)” in the form of “noun + verb”, “重要能力 (important ability)” in the form of “adjective + noun”, “能力差 (ability is poor)” in the form of “noun + adjective” and “学习能力(learning ability)” in the form of “noun + noun”, etc. The study also shows that almost all the function words in the sentence are selected as in the syntagmatic relations with the head word. This may be because function words are frequently used and they exist in almost all the sentences in the corpus. Thus, the similarity computation will find them all the time. Take the function word “的 (of)” as an example. As long as there’s “的 (of)” in the sentence, the similarity ranking of the word “的 (of)” and the headword “能力(ability)” ranks mainly in the first or second place.

3.5. **Cross Linguistic Applicability of the Proposed Method.** The above experiments are all on the measurement of syntagmatic relations in Chinese, then is the method cross linguistically applicable? In this section, we conduct a similar study in English. We choose the English part of our parallel training and testing corpus as the training and testing corpus in this experiment. The focus word is the most frequent word in the testing set “ability”. The experimental results are shown in TABLE 4. There are altogether 75 syntagmatic relations of “ability” in the testing corpus, the proposed method can discover 52 correctly, numbers of correctly discovered relations by the other methods in the literature are: 46 for DICE, 61 for χ^2, 56 for MI, and 42 for LLR. There are no significant differences between the proposed method and the methods in literature, and further analysis shows that the proposed method can discover some different relations from the methods in the literature. Among all the 52 correctly recognized relations, 21 cannot be recognized by Dice, 6 cannot be recognized by χ^2, 8 cannot be recognized by MI and 23 cannot be recognized by LLR. The experimental discovery further proves that our proposed method can be a good complement to the methods in the literature. The discovering here is consistent to the experimental results in the Chinese part. It proves that the proposed method is not dependent on language, and can be used to discover any syntagmatic relations in any language.

Table 4. Comparison of several association measures in English

Methods	Correct/All	P	R	F	C
Our Method	52/212	0.245	0.693	0.362	/
DICE	46/196	0.238	0.613	0.340	21
χ^2	61/196	0.311	0.813	0.450	6
MI	56/196	0.286	0.747	0.413	8
LLR	42/196	0.214	0.560	0.310	23

4. **Application of Proposed Method in Collocation Extraction.** Measurement of syntagmatic relations can be used in many syntagmatic relation related fields, such as parsing, collocation extraction, and multi-word expression recognition. The experiment in part 3 shows that word embedding based measurement of syntagmatic relations is effective, competitive and complementary, but what is the practical effect of this measurement in a specific task? In order to answer this question, we conduct a pilot study on the task of collocation extraction. The training and testing corpora for collocation extraction are the same as in part 3.2. The method to extract collocations follows the traditional two phase method: get the candidates in the first phase and filter the noises in the second. In the first phase, dependency relations in a sentence are regarded simply as the candidates. We first conduct dependency parsing on the test corpus by using Stanford CoreNLP package [41], then extract each dependency relation as the candidate of our collocation extraction task. In the filtering phase, in order to compare our proposed method with those in literature, we use the proposed word embedding based measurement of syntagmatic relations and those mentioned in part 3.3.

To determine whether a candidate is a real collocation or not, in our method, three rankings of the candidate in study are considered. For a candidate collocating word W to the head word H, Rank 1, the ranking of similarity between H and W among all the candidates in the sentence, Rank 2, the ranking of similarity between H and W among all the candidates that include word H in the sentence, and Rank 3, the ranking of similarity between H and W among all the candidates that include word W in the sentence. All these rankings are computed according to the word embedding based similarity computation. So long as one ranking of the candidate is at first in the three rankings, we regard this candidate as a real collocation.

For other association measures, since they are usually used to measure the association between words in the whole corpus level, for the measuring at sentence level, we simply use the ranking method. In order to be comparable with our method, only those ranks at first are regarded as the real collocations.

The extraction results of different filtering methods are in TABLE 5. The results show that this word embedding based method is effective in filtering the collocation candidates and it outperforms the methods using the traditional association measures both in precision, recall and F_1 measure.

Table 5. Comparison of proposed method with traditional methods in collocation extraction

Methods	P	R	F
Our Method	0.643	0.465	0.540
DICE	0.606	0.387	0.472
χ^2	0.606	0.387	0.472
MI	0.606	0.387	0.472
LLR	0.525	0.335	0.409

5. **Conclusion.** Both kinds of relations between word, syntagmatic and paradigmatic, are closely related to the semantic and syntactic information of words. Word embedding, a state-of-the-art distributional semantic model, which is considered to be able to capture the semantic and syntactic information of a word, is widely applied to many natural language processing tasks in the sense of finding paradigmatic relations of words. When the whole corpus is considered, word embedding based similarity computing discovers words

in paradigmatic relation, but when a sentence is considered, where words in paradigmatic relations are a few, word embedding based similarity computing can discover words in syntagmatic relations. Based on this hypothesis, this paper proposes to use word embedding to discover words in syntagmatic relations. We conduct experiments to test the practicality, effectiveness and cross language applicability of the proposed method and also apply it to the practical natural language processing task - collocation extraction. All the experimental results show that word embedding can be used to discover the syntagmatic relations at the sentence level and it is effective, competitive and complementary to state-of-the-art methods in the literature. And cross language experimental results show that our proposed method is also language independent.

As the first attempt to discover the syntagmatic relations at sentence level, this work is of significance to syntagmatic based natural language processing tasks at sentence level, such as parsing and machine translation. As the experimental results show, the proposed method is effective and complementary to the state-of-the-art methods. In the future, we will make full use of this complementary effect and search for a clever combination of the proposed method with the state-of-the-art methods in order to further improve the performance precision, recall and F_1 measure of the task and also extend our method to the discovering of multi-gram syntagmatic relations.

Acknowledgment. This work was supported by National Natural Science Foundation of China under Grant No. 61672127 and Fujian Young Teacher Education Research Foundation under Grant No. JA14217.

REFERENCES

[1] F. de Saussure, Course in General Linguistics , *Eds. Charles Bally and Albert Sechehaye. Trans. Roy Harris. Open Court, La Salle, Illinois*, 1983.

[2] P. Pecina, Lexical Association Measures and Collocation Extraction, *Language Resources and Evaluation*, vol.44, no. 1-2, pp.137-158, April 2010.

[3] J. Antoch, L. Prchal and P. Sarda, Combining Association Measures for Collocation Extraction Using Clustering of Receiver Operating Characteristic Curves , *Journal of Classification*, vol.30, no. 1, pp.100-123, April 2013.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, Efficient Estimation of Word Representations in Vector Space, *arXiv preprintarXiv*, 1301.3781, 2013.

[5] S. Evert and B. Krenn, Methods for the Qualitative Evaluation of Lexical Association Measures , *the 39th Annual Meeting of the Association for Computational Linguistics,* pp.188-195, July 2001.

[6] S. X. Yang, Machine Learning for Collocation Identification, *International Conference on Natural Language Processing and Knowledge Engineering,* pp. 315-320, October 2003.

[7] P. Pecina, An Extensive Empirical Study of Collocation Extraction Methods, *the Association for Computational Linguistics Student Research Workshop,* pp.13-18, June 2005.

[8] W. Y. Li, Chinese Collocation Extraction and Its Application in Natural Language Processing, *PhD Thesis, The Hong Kong Polytechnic University*, 2007.

[9] S. Martens and V. Vandeghinste, An Efficient, Generic Approach to Extracting Multi-word Expressions from Dependency Trees, *CoLing Workshop on Multiword Expressions: From Theory to Applications,* pp.1-4, August 2010.

[10] A. Dinu, L. P. Dinu and I. T. Sorodoc, Aggregation Methods for Efficient Collocation Detection, *the 9th International Conference on Language Resources and Evaluation,* pp.4041-4045, May 2014.

[11] K. W. Church and P. Hanks, Word Association Norms, Mutual Information, and Lexicography, *Computational Linguistics,* vol. 16, no. 1, pp.22-29, March 1990.

[12] M. P. Oakes, Statistics for Corpus Linguistics, *Edinburgh University Press, Edinburgh*, UK, 1998.

[13] M. S. Sun, C. N. Huang and J. Fang, A Pilot Study on Corpus-based Quantatitive Analysis of Chinese Collocations, *ZHONGGUOYUWEN,* no. 01, pp.29-38, 1997.

[14] D. K. Lin, Automatic Identification of Non-compositional Phrases, *the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics,* pp.317-324, June 1999.

[15] H. Takayama, Y. Kato, T. Ohno, S. Matsubara and Y. Ishikawa, Collocation Extraction Using a PMI-Based Association Measure for Dependency Tree Pattern, *The Tenth Symposium on Natural Language Processing,* pp.138-143, October 2013.

[16] G. Bouma, Collocation Extraction beyond the Independence Assumption, *the Meeting of the Association for Computational Linguistics, Short paper track,* pp.109-114, July 2010.

[17] R. Carlini, J. Codina-Filba and L. Wanner, Improving Collocation Correction by Ranking Suggestions Using Linguistic Knowledge, *the 3rd Workshop on NLP for Computer-Assisted Language Learning,* pp.1-12, November 2014.

[18] L. R. Dice, Measures of the Amount of Ecologic Association Between Species, *Ecology,* vol. 26, no. 3, pp.297-302, July 1945.

[19] F. Smadja, Retrieving Collocations from Text: Xtract, *Computational Linguistics,* vol. 19, no. 1, pp.143-177, March 1993.

[20] A. Kilgarriff, Which Words Are Particularly Characteristic of a Text? A Survey of Statistical Approaches, *AISB Workshop on Language Engineering for Document Analysis and Recognition,* pp.33-40, 1996.

[21] C. D. Manning and H. Schütze, Foundations of Statistical Natural Language Processing, *MIT press, Cambridge, UK*, 1999.

[22] T. Dunning, Accurate Methods for the Statistics of Surprise and Coincidence, *Computational Linguistics,* vol. 19, no. 1, pp.61-74, 1993.

[23] S. Evert, The Statistics of Word Cooccurrences: Word Pairs and Collocations, *PhD Thesis, Uiversity of Stuttgart*, 2005.

[24] V. Seretan, L. Nerima and E. Wehrli, Extraction of Multi-word Collocations Using Syntactic Bigram Composition, *the Fourth International Conference on Recent Advances in NLP (RANLP-2003),* pp.424-431, January 2003.

[25] Z. Y. Liu, Alignment Based Acquisition of Collocation and Application in Machine Translation, *PhD Thesis, Harbin Institute of Technology*, 2013.

[26] R. F. Xu, Q. Lu, K.-F. Wong, W. J. Li, Classification-based Chinese Collocation Extraction, *International Conference on Natural Language Processing and Knowledge Engineering,* pp.308-315, August 2007.

[27] T. Mikolov, Q. V. Le and I. Sutskever, Exploiting Similarities among Languages for Machine Translation, *arXiv preprint ar-Xiv:1309.4168*, 2013.

[28] M. Bansal, K. Gimpel and K. Livescu, Tailoring Continuous Word Representations for Dependency Parsing, *the Association for Computational Linguistics,* pp.809-815, June 2014.

[29] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng and C. Potts, Recursive Deep Models for Semantic Compositionality over a Sentiment Treebank, *the 2013 Empirical Methods on Natural Language Processing (EMNLP-13),* pp.1631-1642, October 2013.

[30] R. Collobert and J. Weston, A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning, *International Conference on Machine Learning (ICML),* pp.160-167, July 2008.

[31] A. Passos, V. Kumar and A. McCallum, Lexicon Infused Phrase Embeddings for Named Entity Resolution, *the Eighteenth Conference on Computational Natural Language Learning,* pp.78-86, June 2014.

[32] J. Devlin, R. Zbib, Z.Q. Huang, T. Schwartz and J. Makhoul, Fast and Robust Neural Network Joint Models for Statistical Machine Translation, *Meeting of the Association for Computational Linguistics,* pp.1370-1380, June 2014.

[33] R.J. Fu, J. Guo, B. Qin, W.W. Che, H.F. Wang and T. Liu, Learning Semantic Hierarchies via Word Embeddings, *the 52th Annual Meeting of the Association for Computational Linguistics,* pp.1199-1209, June 2014.

[34] L.C. Tan, H.T. Zhang, C. Clarke, and M. Smucker, Lexical Comparison between Wikipedia and Twitter Corpora by Using Word Embeddings, *the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing,* pp.657-661, July 2015.

[35] O. Melamud, O. Levy and I. Dagan, A Simple Word Embedding Model for Lexical Substitution, *the 1st Workshop on Vector Space Modeling for Natural Language Processing,* pp.1-7, 2015.

[36] B. Salehi, P. Cook and T. Baldwin, A Word Embedding Approach to Predicting the Compositionality of Multiword Expressions, *the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* pp.977-983, June 2015.

[37] S. Rodrıguez-Fernández, L. Espinosa-Anke, R. Carlini and L. Wanner, Semantics-driven Recognition of Collocations Using Word Embeddings, *the 54th Annual Meeting of the Association for Computational Linguistics,* pp.499-505, August 2016.

[38] M. Farahmand and J. Henderson, Modeling the Non-Substitutability of Multiword Expressions with Distributional Semantics and a Log-Linear Model, *The Workshop on Multiword Expressions,* pp.61-66, August 2016.

[39] R. Řehůřek and P. Sojka, Software framework for topic modelling with large corpora, *the LREC 2010 Work-shop on New Challenges for NLP Frameworks,* pp.46-50, May 2010.

[40] D.G. Huang, D.Q. Tong and Y.Y. Luo, HMM Revises Low Marginal Probability by CRF for Chinese Word Segmentation, *CIPS-SIGHAN Joint Conference on Chinese Language Processing (CLP2010),* pp.216-220, July 2010.

[41] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard and D. McClosky, The Stanford CoreNLP Natural Language Processing Toolkit, *The 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp.55-60, June 2014.