

N-best Rescoring for Parsing Based on Dependency-Based Word Embeddings

Yu-Ming Hsieh* and Wei-Yun Ma*

Abstract

Rescoring approaches for parsing aims to re-rank and change the order of parse trees produced by a general parser for a given sentence. The re-ranking performance depends on whether or not the rescoring function is able to precisely estimate the quality of parse trees by using more complex features from the whole parse tree. However it is a challenge to design an appropriate rescoring function since complex features usually face the severe problem of data sparseness. And it is also difficult to obtain sufficient information requisite in re-estimation of tree structures because existing annotated Treebanks are generally small-sized. To address the issue, in this paper, we utilize a large amount of auto-parsed trees to learn the syntactic and semantic information. And we propose a simple but effective score function in order to integrate the scores provided by the baseline parser and dependency association scores based on dependency-based word embeddings, learned from auto-parsed trees. The dependency association scores can relieve the problem of data sparseness, since they can be still calculated by word embeddings even without occurrence of a dependency word pair in a corpus. Moreover, semantic role labels are also considered to distinct semantic relation of word pairs. Experimental results show that our proposed model improves the base Chinese parser significantly.

Keywords: Word Embedding, Parsing, Word Dependency, Rescoring.

1. Introduction

How to solve structural ambiguity is an important task in building a high-performance statistical parser, particularly for Chinese. Since Chinese is an analytic language, words play different grammatical functions without inflections. A great deal of ambiguous structures will be produced by a parser if no structure evaluation is applied. Therefore, the major task of a

* Institute of Information science, Academia Sinica, Taipei, Taiwan
E-mail: {morris, ma}@iis.sinica.edu.tw

parser is to determine the most plausible parse tree from these ambiguity structures. Re-ranking approaches are widely used in parsing natural language sentences for further advancing the performance of statistical parsers (Shen, Sarkar & Toshi, 2003; Hsieh, Yang & Chen, 2007; Johnson & Ural, 2010; Le, Zuidema & Scha, 2013; Zhu, Qiu, Chen & Huang, 2015). It is an intuitive and efficient strategy to determine the most plausible parse tree from a set of candidate parse trees of a sentence through a rescoring approach.

Treebanks are a widely used resource in parsing task, as it provides useful statistical distributions regarding grammar rules, words, part-of-speeches (PoS), and word-to-word association¹. However it is difficult to obtain sufficient information requisite in re-estimation of tree structures from existing annotated Treebanks since sizes of treebanks are generally small and insufficient, resulting in a common problem of data sparseness, especially for more complex features in a re-scoring scenario, such as word-to-word dependency associations. So learning information and knowledge from analyzing large-scaled unlabeled data is a compulsory strategy, which is proved useful in the previous works (Wu, 2003; Chen, 2008; Yu *et al.*, 2008).

In this paper, we utilize a large amount of auto-parsed trees to learn the syntactic and semantic information and present a simple but effective score function in order to integrate the scores provided by the base parser and word-to-word dependency association scores. The dependency association scores are based on dependency-based word embeddings, learned from a large amount of auto-parsed trees. The score function can relieve the problem of data sparseness, since the dependency association scores can still be calculated by word embeddings even without the occurrence of a dependency word pair in a corpus. In addition, Kim, Song, Park & Lee (2015) proves that the dependency labels (i.e., semantic role labels) in re-ranking parsed tree are important information. As a result, semantic role labels are also considered to distinct semantic relation of word pairs.

Word embeddings have become increasingly popular lately, proving to be valuable as a source of features in a broad range of NLP tasks (Turian, Ratinov & Bengio, 2010; Socher *et al.*, 2013; Bansal, Gimpel & Livescu, 2014). The *word2vec* package (Mikolov, Chen, Corrado & Dean, 2013) is among the most widely used word embedding models today. Their success is largely due to an efficient and user-friendly implementation that learns high quality word embeddings from very large corpora. The *word2vec* package learns low dimensional continuous vector representations for words by considering window-based contexts, i.e., context words within some fixed distance of each side of the target words. Another different context type is dependency-based word embedding (Bansal *et al.*, 2014; Levy & Goldberg,

¹ Word-to-word association is also called word dependency, a dependency implies its close association with other words in either syntactic or semantic perspective.

2014; Melamud, McClosky, Patwardhan & Bansal, 2016), which considers syntactic contexts rather than window contexts in *word2vec*. Dependency-based word embedding is able to capture functional similarity (as in *lion:cat*) rather than topical similarity or relatedness (as in *lion:zoo*) that *word2vec* would probably provide. Further, Melamud *et al.* (2016) prove that the approach should depend on the tasks to choose the right context type, windows size, and dimensionality in word embedding. From the experiments done by Bansal *et al.* (2014) and Melamud *et al.* (2016), results show benefits of such modified-context embeddings in dependency parsing task. Kim *et al.* (2015) proclaim similar arguments that semantic view should be taken into consideration in re-ranking parse trees because a dependency word pair implies both syntactic and semantic relations.

We propose a rescoring approach for parsing based on a combination of original parsing score and semantic plausibility of dependencies to assist the determination of the parse tree among the n -best parse trees. The original parsing score is produced from the Chinese parser (Hsieh, Bai, Chang & Chen, 2012), and the semantic plausibility of dependencies is calculated from dependency-based word embedding. There are three main steps in our rescoring approach. The first step is to have the parser produce n -best structures. Second, we extract word-to-word associations (word dependency) from a large amount of auto-parsed data and build dependency-based word embedding. The last step is to build a structural rescoring method to find the best tree structure from the n -best candidates. We conduct experiments on the standard data sets of the Chinese Treebank. The results indicate that our proposed approach improves the base Chinese parser significantly.

The remainder of this paper is organized as follows. In Section 2, we describe the rescoring approach and introduce a strategy to extract word dependency associations from a large-scale unlabeled corpus. In Section 3, we report the results of experiments conducted to evaluate the proposed rescoring approaches on different scores of dependency. Section 4 provides a discussion on the related work. Section 5 contains our concluding remarks.

2. Rescoring Syntactic Parse Trees with Dependency Embeddings

In this section, we will describe our rescoring approach. First, we need a parser to generate n -best parse trees with their structural scores, and then select the best parse tree through a score function which considers the structure and the dependency embeddings. Figure 1 shows a flow chart of our rescoring approach. Given an input sentence, the ‘parser’ is responsible for word segmentation, part-of-speech tagging, semantic role labeling, and generate n -best parse trees. And then the ‘rescoring’ is based on a combination function of the original parsing score and the semantic score of dependencies to determine from the quality of the n -best parse trees.

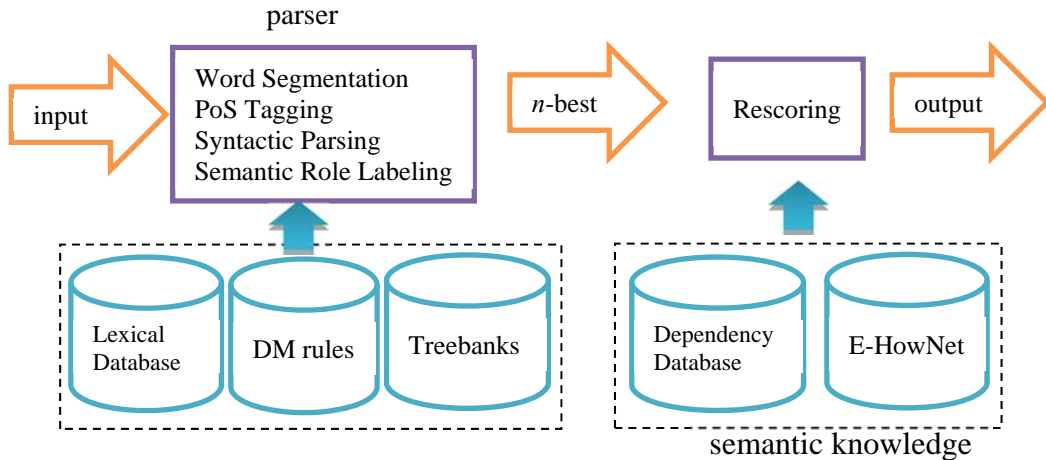


Figure 1. A flow chart of rescoring approach.

We adopt *word2vec* (Levy & Goldberg, 2014) package to train dependency-based word embeddings from the parsed trees of our corpus. Similar to the approach of Levy & Goldberg (2014), the two steps are needed to achieve this training stage.

- Step 1: Extract word associations from each tree.
- Step 2: Train a dependency embeddings including target embeddings and context embeddings.

Figure 2 illustrates the first step of word association extraction. Based on the head word information (i.e. the semantic role of the word is ‘Head’ or ‘head’, called the ‘head word’), we extract dependence word-pairs between head words and their arguments or modifiers. For example, if we have a sentence ‘他穿著破舊的上衣 / he is wearing shabby clothes’, five word dependency pairs will be extracted from this tree structure including head words, modifier words, part-of-speeches, semantic role labels, frequencies and etc. The word dependency in (*agent* 他/*he Nh*, *Head[S]* 穿/*wear VC*) represents a head word ‘穿/*wear*’, its PoS VC (Active Transitive Verb), and its modifier (他/*he, Nh*) with the semantic role label ‘*agent*’².

² The ‘agent’ is a semantic role label. There are 60 semantic roles including thematic roles of events such as ‘agent’, ‘theme’, ‘instrument’, and secondary roles of ‘location’, ‘time’, ‘manner’ and roles for nominal modifiers. Please refer to CKIP technical report (Chinese Knowledge Information Processing Group [CKIP], 2013) for detail information.

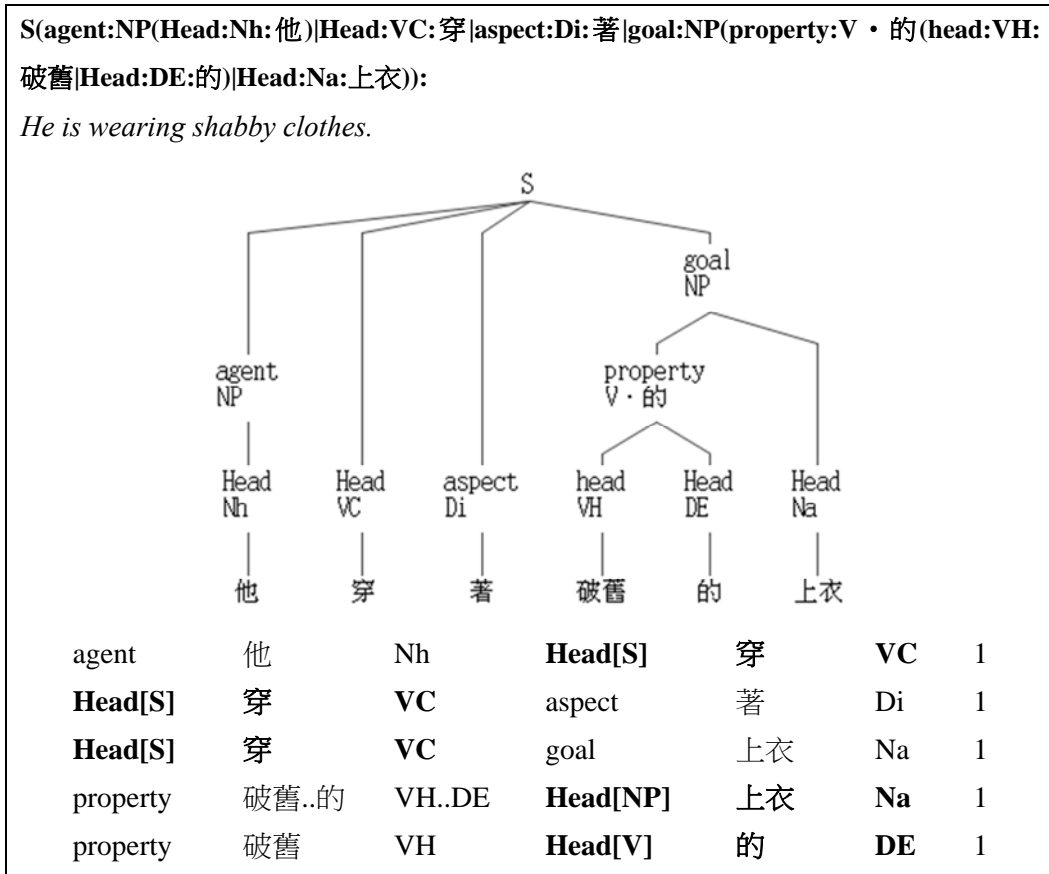


Figure 2. Extraction of word-to-word association (word dependency) from a parse tree.

The second step is to train dependency embeddings. We transform the information of word-to-word associations (in Figure 2) into the *word2vecf* format (in Figure 3). The specific format actually contains two types: the left column is the original word dependency, and right column is the inversion of these words.

word dependency	Inverse dependency
(穿, agent_他)	(他, agentI_穿)
(穿, aspect_著)	(著, aspectI_穿)
(穿, goal_上衣)	(上衣, goalI_穿)
(上衣, property_破舊)	(破舊, propertyI_上衣)
(的, property_破舊)	(破舊, propertyI_的)

Figure 3. The transformed word dependency knowledge and the dependency-based embedding format.

The final step is rescoreing. We integrate the original parsing score with dependency embedding score of parsed n -best trees, and then select the best tree based on the rescoreing scores. This step will be illustrated more clearly in the next section.

2.1 Measuring Dependency Plausibility of a Parse Tree

We use trained dependency-based word embeddings, including the target and the context embeddings, to design our score function. Figure 4 indicates an example of word/context embeddings. The symbol u is target word embedding, and the symbol v is context word embedding, where n and m represent size of u and v embeddings respectively. Both embeddings dimension is 300. For example, the embedding of the target word ‘穿/wear’ is $u_{t=穿} = [0.35, -0.33, -0.01, \dots, -0.17]$, and the embedding of the target word ‘goal_旗袍’ is $v_{t=旗袍} = [0.05, -0.06, 0.01, \dots, 0.04]$.

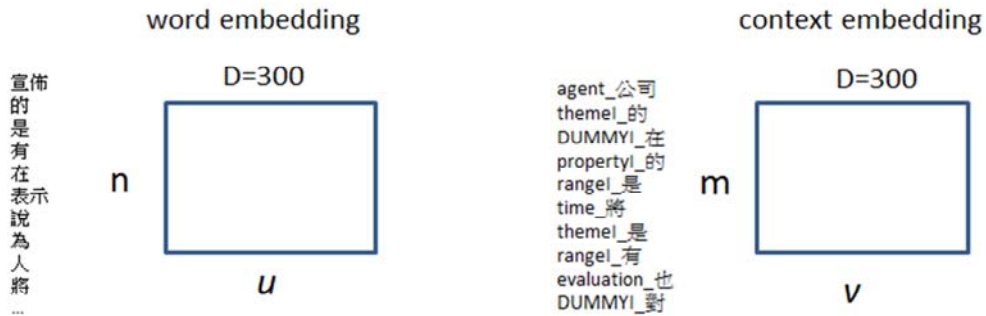


Figure 4. Dependency embeddings: Word and context embedding.

Table 1. The top-10 closest words to the target word ‘穿/wear’ and its context activated.

Words similar to: '穿'		Contexts activated by: '穿'	
穿上	0.7544027174	goal_旗袍	0.6248433766
穿戴	0.7354367107	goal_泳裝	0.6048768995
穿著	0.7281331792	goal_母乳	0.5800340081
改穿	0.7054656651	agent_梁朝偉	0.5690256649
身穿	0.6858005123	goal_枕頭	0.5657152053
挑	0.6851677693	agent_模特兒	0.5602549151
換穿	0.6826460874	complement_昏倒	0.5553763580
穿出	0.6747521378	agent_樂器	0.5528911113
試穿	0.6619956193	Head_丟擲	0.5506488225
		goal_拳頭	0.5498251675

The word similarity list and context activated of the word ‘穿/wear’ are shown in Table 1³. The *DepScore* is our dependency embedding score in Equation (1). The *DepEmb* of a parse $y(s)$ tree with dependency embeddings is to represent the dependency associations of a parse tree as a set of target t and context c . Each dependency in a parse tree can be regarded as a (t,c) set. The semantic plausibility of a parse tree is then defined as the sum of the scores of all dependencies in the tree. That is, the semantic score of a parse tree y is defined as

$$DepScore(y(s)) = \sum_{(t,c) \in y(s)} DepEmb(u_t, v_c) \tag{1}$$

$$DepEmb(u_t, v_c) = \log \frac{\exp(u_t \cdot v_c)}{\sum_{x \in Rel(t)} \exp(u_t \cdot v_x)}$$

DepEmb(u_t,v_c) is calculated by taking exponential of dot product $u_t \cdot v_c$, following by a normalization. $x=Rel(t)$ means a word t and its dependency word x from dependency database (see Equation 1). Finally, we obtain dependency embedding score and frequency for each pair shown in Table 2. With the dependency embedding and frequency information, our design of rescoring will be discussed in the next section.

Table 2. The target word ‘穿/wear’ and its dependency word *DepEmb* score and frequency .

By dependency embedding score:		By frequency:	
TOTAL_穿	1.0000000000	TOTAL_穿	9754.000000
goal_高鞋	0.0008847827	goal_衣服	256.000000
goal_护士服	0.0007814124	aspect_了	197.000000
goal_麵粉袋	0.0007664880	goal_制服	186.000000
goal_官服	0.0007657732	quantity_都	160.000000
goal_衣裙	0.0007545836	quantity_所	146.000000
agent_靴子	0.0007487881	aspect_起	145.000000
reason_擊	0.0007484265	deontics_要	127.000000
goal_西褲	0.0007455655	negation_不	118.000000
goal_健體鞋	0.0007444242	evaluation_也	102.000000
theme_楊孟霖	0.0007376180	agent_他	100.000000

2.2 Rescoring Model for Parse Trees

A re-ranking model ranks a set of candidate dependency parse trees according to its criterion. The criterion of our re-ranking model is a combination of syntactic and semantic score. Given

³ The interface of the provided information is revealed in the following website.
<http://irsrv2.cs.biu.ac.il:9998/?word=wear>.

a sentence s , we define the rescoring model as follows, and the best parse tree \hat{y} of s is obtained from y .

$$\hat{y} = \operatorname{argmax}_{y \in \text{gen}(s)} \lambda * \text{CDMScore}(y(s)) + (1 - \lambda) * \text{DepScore}(y(s)) \quad (2)$$

where $\text{gen}(s)$ is a set of n -best outputs of a baseline parser. In Equation (2), $\text{CDMScore}(y)$ is the original parsing score generated by CDM Parser (a context-dependent PCFG Parser). The log probability of a parse tree y is used. $\text{DepScore}(y)$ is the final semantic score of a parse tree y . CDMScore and DepScore are normalized, i.e. $(i-\text{min})/(\text{max}-\text{min})$. The symbol λ is an weighting parameter between $\text{CDMScore}(y)$ and $\text{DepScore}(y)$. We substitute λ for every interval of 0.1 from 0 to 1, and design the relating λ from development sets.

3. Experiments

We conducted experiments on our experimental data setting and the evaluation results. And we investigate different types of context in word dependency extraction process and analyze the test results.

3.1 Experimental Settings

Several parts are introduced below to illustrate our experimental design, including corpus, software, evaluation criteria.

Trebank: We employ Sinica Treebank⁴ as our experimental corpus. It contains 61,087 syntactic tree structures and 361,834 words. The syntactic theory of Sinica Treebank is based on the Head-Driven Principle; that is, a sentence or phrase is composed of a phrasal head and its arguments or adjuncts. We use the same dataset in Hsieh *et al.* (2012), and divide the treebank into four parts: the training data (55,888 sentences), the development set (1,068 sentences), the test data T06 (867 sentences), and the test data T07 (689 sentences). The test datasets (T06, T07) are the datasets used in CoNLL06 and CoNLL07 dependent parsing evaluation individually. The only difference between Sinica Treebank data and CoNLL data is that the CoNLL is in dependency format. We use labeled information of gold-standard word segmentation and POS tags as our input data in all our experiments.

Large Corpus: The Gigaword corpus contains about 1.12 billion Chinese characters, including 735 million characters from Taiwan's Central News Agency (traditional characters), and 380 million characters from Xinhua News Agency (simplified characters). We used the Central News Agency (CNA) portion of Chinese Gigaword Version 2.0 (LDC2009T14). We need to perform word segmentation and part-of-speech tagging before parsing, and the baseline parser is used to

⁴ Please refer to the Sinica Treebank webpage for further information: <http://treebank.sinica.edu.tw/>.

parse the sentences in the Gigaword corpus. In our experiments, word dependencies are extracted from CNA texts. Finally, we obtain 37,711,822 parse trees and extract 224,371,806 word-to-word associations (word dependency).

Chinese Parser: The parser includes some components: Chinese word segmentation, PoS tagging, syntactic parsing, and semantic role labeling. The Chinese word segmentation system and part-of-speech tagging system reaches a high performance of over 95% and 96% in accuracy respectively (Tsai & Chen, 2014). The extracted grammar rules of Sinica Treebank are used in the syntactic parser. We follow You and Chen’s method (You & Chen, 2004) to assign semantic role automatically. The system adopts a probabilistic model of head-modifier relations and achieves 92.71% accuracy in labeling the semantic roles.

Estimating Parsing Performance: In evaluation, we use a structural evaluation system called PARSEVAL to compare the parsing results with the gold standard. Throughout the experiment, the bracketed f -score (BF) from PARSEVAL is used as the parsing performance metric.

Dependency-based Embedding: Following the illustration about word dependency extraction in section 2.1, we create word embedding from large-scale corpus by *word2vecf* tool with parameter *dimension* = 300, *negative sample*=15, *minimum frequency*=5, and *iterations*=5 in our experiments. Finally, we have 568,180 words and 1,721,301 contexts in u and v embedding respectively. $\lambda = 0.7$ in Equation (1) is used in the all experiments below.

3.2 Results of Rescoring Approach

First, we use the n -best tree structures produced from F-PCFG parser and observe their bracketed f -scores (BF) variation. The oracle n -best BF of F-PCFG parser are listed in Table 3. In the data set T07, we find that for the 20-best result, the oracle BF score is 94.66%. In contrast, in the 1-best result, the oracle BF score is 83.91%.

Table 3. Oracle BF score as a function of number N of N -best parses.

	sent. #	1	3	5	10	20
T06	867	88.56	92.77	94.11	95.69	96.35
T07	689	83.91	89.83	91.50	93.57	94.66

Rescoring evaluation: The rescoring evaluation results of the proposed model ‘Rescoring-emb’ and its competitors are given in Table 4. The ‘F-PCFG’ parser adopts a linguistically-motivated grammar generalization method to obtain a binarized grammar from original CFG rules extracted from treebank (Hsieh, Yang & Chen, 2015). The ‘CDM’ Parser proposed by Hsieh *et al.* (2012) achieves the best score in Traditional Chinese Parsing task of SIGHAN Bake-offs 2012 (Tseng, Lee & Yu, 2012). Compared with the other two parsers of F-PCFG and CDM, the approach of ‘Rescoring-emb’ takes additional semantic score of

dependency parse trees into consideration and achieves high performance on BF scores.

Table 4. Results on T06 and T07 data set.

	T06	T07
F-PCFG	88.56	83.91
CDM	89.91	85.86
Rescoring-emb	90.55	86.41

From Table 4, our rescoring method obtains improvement from 88.56% to 90.35% and the BF score is between the oracle score 1-best and 3-best. The result of the experiment is similar to Charniak & Johnson (2005) proposed re-ranking model. An example of the improved n -best parse tree after baseline parsing is presented below. The sequence of these results represents the scores of the original tree in order and the best result is in the 4th tree after rescoring approach.

S(NP(DM:第一天|DM:有 5 0 0 0 多個|Head:Na:人)|Head:VC:參觀)
 VP(NP(DM:第一天|DM:有 5 0 0 0 多個|Head:Na:人)|Head:VC:參觀)
 VP(DM:第一天|NP(DM:有 5 0 0 0 多個|Head:Na:人)|Head:VC:參觀)
 S(DM:第一天|NP(DM:有 5 0 0 0 多個|Head:Na:人)|Head:VC:參觀)
 ...

In addition, we conducted another experiment without a semantic role label in word dependency pairs. The BF score decreased from 90.55% to 90.05% in T06 data set. The results show that using semantic role labels in word dependency is useful. Furthermore we attempt to compare the effect on using the traditional conditional probability method called ‘Rescoring-freq’. Therefore, we modify the Equation (2) into Equation (3). The symbol m is the modifier word of the dependency word pair, h is the head word, $freq(m,h)$ is frequency of the (m,h) dependency, and $freq(h)$ is frequency of the h . If $freq(m,h)$ is 0, we will replace it by $1/total\ word\ dependency$.

$$\hat{y} = \operatorname{argmax}_{y \in gen(s)} \lambda * CDMScore(y(s)) + (1 - \lambda) * WAScore(y(s)) \quad (3)$$

$$WAScore(y(s)) = \sum_{(m,h) \in y(s)} \log P(m|h)$$

$$P(m|h) = \frac{freq(m,h)}{freq(h)}$$

We also compare the performance of using the traditional conditional probability method with our approach. From the experimental results, the *BF* fell by 0.3% in T07 dataset in Table 5, denoting that the embedding-based scoring has better results than the traditional approach since the embedding score can relax the data sparseness problem, i.e., since dependency scores can be still calculated by word embeddings even without the occurrence of a dependency word pair in a corpus. This verifies the finding of Melamud, Levy & Dagan (2015) in their lexical substitution research based on the word embedding model.

Table 5. Results of ‘Rescoring-emb’ and ‘Rescoring-freq’

	T06	T07
F-PCFG	88.56	83.91
CDM	89.91	85.86
Rescoring-emb	90.55	86.41
Rescoring-freq	90.35	86.19

3.3 Effects of Word Sense Information

In addition to word information, we give a study about the effect on embedding with word sense information. Regarding to word sense information, we use the head senses of words expressed in E-HowNet⁵ as words’ semantic information. For example, the E-HowNet definition of 車輛(Na), is {LandVehicle|車:quantity={many|多}}, and its head sense is “LandVehicle|車”. For detailed description about E-HowNet, readers may refer to Huang, Chung & Chen (2008).

Therefore, to obtain sense definition of lexicons, we convert the word dependency data in Figure 3 corresponding to the E-HowNet dependency (or concept-to-concept relation) as shown in Figure 5. We have two special cases to handle during the process: 1) unknown words, 2) sense ambiguity. The unknown words are skipped in the present experiment. As for the sense ambiguity, we retain the ambiguity of words in E-HowNet, since Zhao & Huang (1999) demonstrated that the retained ambiguity does not have an adverse impact on their identification system.

⁵ The E-HowNet information please refer to webpage: <http://ehownet.iis.sinica.edu.tw/>

Concept-to-concept dependency	Inverse dependency
(PutOn 穿戴, agent_3rdPerson 他人)	(3rdPerson 他人, agentI_PutOn 穿戴)
(PutOn 穿戴, aspect_AspectValue 時貌值)	(AspectValue 時貌值, aspectI_PutOn 穿戴)
(PutOn 穿戴, goal_clothing 衣物)	(clothing 衣物, goalI_PutOn 穿戴)
(clothing 衣物, property_used 舊)	(used 舊, propertyI_clothing 衣物)
(relation 關聯, property_used 舊)	(used 舊, propertyI_relation 關聯)

Figure 5. The concept-to-concept relation and its inverse dependency.

After mapping, we obtain 1,858 target concepts and 93,058 context concepts. We train the concept level embedding and obtain *DepScore* in Equation (1). The experimental result does not seem to improve the overall performance. However some lexicons with the same PoS tag appearing in the similar context may improve through concept relation. For example the common noun “桌子/table (Na)” and “車子/car (Na)” can be as a patient of the verb to move “移動桌子 vs. 移動車子” and in the context denoting location, “在桌子上 vs. 在車子上.” But if these words are tagged with sense information, table as {furniture|家具} and car as {LandVehicle|車}, the two nouns are distinguished more easily. Furthermore the unknown word influence may cause deficiency in sense information. In ten-thousand-word Sinica Corpus⁶, there are around 8.04% unknown words which do not appear in E-HowNet. And the concept information deficiency is even more serious in CNA Corpus. We suspect that the *DepScore* may not be precise enough because of this factor. In the future, we aim to solve the problem by developing a sense predictor based on lexical analysis and word embeddings to predict the sense of unknown words.

4. Related Work

Our re-ranking estimation approach can be divided into two parts. The first one is rescoring model based on large scale corpus, and the second part is about designing a score function based on word dependency associations.

4.1 Rescoring Model Based on Large Scale Corpus

Most rescoring approaches rely on a post-processing to select the best structure from the n-best parse trees (Shen *et al.*, 2003; Hsieh *et al.*, 2007; Johnson & Ural, 2010; Hayashi, Kondo & Matsumoto, 2013; Le *et al.*, 2013; Zhu *et al.*, 2015) or a robust structural evaluation in their parsing models (Wang, Sagae & Mitamura, 2006; Hsieh *et al.*, 2012). Treebank is a

⁶ Sinica Corpus is the first Balanced Modern Chinese Corpus with part-of-speech tagging. Please refer to the website for detail information (<http://asbc.iis.sinica.edu.tw/>).

widely used resource, but it is generally small-sized. To overcome the data sparseness problem, some certain strategies of rule generalization and specialization are devised to improve the coverage and precision of the extracted grammar rules (Johnson, 1998; Sun & Jurafsky, 2003; Klein & Manning, 2003; Hsieh *et al.*, 2015). However, these studies focused only on syntactic information of parse trees and no semantic information is used in their model. Kim *et al.* (2015) proves that the dependency labels (i.e., semantic role labels) in re-ranking parsed tree are important information. As a result we add semantic role label information in word dependency to distinct semantic relation of word pairs.

4.2 Word Dependency Associations

Common knowledge is needed in a robust parser. How to extract useful information from unannotated large scale corpus and represent the knowledge has been a research issue (Wu, 2003; Chen, 2008; Yu *et al.*, 2008; Hsieh, Chang & Chen, 2014). Similarly, we train our word dependency associations from large-scale corpus. The representation of the dependency associations is like knowledge graph embedding (TransR) (Wang, Zhang, Feng & Chen, 2014) or dependency-based word embedding (*word2vecf*). TransR proposed by Lin *et al.* is adopted for representing semantic scores (Lin, Liu, Sun, Liu & Zhu, 2015). TransR models entities and relations in distinct spaces, and then translates the entities in an entity space into the space of a specific relation. Melamud *et al.* (2016) indicate *word2vecf* (Levy & Goldberg, 2014) in pre-trained embedding on unlabeled data in the Stanford Neural Network Dependency (NNDEP) parser (Chen & Manning, 2014) yields improved performance. In our approach, we use *word2vecf* to train dependency-based word embeddings (target, context) and calculate word dependency association scores in our task.

5. Conclusions

In this paper, we present a rescoring approach for parsing based on a combination of original parsing score and word dependency associations to assist the determination of the best parse tree among the n -best parse trees. To overcome the data sparseness problem, our word dependency associations are modeled through dependency-based word embeddings, learned from a large amount of auto-parsed trees, and semantic role labels are also considered to distinct semantic relation of word pairs. The experiment results indicate that our proposed approach improves the base Chinese parser significantly.

Reference

Bansal, M., Gimpel, K., & Livescu, K. (2014). Tailoring continuous word representations for dependency parsing. In *Proceedings of ACL 2014*, 809-815.

- Charniak, E. & Johnson, M. (2005). Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of ACL 2005*, 173-180.
- Chen, L. (2008). Autolabeling of VN Combination Based on Multi-classifier. *Journal of Computer Engineering*, 24(5), 79-81.
- Chen, D. & Manning, C.D. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of EMNLP 2014*, 740-750.
- Chinese Knowledge Information Processing Group. (2013). *Introducing Semantic Roles in Sinica Treebank*. CKIP Technical Report 2013-01, Institute of Information Science, Academia Sinica, Taipei.
- Hayashi, K., Kondo, S., & Matsumoto, Y. (2013). Efficient stacked dependency parsing by forest reranking. *Transactions of the ACL*, 1(1), 139-150.
- Hsieh, Y.M., Bai, M.H., Chang, J.S., & Chen, K.J. (2012). Improving PCFG Chinese parsing with Context-Dependent Probability Re-estimation. In *Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing*, 216-221.
- Hsieh, Y.M., Chang, J.S., & Chen, K.J. (2014). Ambiguity Resolution for Vt-N Structures in Chinese. In *Proceedings of EMNLP 2014*, 928-937.
- Hsieh, Y.M., Yang, D.C., & Chen, K.J. (2007). Improve Parsing Performance by Self-Learning. *Computational Linguistics and Chinese Language Processing (IJCLCLP)*, 12(2), 195-216.
- Hsieh, Y.M., Yang, D.C., & Chen, K.J. (2015). Linguistically-Motivated Grammar Extraction, Generalization and Adaptation. In *Proceedings of IJCNLP 2005*, 177-187.
- Huang, S.L., Chung, Y.S. & Chen, K.J. (2008). E-HowNet: the Expansion of HowNet. In *Proceedings of the First National HowNet Workshop*, 10-22.
- Johnson, M. (1998). PCFG Models of Linguistics Tree Representations. *Computational Linguistics*, 24(4), 613-632.
- Johnson, M. & Ural, A. E. (2010). Reranking the Berkeley and Brown Parsers. In *Proceedings of HLT-NAACL 2010*, 665-668.
- Kim, A., Song, H., Park, S., & Lee, S. (2015). A Re-ranking Model for Dependency Parsing with Knowledge Graph Embeddings. In *Proceedings of IALP 2015*, 177-180.
- Klein, D. & Manning, C. (2003). Accurate Unlexicalized Parsing. In *Proceedings of the ACL 2003*, 423-430.
- Le, P., Zuidema, W., & Scha, R. (2013). Learning from errors: Using vector-based compositional semantics for parse reranking. In *Proceedings of CVSC*, 11-19.
- Levy, O. & Goldberg, Y. (2014). Dependency-Based Word Embeddings. In *Proceedings of ACL 2014*, 302-308.
- Lin, Y., Liu, Z., Sun, M., Liu, Y., & Zhu, X. (2015). Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of AAAI 2015*, 2181-2187.
- Melamud, O., Levy, O., & Dagan, I. (2015). A Simple Word Embedding Model for Lexical Substitution. In *Proceedings of the Workshop on Vector Space Modeling for NLP 2015*.

- Melamud, O., McClosky, D., Patwardhan, S., & Bansal, M. (2016). The Role of Context Types and Dimensionality in Learning Word Embeddings. In *Proceedings of NAACL-HLT 2016*, 1030-1040.
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of ICLR*.
- Shen, L., Sarkar, A., & Toshi, A. (2003). Using LTAG based features in parse reranking. In *Proceedings of EMNLP*, 89-96.
- Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A.Y. & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*, 1631-1642.
- Sun, H. & Jurafsky, D. (2003). The effect of rhythm on structural disambiguation in Chinese. In *Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing*, 17(1), 39-46.
- Tsai, Y.F. & Chen, K.J. (2014). Reliable and Cost-Effective Pos-Tagging. *Journal of Computational Linguistics & Chinese Language Processing*, 9(1), 83-96.
- Tseng, Y.H., Lee, L.H., & Yu, L.C. (2012). Traditional Chinese Parsing Evaluation at SIGHAN Bake-offs 2012. In *Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing*, 199-205.
- Turian, J., Ratinov, L., & Bengio, Y. (2010). Word representations: A simple and general method for semi-supervised learning. In *Proceedings of ACL*, 384-394.
- Wang, M., Sagae, K., & Mitamura, T. (2006). A Fast, Accurate Deterministic Parser for Chinese. In *Proceedings of COLING-ACL 2006*, 425-432.
- Wang, Z., Zhang, J., Feng, J. & Chen, Z. (2014). Knowledge Graph Embedding by Translating on Hyperplanes. In *Proceeding AAAI-2014*, 1112-1119.
- Wu, A. (2003). Learning verb-noun relations to improve parsing. In *Proceedings of the Second SIGHAN workshop on Chinese Language Processing*, 119-124.
- You, J.M. & Chen, K.J. (2004). Automatic Semantic Role Assignment for a Tree Structure. In *Proceedings of the 3rd SigHAN Workshop on Chinese Language Processing (2004)*, 109-115.
- Yu, K., Kawahara, D., & Kurohashi, S. (2008). Chinese Dependency Parsing with Large Scale Automatically Constructed Case Structures. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING2008)*, 1049-1056.
- Zhao, J. & Huang, C. (1999). The Complex-feature-based Model for Acquisition of VN construction Structure Templates. *Journal of Software*, 10(1), 92-99.
- Zhu, C.X, Qiu, X.P., Chen, X.C. & Huang, X.J. (2015). A Re-ranking Model for Dependency Parser with Recursive Convolutional Neural Network. In *Proceedings of ACL 2015*, 11-19.

