SEEN: A Semantic Dependency Analyzer for Chinese

Jiajun YAN¹, David B. BRACEWELL¹, Fuji REN¹,², and Shingo KUROIWA¹
¹Department of Information Science and Intelligent Systems
Faculty of Engineering, The University of Tokushima
Minami-Josanjima-Cho Tokushima-shi 770-8506
JAPAN
²School of Information Engineering
Beijing University of Posts and Telecommunications
Beijing 100876
CHINA

Abstract: - Determining the semantic structure of sentences is a difficult but desired task. In this paper, we propose a system for determining semantic dependency in Chinese sentences. The system is composed of 3 main modules; Syntactic analysis, headword assignment, and semantic dependency assignment. For the semantic dependency module, many classifiers are tested. The best is able to achieve an accuracy of just under 84%.

Key-Words: - Natural Language Processing, Semantic Dependency, Chinese

1 Introduction

Determining the semantic structure of sentences is strongly desired. If the semantic structure can be determined then machine translation, question and answering, etc. can be improved as they would have a greater insight on the meaning of the sentence. There has been much research on determining semantic structure in English [1], [2] are examples. Also there are freely available corpora with semantic annotation such as [3].

There has, however, been much less research done for Chinese. The most prominent research done is by You and Chen [4]. However, in their research they used the Sinica corpus [5], which uses Taiwanese Chinese. We want to create a semantic dependency analyzer for Mandarin Chinese. The two forms of Chinese are different enough that a system for one may not be effective for the other. Also, since the Sinica corpus already has semantic information tagged, You and Chen used that tag set. With the growing popularity of HowNet among the Chinese NLP community we would like our semantic tag set to be based on HowNet’s. This makes the system more usable for Chinese NLP.

Because of the mentioned differences this paper proposes the SEEN (Semantic dEpendency analyzEr for chiNese) system. The system is broken into 3 main modules, as seen in figure 1. The first module is syntactic analysis. This module takes care of morphological analysis and parsing. The second module is headword assignment. This module assigns headwords to sentences and phrases. The final module is semantic dependency assignment. This module uses varying features and a classifier to determine semantic dependency.

This paper will continue as follows. In section 2, the syntactic analysis module will be discussed. In section 3, the headword assignment will be discussed. Then, in section 4, the semantic dependency assignment module will be explained in detail. In section 5, evaluation results will be given. Finally, in section 7, concluding remarks and future work will be discussed.

2 Syntactic Analysis

The syntactic analysis module is made up of morphological analysis and parsing. While not as far along as English, Chinese programs do exist and are
rapidly improving. In the following sub-sections Chinese morphological analysis and parsing will be briefly discussed.

2.1 Morphological Analysis
Morphological analysis consists of word segmentation and part-of-speech tagging. It is a widely researched topic and many languages have these tools available including Chinese. ICTCLAS 2.0, based on Hidden Markov Models, is a free a morphological analyzer that has a precision of 97.58% for segmentation and 87.32% for part-of-speech tagging [6]. In SEEN, this morphological analyzer will be used.

2.2 Chinese Parsing
There has been a lot of research done in Chinese Parsing. Some of the notable research is [7], which reported that their statistics-based Chinese parser had 86% precision and 86% recall. Levy and Manning developed a factored-model statistical parser and used on the Penn Chinese Treebank [8]. Because the parser was used on the Penn Chinese Treebank, which this paper uses as its corpus it was chosen as the parser for the SEEN system.

3 Headword Assignment
In this module, headwords are assigned to each chunk (phrase). In a semantic dependency grammar, headwords are the constituent that can represent the main meaning of the chunks. Headwords are assigned using a set of handcrafted rules that were created from observations made from the Penn Chinese Treebank. The handcrafted rules were designed to look at the syntactic head of a chunk and the other constituents. It was found that the different syntactic heads followed certain patterns that could easily be defined in rules. Figure 2, shows some example rules.

![Sample Headword Assignment Rules](image)

4 Semantic Dependency Assignment
The semantic dependency assignment module takes the result of a syntactic parse tree and transforms that into a semantic dependency tree. Figure 3, shows an example of going from a parse tree to a semantic dependency tree. In this paper a classifier is used to determine the relation between each headword-modifier pairs.

![From Parse Tree to Semantic Dependency Tree](image)

4.1 Classifiers
Four classifiers, listed below, were used to test the applicability of classifiers to the task and to determine which would perform the best. All of the classifiers are capable of doing multi-category classification and are easy to apply to the problem at hand. Each of the classifiers is briefly described in the follow subsections.

- Naive Bayesian Classifier (NBC)
- Naive Possibilistic Classifier (NPC)
- Decision Tree Classifier (DT)
- Maximum Entropy Classifier (ME)

4.2 Naive Bayesian Classifier
The Naive Bayesian Classifier is widely used due to its efficiency and its ability to combine evidence from a large number of features [4]. It is a probabilistic model that assigns the most probable class to a feature vector. Despite its naive assumption that the features are independent it has been shown to generally do well in classification [9].

4.3 Naive Possibilistic Classifier
The Naive Possibilistic Classifier is based on possibilistic networks. They are able to handle imprecise data better than Naive Bayesian Classifiers. Borgelt and Gebhardt showed that for certain data sets the Naive Possibilistic Classifier can perform as well and sometimes better than Naive Bayesian Classifiers and Decision Tree Classifiers [10].

4.4 Decision Tree Classifier
Decision Trees use a tree structure where at each node a decision is made until a leaf node is reached where the class is given. The most widely used variant is the ID3 decision tree introduced by Quinlan [11]. The ID3 decision tree uses information gain and entropy to induce the tree structure.

4.5 Maximum Entropy Classifier
Maximum Entropy modeling creates a model based on facts from the underlying data while trying to stay uniform as possible [12]. As a classifier it achieves state-of-the-art results. It has also been widely adopted in the NLP field.

5 Experimentation
In the following subsections we will look at the results for headword and semantic dependency assignment. The Penn Chinese Treebank 5.0 [2] was used as the base corpus. Random sentences were selected from the Penn Chinese Treebank and manually annotated with headword and semantic dependency relation information. The manually annotated corpus ended up with 13,010 chunks, 18,827 semantic dependency relations from 605 sentences consisting of 19,642 words.

First, the headword assignment results will be briefly discussed. Then, the semantic dependency analyzer’s results will be examined. Finally, an analysis of errors by the analyzer will be given.

5.1 Headword Assignment Results
To evaluate headword assignment 3,296 chunks from the Treebank were extracted and the handcrafted rules described in the previous section were tested. 3,281 of the chunks had the correct headword assigned resulting in a precision of 99.54%. The result is encouraging and shows that the different syntactic heads do to a high degree follow a pattern. In the

Phrase Type (PT) -- The phrase type feature uses the phrase head, in the example the phrase type is NP.
Phrase Length (PL) -- The phrase length is the number of dependents that makes up the phrase plus one for the headword, in the example the NP phrase has a length of 4.
Headword & Dependent (WORDS) -- The headword and dependent feature is the words that makeup the headword and the currently looked at dependent. When semantic relations are assigned, each headword-dependent pair will be looked at separately, so in the example there are 3 headword-dependent pairs.
Headword & Dependent Part-of-Speech (POS) -- The headword and dependent part-of-speech feature is the parts-of-speech for the headword-dependent pair. In the example these are “NN PU,” “NN NN,” and another “NN PU.”
Context (CON) -- Finally, the context feature is the set of dependent parts-of-speech that fall between the headword and the currently looked at dependent. In the example only one headword-dependent pair has a context that is not empty. For the first “NN PU” pair, the context is “NN (learning)” as it is located in between the dependent “PU” and the headword “NN (revival)”.

Fig.4. Sample parse tree

4.6 Features
Five features, shown below, were chosen to be used in classification. They make up the features that were thought to be the most useful for determining the semantic relations. Using the NP in figure 4 the features will be described. The bold line means headword.
following subsections we will look at the results for headword and semantic dependency assignment.

5.2 Semantic Analysis Results

For testing the classifiers, a 10-fold-cross-validated experiment was used. The parse trees in the Penn Chinese Treebank were used so that the performance of the analyzer in an ideal situation can be determined. Table 1 shows the average accuracy and the standard deviation for each of the classifiers with varying feature combinations.

<table>
<thead>
<tr>
<th>Features</th>
<th>NBC</th>
<th>NPC</th>
</tr>
</thead>
<tbody>
<tr>
<td>POS</td>
<td>70.9% (±4.1%)</td>
<td>71.3% (±3.8%)</td>
</tr>
<tr>
<td>WORDS</td>
<td>61.3% (±6.0%)</td>
<td>44.5% (±4.4%)</td>
</tr>
<tr>
<td>CON</td>
<td>61.6% (±3.7%)</td>
<td>60.9% (±3.9%)</td>
</tr>
<tr>
<td>POS+WORDS</td>
<td>78.1% (±3.6%)</td>
<td>63.2% (±2.6%)</td>
</tr>
<tr>
<td>POS+WORDS+CON</td>
<td>70.5% (±4.5%)</td>
<td>65.7% (±3.1%)</td>
</tr>
<tr>
<td>POS+WORDS+PT</td>
<td>79.2% (±3.7%)</td>
<td>61.8% (±3.7%)</td>
</tr>
<tr>
<td>POS+WORDS+PT+PL</td>
<td>79.1% (±3.7%)</td>
<td>63.2% (±2.6%)</td>
</tr>
<tr>
<td>ALL</td>
<td>80.5% (±3.3%)</td>
<td>64.8% (±2.9%)</td>
</tr>
</tbody>
</table>

The first tests used only a single feature. The three different single features looked at were POS, WORDS, and CON. It can be seen, from the table, that the part-of-speech feature worked much better than the words and context features.

The second test was to combine the part-of-speech and words feature. As can be seen the combination of these two features increased the accuracy over using single features about 10%. Adding the context feature to the part-of-speech and words features helped to increase the performance a little for all the classifiers.

Adding the phrase type feature to the part-of-speech and words features increased the performance of the NBC and DT, but either had no effect or was detrimental to the other classifiers. Adding the phrase length yielded surprising results. The ME classifier's accuracy was greatly reduced.

The final test used the part-of-speech, words, context, and phrase type features. It can be seen that, except for the NPC, the classifiers were able to achieve over 80% accuracy. The interesting result was that the decision tree classifier's accuracy was only 0.2% less than the maximum entropy classifier.

6 Error Analysis

When annotating the headword, some non-proper annotations in the original bracketed data of the Penn Chinese Treebank were found. Examining the errors from the classifiers, we found most were caused by shallow parses. In the original corpus, some bracketed sentences were too shallowly parsed and the dependent mhm Entropy was parsed at the same level as the word that should become the headword of the parse tree.

Fig.5. Some examples of shallow parsing

```plaintext
EX1. (NP (NN (国家) (NN (中国) (NN (经济) (CC (与) (NN (外交) (NN (发展))

English: integral national economic policy and diplomatic development
NP 6-NN NN NN NN CC NN NN

EX2. ( (FRAG (NN (文化) (NN (艺术) (NN (用) (PU ) (NN (宣传) (NN (文化)

(INN (事业) (NN (有) (NN (公司) (VV (提) )))

English: Articles were offered by Angra-Zhong, pictures, by Yilan Cultural Enterprise Limited Company
FRAG 6-NN PU NR NN PU NR NN NN JJ NN VV
```

Fig 6. The original Tree

Fig 7. Tree structure parsed more deeply

Figure 5 shows some examples of such difficult sentences. The tree structure of the original sentence for the first example is shown in Figure 6. The sentence was left ambiguous, and if there had been a deeper parse, then the resulting tree would most likely look that in figure 7 and selecting the headword and relations would be more straightforward. However, as
it is in figure 6 it is difficult to decide which word is
the headword and which relation is correct.
Another part of the problem is that it is a fragment
and not a sentence as shown in example 2 of Figure 5.
However, in Chinese, much information can be gained
from fragments and semantic relations can and should
be assigned. It is however more difficult to assign
headwords and relations to fragments than it is regular
sentences. Since fragments were not omitted, the
system’s accuracy was lower than it would be with just
complete sentences.

7 Conclusion and Future Work
In this paper, we have proposed a new semantic
dependency analyzer for Chinese. The system uses
preexisting components for syntactic analysis and this
paper proposed methods for headword and semantic
derpendency assignment. The rules created for
headword assignment resulted in near perfect
performance. The results of the semantic dependency
analyzer show a high accuracy.
In the future we hope to look at ways of handling too
shallowly parsed sentences. If this can be done then
the accuracy of the classifiers should increase. We will
also look at using support vector machines and other
classifiers. Finally, we hope to use the semantic
analysis in combination with HowNet to create
Chinese predicates that contain semantic roles and
selectional restrictions for those roles based on
sememes in HowNet.

Acknowledgement
This research has been partially supported by the
Ministry of Education, Culture, Sports, Science
and Technology of Japan under Grant-in-Aid for
Scientific Research (B), 14380166, 17300065,
Exploratory Research, 17656218, and the Outstanding
Overseas Chinese Scholars Fund of the Chinese
Academy of Sciences No. 2004-1-1.

References:
[1] Gildea, D., Jurafsky, D.: Automatic labeling of
semantic roles. Computational Linguistic 28
(2002) 496-530
interpretation using an enhanced WorldNet. In:
Proceedings of the 2nd Meeting of the North
American Chapter of the Association for
Technical report, University of Central Florida
(2004)
assignment for a tree structure. In: Proceedings of
the 3rd SIGHAN Workshop on Chinese Language
Computational Linguistics and Chinese Language
Processing 4 (1999) 87-104
the Second SIGHAN workshop affiliated with 41st
ACL. (2003)
Proceedings of the Fifth Workshop on Very Large
Corpora. (1997)
[8] Levy, R., Manning, C.D.: Is it harder to parse
chinese, or the Chinese Treebank? In: Proceedings
of the Association of Computational Linguistics
[9] Rish, I.: An empirical study of the naive bayes
classifier. In: Proceedings of IJCAI- 01 workshop
on Empirical Methods in Artificial Intelligence.
(2001) 41-46
possibilistic classifier. In: Proc.7th European
Congress on Intelligent Techniques and Soft
Computing, Aachen, Germany (1999)
Machine Learning 1 (1986) 81-106
maximum entropy approach to natural language