Abstract

This paper proposes a method for assigning Japanese FrameNet semantic roles to constituents of Japanese sentences. The method employs stochastic models trained with support vector machines. Our system based on the method achieved 79% precision and 70% recall in identifying semantic roles of pre-segmented arguments. For more difficult task to identify the arguments and to assign their semantic roles, the system achieves 72% precision and 61% recall. These results surpass previous research.

1 Introduction

A Japanese specific representation of semantic relationship between a predicate and its arguments is required in order to let computers understand Japanese deeply. Ohara et al. proposed Japanese FrameNet (JFN) as a lexical resource which represents those relationships of the Japanese language (Ohara et al., 2004; Ohara, 2008). FrameNet is known as a lexical resource based on frame semantics (Fillmore, 1982), in which collections of semantic roles for each predicate sense called ‘semantic frame’ are defined. Ohara et al. are constructing JFN with the methodology and the framework of FrameNet, in accordance with differences between English and Japanese.

Each semantic frame contains several lexical units as target predicates and semantic roles called frame elements such as Buyer or Victim. 400 lexical units in 150 semantic frames are currently contained in JFN. Moreover it contains 2000 example sentences on which semantic roles are annotated.

We build a system which annotates semantic roles based on JFN. As proposed in several previous works, we split the task into three consecutive parts, namely sentence segmentation, argument identification, and semantic role assignment.

The system achieves argument identification and semantic role assignment with stochastic models trained with support vector machines.

The rest of this paper is as follows: we review related works in Section 2, describe the detailed method in Section 3, show the experimental results in Section 4, and conclude in Section 5.

2 Related Works

The Japanese language has a feature, which western languages especially English do not have, on its predicate argument structure. The feature is that each argument’s grammatical function is determined by its function words or ‘case’ rather than its position. A Japanese sentence for “I ate fish” can be not only “watashi-ha(I) sakana-wo(fish) tabeta(ate)’ but also “sakana-wo(fish) watash-ha(I) tabeta(ate)” or “sakana-wo(fish) tabeta-yo(ate) watashi-ha(I)”. In those examples, “ha” in “watashi-ha” is a key which determines a grammatical function of “watashi” as a subject (or a theme). Same as above, “wo” in “sakana-wo” makes “sakana” be an object. Because of this feature, the Japanese language allows a particular level of semantic analysis called case frame parsing.

Kawahara et al. proposed an example-based case frame dictionary (Kawahara and Kurohashi, 2002). They claim its effectiveness for indirect anaphora resolution (Sasano et al., 2004). However, case frame parsing is not enough for machine translation and several other tasks which semantic role labeling would have some impacts for, because constituents whose cases are same often have different semantic roles in some semantic frames. A Japanese sentence for “I bought it by cash” is “watashi-ha(I) sore-wo(it) genkin-de(by cash) katta(bought)”. A Japanese sentence for “I bought it over there” is “watashi-ha(I) sore-wo(it) asoko-de(over there) katta(bought)”. It is obvious that each constituent whose case is “de” plays a different semantic role, namely Means for the former and Place for the latter.
The first method for semantic role labeling based on FrameNet was proposed by Gildea et al. (Gildea and Jurafsky, 2002). They achieved argument identification and semantic role assignment with conditional probabilistic models. Their method includes example boosting in order to cover the shortage of annotated examples.

Several improvements were proposed after Gildea et al.’s work. One of the improved areas was a conditional probabilistic model. Maximum entropy (ME) methods (Berger et al., 1996) or support vector machines (SVMs) (Vapnik, 1999) are known as effective methods to acquire stochastic models in particular applications such as question answering (Suzuki et al., 2002) or English-Japanese dictionary construction (Sato and Saito, 2002). Kwon et al. employed ME methods for semantic role parsing based on FrameNet (Kwon et al., 2004). Pradhan et al. and Bejan et al. proposed semantic parsing based on FrameNet or PropBank (Kingsbury and Palmer, 2002) with SVM (Pradhan et al., 2004; Bejan et al., 2004).

Another improved area was sentence segmentation. Baldewein et al. proposed chunk sequences for sentence segmentation (Baldewein et al., 2004). Chunk sequence is a representation which can describe multi-chunk and part-chunk arguments based on part-of-speech and grammatical function such as NP, VP, etc. Since chunk sequences approximate constituents, they allow the use of linguistically illformed features. In other words, Baldewein et al.’s idea is that constituents can be approximated by syntactic information which is shallow rather than semantic one. In terms of the Japanese language, Kawahara et al.’s case frame has the same underlying idea.

Hizuka et al. proposed a method of semantic role labeling based on JFN (Hizuka et al., 2007). They employed ME and SVM for argument identification and semantic role assignment. In order to train stochastic models, their method boosts annotated examples also. Furthermore they proposed a sentence segmentation based on syntactic parse trees.

This paper describes some progress on Hizuka et al.’s work along with the common ground of both research, because their work was reported only in Japanese.

3 Models

Semantic role labeling task is usually split into three subtasks as introduced in Section 1. First of all, a sentence is divided into segments in sentence segmentation. Then, some segments to which semantic roles should be assigned are selected. In other words, constituents are identified in argument identification. Finally, an appropriate semantic role is labeled for each constituent in semantic role assignment. Our method is based on this pipeline framework, which is described in Figure 1.

3.1 Sentence Segmentation

Our method first divides a given sentence into several segments in sentence segmentation. Sentence segmentation allows us to reduce constituent candidates which will be identified whether they are arguments of the given predicate. We used CaboCha, MeCab to be exact, for syntactic parsing.

Since the framework of our automatic labeling is pipelined, failure of this stage is critical. Weakness of Hizuka et al.’s work, indeed, lies in this segmentation part. Namely, they heavily relied on the accuracy of the syntactic tool. Thus if the parser fails to create correct candidates which modify the predicate, all the remaining process fails. To compensate this situation, our system prepares various other phrases which locates before the target predicate:

- A phrase which directly modifies the segment after the predicate.
- A phrase which directly modifies the verbal segment.
- A phrase which ends with a punctuation.

3.2 Argument Identification

Our method next identifies constituents whether semantic role will be labeled to. In argument identification, we estimate the likelihood whether each segment is constituent or not.

Our method estimates likelihoods based on a stochastic model, which we call an argument identification model. We train the argument identification model with SVM. Since SVM is originally a binary classifier, it does not provide likelihood as a probability. Therefore we substitute the distance

3 http://chasen.org/~taku/software/cabocha/
between the input vector and the separating hyperplane for a sigmoid function, and regard the result as its likelihood. The following features are extracted for machine learning:

from the target predicate
- the root form of the predicate
- the conjugation type of the predicate

against the target predicate
- phrasal distance to the predicate
- modification (i.e. structural) distance to the predicate
- word distance to the predicate

against other segments
- phrasal distance to the segment
- part-of-speech of the modified segment
- part-of-speech of the adjacent segments
- part-of-speech of the last morpheme of the modified phrase
- part-of-speech of the first morpheme of the modified phrase

against the segment itself
- whether the segment ends with a punctuation
- part-of-speech of the segment
- type of the function word of the segment
- the number of constituent phrases

3.3 Semantic Role Assignment

Our method then assigns an appropriate semantic role to each constituent. We focus on the fact that there are two kinds of frame elements in JFN: core and non-core. The former is obligatory for a frame and the latter is optional. Hizuka et al. dealt with them equally. However, it would be appropriate to separate them, because the core elements plays a more important role in a sentence. Thus, we build two SVM models; one for the core elements and the other for non-cores. Since SVM is originally a binary classifier, we employ the one-versus-rest method for multi-class pattern recognition.

For the SVM for the cores we used the following features in addition to the features used in the argument identification phase in Section 3.2.
- likelihood estimated by the argument identi-
Table 1: Japanese FrameNet data for evaluation

<table>
<thead>
<tr>
<th>Semantic frame</th>
<th>#Semantic roles</th>
<th>Lexical units</th>
<th>#examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arriving</td>
<td>11 (Goal, Theme, Cotheme, ...)</td>
<td>itaru(arrive, reach), hairu(enter, go into)</td>
<td>143</td>
</tr>
<tr>
<td>Commerce, pay</td>
<td>12 (Buyer, Goods, Money, ...)</td>
<td>harau(pay), shiharau(pay)</td>
<td>56</td>
</tr>
<tr>
<td>Departing</td>
<td>17 (Source, Theme, Area, ...)</td>
<td>saru(leave), nukeru(drop out, pull out)</td>
<td>114</td>
</tr>
<tr>
<td>Theft</td>
<td>12 (Goods, Perpetrator, Source, ...)</td>
<td>nusumu(steal), kusuneru(snaffle)</td>
<td>47</td>
</tr>
<tr>
<td>Traversing</td>
<td>29 (Area, Direction, End_points, ...)</td>
<td>wataru(go across, cross, pass)</td>
<td>89</td>
</tr>
</tbody>
</table>

dication model for the constituent
- the concepts retrieved from a thesaurus named NTT Nihongo GoiTaikei\(^4\). The headword(s) in the segment are consulted for this purpose.
- the category of IREX named entities
- whether an unknown word is included in the segment
- the number of segments which have the same function word
- the number of other segments

For the SVM for the non-cores we used the following additional features.
- likelihood estimated by the core element SVM
- the number of segments to which a core element is assigned
- structural relationship against the segment to which a core element is assigned

4 Evaluation

The Japanese FrameNet data we use is shown in Table 1. We evaluate our method, which adopts SVM, with 5-set cross validation for each. We set three different inputs and outputs as follows:

Input1. sentence and its target predicate
Input2. sentence syntactically parsed manually and its target predicate
Input3. sentence argument-identified manually and its target predicate

Output1. segmentation
Output2. argument identification
Output3. semantic role assignment

We use CaboCha as a syntactic parser, and TinySVM\(^5\) as an implementation of SVM.

Our experimental results are shown in Table 3. Note that the figure within parentheses indicates accuracy instead of recall because of a particular input/output assumption. The result of Hizuka et al.'s previous work is shown in Table 2. Although the training and test sets are different, we can say that our result surpasses the previous research.

As we mentioned in Section 2, our primary goal is to classify constituents whose case are identical into different semantic roles. Figure 2 show an example in which different frame elements in Theft frame are successfully annotated to the segments of the same surface case ‘ga’. Table 4 shows another example in which surface case ‘de’ is differently annotated in Traverse frame.

Table 2: Previous Result based on SVM (Hizuka et al., 2007)

<table>
<thead>
<tr>
<th></th>
<th>Input1</th>
<th>Input2</th>
<th>Input3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output1</td>
<td>(0.93)</td>
<td>0.91</td>
<td>0.93</td>
</tr>
<tr>
<td>Output2</td>
<td>0.72</td>
<td>0.61</td>
<td>0.79</td>
</tr>
<tr>
<td>Output3</td>
<td>0.72</td>
<td>0.61</td>
<td>0.79</td>
</tr>
</tbody>
</table>

Table 3: Our experimental result

<table>
<thead>
<tr>
<th>constituent</th>
<th>semantic role</th>
</tr>
</thead>
<tbody>
<tr>
<td>“tanshin-de” (alone)</td>
<td>Depictive</td>
</tr>
<tr>
<td>“kyosei-renkou-ya-samazamana-jijyo-de” (because of repatriation etc.)</td>
<td>Explanation</td>
</tr>
<tr>
<td>“jitensya-de” (by bicycle)</td>
<td>Means</td>
</tr>
</tbody>
</table>

Table 4: Different frame elements annotated against the same surface case in Traverse frame

4.1 Sentence Segmentation

Due to the pipeline framework described in Section 3, the performance of sentence segmentation decides the upperbound of the whole system performance. From Input1-Output1 recall in Table 3, it is safe to say that our system can cover most
Naomi san ga nusuma reta sanjuuyonin bun no nyuuen gansho iri baggu
Victim Target Goods
(A bag which contained 34 application forms stolen from Naomi )
Goods Target Victim

32 daino kuruma ga funsou kakuha niyotte nusuma reta
Goods Perpetrator Target
( 32 cars were stolen by each clan of the dispute )
Goods Target Perpetrator

Figure 2: Successful annotation of different frame elements to the same surface case in Theft frame

of the segment candidates. As for the remaining several percent, we could not cope with chunking errors by the syntactic parser.

4.2 Argument Identification
The performance of our argument identification model was 93% precision and 90% recall, as shown in Table 3.

While we attained better performance compared with the previous research, we found that our system still picked up wrong segments for an argument, because we used surface information only. Namely, a meaningless segment was sometimes judged ok for a predicate. Some kind of thesaurus should be employed for semantic checking.

4.3 Semantic Role Assignment
The performance of our semantic role assignment was 79% precision and 70% recall, as shown in Table 3. While the SVM for the core elements worked fine, we found that the non-core SVM was overfit to the training samples due to data shortage, namely non-core frame elements are not annotated as densely as cores.

When it comes to the overall performance, our system attained 72% precision and 61% recall. The system also achieved to distinguish semantics which conventional methods based on case frame cannot. We will continue to strengthen our system further.

Acknowledgements
This study is partly supported by the Grant-in-Aid for Scientific Research in Priority Areas JAPANESE CORPUS. We would like to thank Prof. Kyoko Ohara for providing us the JFN data and various insightful advice.

References


