Example Based Word Sense Disambiguation towards Reading Assistant System

Pulkit Kathuria **Kiyoaki Shirai**
School of Information Science
Japan Advanced Institute of Science and Technology
{pulkit, kshirai}@jaist.ac.jp

1 Introduction

Japanese learners often look up words in dictionaries/internet when they read Japanese documents. One word has several possible translations, although a word has only one meaning when it appears in the document. It is rather hard for non-native readers of Japanese to read definition sentences of all meanings. It would be useful to build a system which can not only show the target word's definition sentence and its example usage but also select the correct meaning. Currently, ASUNARO[1] is the only reading assistant system for Japanese learners with Word Sense Disambiguation (WSD hereafter) module. However, definition sentences of EDR dictionary[2] produced by ASUNARO are sometimes unnatural and no example sentence is shown for each sense. In our reading assistant system, we use EDICT[3], the Japanese-English bilingual dictionary that includes definition sentences in English as well as example sentences in Japanese and English. We believe that example sentences are indispensable for Japanese learners to understand meanings of words.

WSD in our reading assistant system is a task of translation selection in machine translation. Many researches in translation selection have been devoted. Dagan et al. proposed a method to use word co-occurrence in target language corpus (Dagan, Itai 1994). Later approaches adopting co-occurrence statistics use simple mapping information between a source and its target words. Lee et al. showed the defect of using 'word-to-word' translation and proposed a translation selection method based on the 'word-to-sense' and 'sense-to-word' (Lee, Yoon, Chang 2003). While example based WSD has also been studied. For example, Shirai et al. proposed a method to disambiguate a sense of a word in a given sentence by finding the most similar example sentences in monolingual dictionary (Shirai, Tamagaki 2004).

For our system, an unsupervised WSD method is considered, because the reading assistant system should handle all words, including low frequency words, in a document. These two unsupervised approaches could compensate each other, since they are based on different knowledge sources. In this research, our goal is to combine two approaches to improve the performance of WSD.

As this paper is a part of our ongoing research, we present our first accomplished example based WSD method. It is designed to choose a sense only in reliable cases, that there is a similar sentence in example database. Thus we consider achieving high precision rather than recall.

We present the details of our example based WSD method in Section 2. Evaluation of our method is reported in Section 3. Finally we conclude the paper in Section 4.

[1]http://hinoki.ryu.titech.ac.jp/asunaro/index-e.php
[2]http://www2.nict.go.jp/r/r312/EDR/
[3]http://www.csse.monash.edu.au/~jwb/edict.html

2 Proposed Method

In this paper, word senses or meanings are defined according to the English-Japanese dictionary EDICT. We develop the WSD classifier that calculates similarity between the input Japanese sentence and example sentences from a dictionary. Then choose the example sentence which is most similar to the input sentence. Only the sense associated with the chosen example sentence is shown in the system.

2.1 Overview

In EDICT, word definitions often contain example sentences in both Japanese and English. Figure 1 shows the sense definitions **S** and example sentences **E** for the Japanese word *"hanashi"*.

For example, let us consider the case where the word sense of "*hanashi*" (story) is to be disambiguated in input sentence **I1**.

I1 Han'nin ga tsukamatta to iu **hanashi** wa mettani kikanai. (Rarely hear a story that the culprit got caught)

The classifier measures the similarity between **I1** and the example sentences **E1** and **E2**. Among them, **E1** may have the highest similarity with **I1**. Therefore, the classifier selects **S1** (story) as the correct sense definition for the word "*hanashi*". In the reading assistant system, the classifier is able to show

hanashi
S1: story (n), talk (v), conversation (n)
E1: Mō koreijō sono **hanashi** o watashi ni kika senaide kudasai. (Please let me not hear of that story any more.)
S2: discussions (n), argument (n)
E2: 3-Jikan giron shitaga, wareware wa **hanashi** ga matomaranakatta. (After 3 hours of discussion we got nowhere.)

Figure 1: Sense and Example Sentences of "*hanashi*"

the definition sentences of the sense as well as the example sentences for a target word in input sentence.

In order to choose example sentence (E) which is most similar to input sentence (I), we measure overall similarity $sim(I, E)$ as a sum of syntactic similarity $syn(I, E)$ and similarity obtained from collocation $coll(I, E)$. The classifier chooses the sense associated with the example sentence which has the highest score $sim(I, E)$ and doesn't choose any sense if the overall score $\leq$ threshold T, because the classifier cannot find an example sentence similar enough. In the following two subsections we explain about similarity measures $syn(I, E)$ and $coll(I, E)$, respectively.

2.2 Syntactic Similarity

$syn(I, E)$ refers to syntactic similarity between two sentences I and E, for which we exploited the Japanese dependency structure usually represented by the linguistic unit called *bunsetsu*, which is a group of words consisting one or more content words and zero or more functional words. We use the same input sentence **I1** as an example to show such dependency structure in Figure 3. Each *bunsetsu* has one head which is represented by bold face, followed by a *case marker* such as *ga, wa* or other functional words. Each head bunsetsu is always placed to the right of its modifier and the dependencies do not cross each other. We obtain such Japanese dependency structure by using analyzer Cabocha[4].

We calculate $syn(I, E)$ by comparing syntactic relations r extracted from *bunsetsu* dependencies as:

$$r = w_1 - rel - w_2$$

$$rel = \begin{cases} case\,marker & \textbf{if}\,\text{postpos. follows } w_1 \\ \text{adnominal} & \textbf{elif}\,\text{POS}(w_2) = \text{Noun} \\ \text{adverbial} & \textbf{otherwise} \end{cases}$$

Where w_1 and w_2 are a head of modifier and modifiee *bunsetsus* respectively and rel is the relation type .

[4] http://code.google.com/p/cabocha/

All relations where either w_1 or w_2 is a target word are extracted.

r_1	*tsukama*	−adnominal−	**hanashi**
r_2	**hanashi**	−*wa*−	*kika*

Figure 2: Extracted Relations for Input Sentence **I1**

Figure 2 shows such extracted relations for sentence **I1** from its dependency structure shown in Figure 3. Relations r_1 and r_2 are extracted from sentence **I1** with respect to the target word. Head word *tsukama* (catch) of *bunsetsu* #2 directly modifies bunsetsu #3, where head is the target word "*hanashi*" (story). Further ahead, "*hanashi*" directly modifies bunsetsu #5, therefore head "*kika*" (hear) is extracted as w_2 in r_2.

Next, $syn(I, E)$ is defined as follows.

$$syn(I, E) = \sum_{(r_i, r_e) \in R_I \times R_E} s_r(r_i, r_e) \qquad (1)$$

$$s_r(r_i, r_e) =$$

$$\begin{cases} s_w(r_i(w_1), r_e(w_1)) & \textbf{if}\, r_i(w_2) = r_e(w_2) = t \\ & \quad and\, r_i(rel) = r_e(rel) \\ s_w(r_i(w_2), r_e(w_2)) & \textbf{if}\, r_i(w_1) = r_e(w_1) = t \\ & \quad and\, r_i(rel) = r_e(rel) \\ 0 & \textbf{otherwise} \end{cases} \qquad (2)$$

$$s_w(w_i, w_j) = \begin{cases} 1 & \textbf{if}\, w_i = w_j \\ \frac{x}{8} & \textbf{otherwise} \end{cases} \qquad (3)$$

In Equation (1), $syn(I, E)$ is the sum of similarity scores $\sum s_r(r_i, r_e)$ obtained by comparing all relations r_i and r_e extracted from input and example sentence respectively. Equation (2) chooses to compare two relations of same relation type rel and whose respective target word t is of same dependency structure in both relations i.e either modifier or modifiee. Finally such relations are used to calculate semantic similarity between words $s_w(w_i, w_j)$ as Equation (3). Here w_i and w_j are modifier words from two relations that modifies the target word, vice versa are modifiee words when target word is the modifier. x is the length of common prefix of semantic codes of two words in *Bunrui Goi Hyo* thesaurus[5] (NINJAL 2004).

2.3 Collocation Similarity

$coll(I, E)$ refers to collocation similarity score based on match sequences of n-grams of sizes 4, 5

[5] Note that a semantic code in Bunrui Goi Hyo is represented as 7 digits.

I1 **Han'nin** ga **tsukama**tta toiu $\boxed{\textbf{hanashi}}$ wa **mettani** **kika** nai
(culprit) (got caught) (that) (story) (rarely) (hear)
#1 #2 #3 #4 #5
(Rarely hear a story that the culprit got caught)

Figure 3: Example of Bunsetsu Dependencies

and 6 between sentences I and E. 4-grams are a sequence of 4 words including a target word from a sentence. As shown below, 4 sequences from 4-grams are obtained where $\mathbf{w_0}$ is the target word and w_{-1} and w_1 are previous and next word to the target word, respectively and so on.

w_{-3}-w_{-2}-w_{-1}-$\mathbf{w_0}$

w_{-2}-w_{-1}-$\mathbf{w_0}$-w_1

w_{-1}-$\mathbf{w_0}$-w_1-w_2

$\mathbf{w_0}$-w_1-w_2-w_3

Sequences for 5-grams and 6-grams are defined in the same way. $coll(I, E)$ score by using n-grams is calculated as per Equation (4).

$$coll(I,E) = \begin{cases} 1 & \textbf{if} \;\; \text{one of 6-grams is same} \\ 0.75 & \textbf{elif} \; \text{one of 5-grams is same} \\ 0.5 & \textbf{elif} \; \text{one of 4-grams is same} \\ 0 & \textbf{otherwise} \end{cases} \tag{4}$$

2.4 Using Relations with respect to common words

In Subsection 2.2, only syntactic relations with respect to a target word is considered for the similarity between two sentences. It might be problematic because they seem insufficient to calculate sentence similarities precisely. To use more information for measuring similarity between sentences, we pay attention to common words in two sentences. For syntactic similarity, relations with respect to not only target word but also common words are used to obtain syntactic similarity. That is, in Equation (2), t refers to a target word or a common word. For example, from example sentences shown in Figure 1, there are two common words "*kika*" and "*hanashi*" between **E1** and **I1**. A similarity between "*mettani* - adverbial - *kika*" in I1 and "*mo* - adverbial - *kika*"' in E1 is also added to the score $syn(I1, E1)$. Considering common words to calculate $syn(I, E)$ will naturally affect in an increased recall, but may or may not affect the precision. We show these impacts on precision and recall in Section 3.

3 Evaluation

In this section, we describe the evaluation data and experimental results on different classifiers to evaluate our proposed method.

3.1 Data

For the evaluation, we prepared two sense tagged corpora, a development and an evaluation data. We used the development data (D_d hereafter) to design our example based WSD method. It consists of 390 input sentences of 20 target words (including nouns, verbs and adjectives). While the evaluation data (D_e) is used to measure performance of our proposed method. It consists of 937 input sentences of 49 target words. In both data, input sentences were excerpted from Mainichi Shimbun 1994 articles. The correct senses are manually tagged by authors.

3.2 Classifiers

We checked performances on the following classifiers:

- **RTW:** *Syntactic relations with respect to target word* are compared in this classifier to calculate $syn(I, E)$. RTW chooses the sense associated with the example sentence whose overall similarity score $sim(I, E) = syn(I, E) + coll(I, E)$ is highest.
- **RCW:** As described in Subsection 2.4, *syntactic relations with respect to common words* are compared. Here, RTW is a subset of RCW, as one common word between an input sentence and its example sentence is always the target word.
- **BL:** *Baseline* classifier always selects the sense which has the highest number of example sentences. If more than one senses have same number of example sentences, classifier randomly chooses a sense. This is typically the baseline model when using only example sentences for WSD.
- **RTW + BL and RCW + BL:** Both classifiers RTW and RCT don't choose a sense for a sentence if their $sim(I, E) \leq$ threshold T. In that case sense from the BL classifier is chosen. So, two classifiers are built by combining RTW and RCW with the BL respectively.

Table 1: Results on Development Data D_d

Threshold	RTW				RCW				BL	RTW+BL	RCW+BL
T	P	R	F	A	P	R	F	A	P=R=F	P=R=F	P=R=F
0.00	0.75	0.49	0.59	0.65	0.71	0.56	0.62	0.78	0.64	0.71	0.70
0.25	0.78	0.34	0.48	0.43	0.73	0.47	0.57	0.65	0.64	0.70	0.68
0.50	0.84	0.23	0.37	0.28	0.77	0.38	0.51	0.49	0.64	0.68	0.68
0.75	0.86	0.16	0.28	0.19	0.81	0.31	0.45	0.38	0.64	0.67	0.67
1.00	1.00	0.07	0.13	0.07	0.86	0.26	0.40	0.30	0.64	0.66	0.67

Table 2: Results on Evaluation Data D_e

Threshold	RTW				RCW				BL	RTW+BL	RCW+BL
T	P	R	F	A	P	R	F	A	P=R=F	P=R=F	P=R=F
0.00	0.63	0.43	0.51	0.68	0.60	0.48	0.53	0.80	0.51	0.57	0.57
0.25	0.65	0.31	0.42	0.48	0.63	0.42	0.50	0.66	0.51	0.55	0.57
0.50	0.73	0.21	0.33	0.29	0.66	0.32	0.44	0.49	0.51	0.55	0.56
0.75	0.76	0.14	0.24	0.19	0.69	0.25	0.37	0.36	0.51	0.53	0.55
1.00	0.80	0.06	0.11	0.07	0.71	0.19	0.30	0.27	0.51	0.52	0.53

3.3 Results

Table 1 and 2, reveals the precision P, recall R, F-measure F and applicability A of all the classifiers on development and evaluation data respectively. The applicability is the ratio of the number of instances disambiguated by a classifier to the total number of target instances. For classifiers BL, RTW+BL and RCW+BL, P=R=F and A=1.0, because BL always chooses a sense for all instances.

On D_d, RTW has higher P and lower R, F and A than RCW. RTW's precision is 0.86 with a recall of 0.16 at T=0.75, whereas RCW achieved same precision with around 10% high recall and applicability at T=1.0. Therefore RCW is better than RTW on D_d. On D_e, however, the performances of RTW and RCW are comparable.

The performance of all classifiers on D_e is worse than that on D_d. One of the possible reasons is that WSD of target words in D_e might be more difficult, because (1) there are more senses per target word (4.7 for D_e, while 3.2 for D_d), (2) less example sentences per sense (48.1 for D_e, 67.6 for D_d), (3) the precision of BL is worse.

On both data, precision from BL model is lower than rest of the classifiers at each threshold. Combination of RTW and RCW with baseline decreases the precision, but has higher R, F and A. BL obviously contributes to gain robustness of our example based WSD classifiers. Furtherahead, replacing BL with another better method is our next step.

4 Conclusion

In this paper, we proposed an ensemble of example based WSD and method based on co-occurance statistics as WSD module for reading assistant system for Japanese learners. We presented the results from our first accomplished example based approach. For the classifiers, we used syntactic relations and collocation statistics to measure similarity between two Japanese sentences. We focussed on achieving high precision than recall as our proposed method for future work is combining example base approach with method based on co-occurance statistics. We believe that this ensemble will bring robustness to WSD towards high precision and better recall among unsupervised methods.

References

[1] Ido Dagan and Alon Itai. Word Sense Disambiguation using a Second Language Monolingual Corpus. In *Computational Linguistics*, Vol.20, No.4, pp. 563–596, 1994.

[2] Hyun Ah Lee and Gil Chang Kim. Translation Selection through Source Word Sense Disambiguation and Target Word Selection. In *Proceedings of the 19th international conference on Computational linguistics*, pp. 1–7, 2002.

[3] Kiyoaki Shirai and Takayuki Tamagaki. Word Sense Disambiguation Using Heterogeneous Language Resources. In *First International Joint Conference on Natural Language Processing*, pp. 614-619, 2004.

[4] Natural Institute for Japanese Language and Linguistics, editor. *Bunrui Goi Hyo*, Dainippon Tosho, 2004.