# Word Sense Disambiguation Using Graph-based Semi-supervised Learning

Rie Yatabe
Major in Computer and Information Sciences
Graduate School of Science and Engineering,
Ibaraki University
19nm732r@vc.ibaraki.ac.jp
4-12-1, Nakanarusawa, Hitachi, Ibaraki, 316-8511, Japan

Minoru Sasaki
Dept. of Computer and Information Sciences
Faculty of Engineering, Ibaraki University
minoru.sasaki.01@vc.ibaraki.ac.jp
4-12-1, Nakanarusawa, Hitachi, Ibaraki, 316-8511, Japan

*Abstract*— **Word Sense Disambiguation (WSD) is a well-known problem in the natural language processing. In recent years, there has been increasing interest in applying neural networks and machine learning techniques to solve WSD problems. However, these previous approaches often suffer from the lack of manually sense-tagged examples. Moreover, most supervised WSD methods suffer from small differences of examples within the overall training data or within each of the two sense labels. In this paper, to solve these problems, we propose a semi-supervised WSD method using graph convolutional neural network and investigate what kind of features are effective for this model. Experimental results show that the proposed method performs better than the previous supervised method and the morphological features obtained by the UniDic short-unit dictionary is effective for the semi-supervised WSD method. Moreover, the Jaccard coefficient is the most effective measure among three measures to construct a graph structure.**

***Keywords- word sense disambiguation; graph convolutional neural network; semi-supervised learning.***

## I. INTRODUCTION

In human languages, many words have multiple meanings, depending on the context in which they are used. Identifying the sense of a polysemous word within a given context is a fundamental problem in natural language processing. For example, the English word "bank" has different meanings as "a commercial bank" or "a land along the edge of a river,", etc. Word Sense Disambiguation (WSD) is the task of deciding the appropriate meaning of a target ambiguous word in its context [1].

Among various approaches to the WSD task used over the past two decades, a supervised learning approach has been the most successful. In the supervised learning method, bag-of-words features extracted from a wide context window around the target word are used. However, a common problem of this approach is the lack of sufficient labelled training examples of specific words due to costly annotation work [2].

Moreover, most supervised WSD methods suffer from small differences of examples within the overall training data or within the two sense labels in the whole sense labels. For example, the following two example sentences of the Japanese word "教える (oshieru)" (word ID "5541") have a similar context, but they are used as different meanings.

1. 「そして、仕かけを工夫して、釣り方を教える。」(Sense Label : 5541-0-0-1) (“Then, they teach their customers how to fish using creative fish traps.”)
2. 「1『エルマーのぼうけん』『おばけちゃん』のクイズ大作戦のやり方を教えよう。」(Sense Label : 5541-0-0-2) (“1. I'll show you how to conduct a big plan to take quizzes about the picture books ‘My Father's Dragon’ and ‘Obake-chan’.”)

For these examples, surrounding words can be extracted from the two words, on either side of the target word as follows:

1. ″方″, ″を″, ″教える″, ″。″
2. ″方″, ″を″, ″教えよう″, ″。″

As you can see from these obtained sets of words, almost the same words are contained in both sets. When the difference between the two meanings is small, it is difficult to classify them properly using the existing method. Therefore, if we can distinguish between such example sentences, we can consider improving the performance of WSD systems.

In order to overcome the above problem, semi-supervised learning has been applied successfully to word sense disambiguation. The semi-supervised methods requires only a small amount of sense labelled training examples and can take advantage of unlabelled examples to improve performance. We consider that the semi-supervised learning method is suitable for WSD because a huge amount of unlabelled examples are easily available and the supervised learning methods require a lot of manually sense labelled data. In the semi-supervised learning, we focus on semi-supervised classification method with graph convolutional neural network. This method can jointly train the embedding of an example to predict the sense label of the example and the neighbours in the graph. By using the proposed method, it is possible to incorporate information obtained from unlabelled examples without assigning a sense label to unlabelled examples. Moreover, by learning graph embeddings, it is possible to distinguish between two similar examples with different sense labels to construct a better classifier for WSD. However, it is not clear what kind of features are effective in WSD using the graph convolutional neural network.

In this paper, we investigate what kind of features are effective for graph-based semi-supervised WSD. If we can explore effective features, we consider that it is possible to build a high precision graph-based WSD system. Therefore, this paper aims to find effective features for training WSD classifier using a graph convolutional neural network. Then, we compared the performance for each of the five types of features that include surrounding words and their part of speech in a given window size, local collocations in the context and syntactic properties and so on.

 ISBN: 978-1-61208-813-6

This paper makes mainly two contributions for graph-based semi-supervised WSD as follows:

(1) We employ a graph convolutional neural network for semi-supervised WSD system to incorporate information obtained from unlabelled examples.
(2) We show that it is possible to distinguish between two similar examples with different sense labels using the proposed method.

The rest of this paper is organized as follows. Section 2 is devoted to the related works in the literature. Section 3 describes the proposed semi-supervised WSD method. In Section 4, we describe an outline of experiments and experimental results. Finally, we discuss the results in Section 5 and concludes the paper in Section 6.

## II. Related Works

This section is a literature review of previous work on semi-supervised WSD and various related methods using a neural network.

In recent years, there has been increasing interest in applying neural networks and machine learning techniques to solve WSD problems. [3] employed a Bidirectional Long Short-Term Memory (Bi-LSTM) to encode information of both preceding and succeeding words within the context of a target word. [4] used an LSTM language model to obtain a context representation from a context layer for the whole sentence containing a target word. The context representations were compared to the possible sense embeddings for the target word. Then, the word sense whose embedding had maximal cosine similarity was assigned to classify a target word. [5] considered WSD as a neural sequence labelling task and constructed a sequence learning model for all-words WSD. These approaches are characterized by their high performance, simplicity, and ability to extract a lot of information from raw text.

In recent years, semi-supervised learning has been used in WSD tasks. Semi-supervised learning is a technique that makes use of a small number of sense-labelled examples with a large amount of unlabelled examples. [6] proposed a bootstrapping model that only has a small set of sense-labelled examples that gradually assigns appropriate senses to unlabelled examples. [4] and [7] proposed a semi-supervised WSD method to use word embeddings of surrounding words of the target word and showed that the performance of WSD could be increased by taking advantage of word embeddings.

[8] proposed a semi-supervised WSD method that automatically obtains reliable sense labelled examples using example sentences from the Iwanami Japanese dictionary to expand the labelled training data. Then, this method employs a maximum entropy model to construct a WSD classifier for each target word using common morphological features (surrounding words and POS tags) and topic features. Finally, the classifier for each target word predicts the sense of the test examples. They showed that this method is effective for the SemEval-2010 Japanese WSD task.

Some research in the field of WSD has taken advantage of graph-based approaches. [9] proposed a label propagation-based semi-supervised learning algorithm for WSD, which combines labelled and unlabelled examples in the learning process. [4] also introduced a Label Propagation (LP) for semi-supervised classification and LSTM language model. An LP graph consists of vertices of examples and edges that represent semantic similarity. In this graph, label propagation algorithms can be efficiently used to apply sense labels to examples based on the annotation of their neighbours.

In this paper, we use a semi-supervised learning method that incorporates knowledge from unlabelled examples by using graph convolutional neural network.

## III. WSD Method Using Graph-based Semi-supervised Learning

In this section, we describe the details of the proposed semi-supervised WSD method using a graph convolutional neural network.

### A. Overview of the Proposed Method

Our WSD method is used to select the appropriate sense for a target polysemous word in context. WSD can be viewed as a classification task in which each target word should be classified into one of the predefined existing senses. Word senses were annotated in a corpus in accordance with "Iwanami's Japanese Dictionary (The Iwanami Kokugo Jiten)" [10]. It has three levels for sense Ids, and the middle-level sense is used in this task.

The proposed semi-supervised WSD method requires a corpus of manually labelled training data to construct classifiers for every polysemous word and a graph between labelled and unlabelled examples. For each labelled and unlabelled example, features are extracted from a context around the target word, and the feature vector is constructed. Given a graph structure and feature vectors, we learn an embedding space to jointly predict the sense label and neighbourhood similarity in the graph using Planetoid [11] which is a semi-supervised learning method based on graph embeddings. When the WSD classifier is obtained, we predict one sense for each test example using this classification model.

### B. Preprocessing

To implement the proposed WSD system, we extracted features from training data and test data of a target word, unlabelled examples from the Balanced Corpus of Contemporary Written Japanese (BCCWJ) corpus [12], and example sentences extracted from Iwanami Japanese Dictionary [10]. To segment a sentence into words, we use popular Japanese morphological analyser MeCab with the morphological dictionary UniDic or ipadic.

In this paper, we use the following twenty features (BF) for the target word $w_i$, which is the $i$-th word in the example sentence.

e1: the word $w_{i-2}$
e2: part-of-speech of the word $w_{i-2}$
e3: subcategory of the e2
e4: the word $w_{i-1}$
e5: part-of-speech of the word $w_{i-1}$
e6: subcategory of the e5
e7: the word $w_i$

e8: part-of-speech of the word $w_i$
e9: subcategory of the e8
e10: the word $w_{i+1}$
e11: part-of-speech of the word $w_{i+1}$
e12: subcategory of the e11
e13: the word $w_{i+2}$
e14: part-of-speech of the word $w_{i+2}$
e15: subcategory of the e14
e16: word that contains dependency relation with the $w_i$
e17: thesaurus ID number of the word $w_{i-2}$
e18: thesaurus ID number of the word $w_{i-1}$
e19: thesaurus ID number of the word $w_{i+1}$
e20: thesaurus ID number of the word $w_{i+2}$

To obtain the thesaurus ID number of each word, we use five-digit semantic classes obtained from a Japanese thesaurus "Bunrui Goi Hyo" [13]. When a word has multiple thesaurus IDs, e17, e18, e19, and e20 contain multiple thesaurus IDs for each context word. As additional local collocation (LC) features, we use bi-gram, tri-gram, and skip-bigram patterns in the three words on either side of the target word like IMS [14]. Skip-bigram is any pair of words in an example order with arbitrary gaps. Then, we can represent a context of word $w_i$ as a vector of these features, where the value of each feature indicates the number of times the feature occurs.

To obtain additional example sentences from a dictionary, we use the same extraction method as in the previous work of [8]. In [8], sentences that include an exact match of Iwanami's example for each sense of headword are collected.

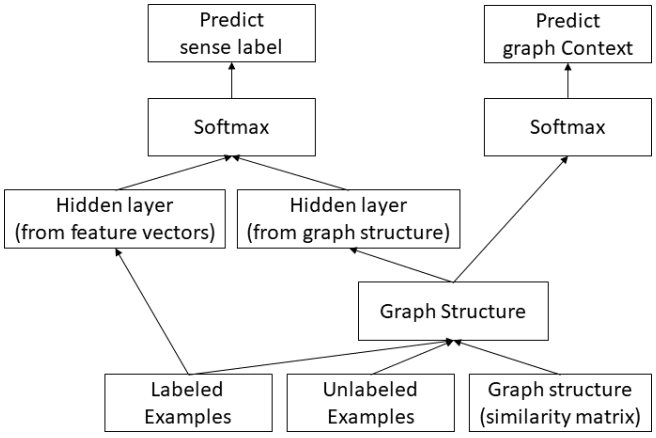


Figure 1. WSD model using graph convolutional neural network

## C. Graph-based Semi-supervised Learning

We employ the Planetoid for the WSD model and predicts the sense of target word. In this method, as shown in Figure 1, we use a set of training examples, unlabelled examples and a graph structure representing the relationship between examples as input and learn a WSD classifier and graph context simultaneously. The classifier predicts the sense of the target word for unknown example.

The training examples and unlabelled examples are represented by feature vectors. The graph structure is constructed from the similarity between the obtained vectors. We learn a WSD model from the training data vector and the graph structure.

Planetoid utilizes stochastic gradient descent (SGD) in the mini-batch mode to train the WSD model. The mini-batch SGD is the popular optimization method for training deep neural networks. The mini-batch SGD is a first order optimization technique which computes the gradient of loss function $L(w)$ with respect to a certain subset of the data points. Using the learning rate $\varepsilon$ and the loss function $L(w)$ of class label and node embedding prediction, the optimal model parameters are obtained by taking the following gradient steps.

$$\boldsymbol{w} = \boldsymbol{w} - \varepsilon(\partial L(\boldsymbol{w})/\partial \boldsymbol{w}). \quad (1)$$

Finally, we predict the appropriate sense label of the target word for the unknown examples using the optimized WSD model.

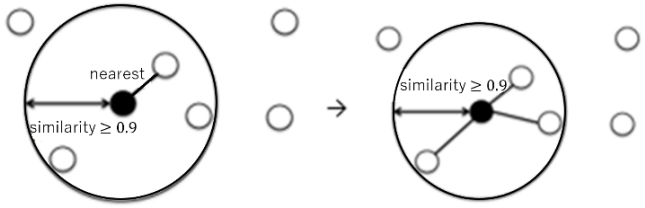


Figure 2. How to connect edges between examples

## D. Input Graph Structure

The input graph structure is constructed by the relation between the training data and the unlabelled data. In the graph structure, each node is an example and an edge is the similarity between nodes. The similarity between nodes is calculated by using the following calculation method between two vectors of examples. In the proposed method, nodes with the highest similarity and nodes that have a similarity greater than the threshold are connected by edge. Figure 2 shows how the edges are connected.

The similarity calculation method between nodes uses Jaccard similarity $J$ or cosine similarity. Jaccard similarity $J$ is the ratio of the number of words in common between the two sets. Given a set of word vectors $A$ and $B$, the similarity $J$ is represented as follows:

$$J(A,B) = |A \cap B|/|A \cup B|, (0 \leq J(A,B) \leq 1). \quad (2)$$

Moreover, we use a mutual $k$-nearest neighbour graph to construct a graph structure. The mutual $k$-nearest neighbour graph is defined as a graph that connects edge between two nodes if each of the nodes belongs to the $k$-nearest neighbours of the other. In this method, the edges with the highest similarity between nodes are also added to the graph structure obtained by the mutual $k$-nearest neighbour graph. In our experiments, we use $k$=3 for the number of neighbours that have been provided by the user.

 ISBN: 978-1-61208-813-6 

## IV. EXPERIMENTS

To evaluate the efficiency of the proposed WSD method using a graph convolutional neural network, we conducted some experiments to compare the results to the baseline system. In this section, we describe an outline of the experiments.

### A. Data Set

We used the Semeval-2010 Japanese WSD task data set, which includes 50 target words comprising 22 nouns, 23 verbs, and 5 adjectives [15]. In this data set, there are 50 training and 50 test instances for each target word.

As unlabelled example data for the construction of a graph structure, we used the BCCWJ developed by the National Institute for Japanese Language and Linguistics. The BCCWJ corpus comprises 104.3 million words covering various genres.

### B. Settings

In our experiments, to construct a graph for all examples, two nodes that represent two examples are linked if they are nearest and if their similarity (based on the Jaccard coefficient) is not less than a specified threshold value of 0.9, which is the highest precision in parameter estimation. The basic idea behind this is that two nodes tend to have a high similarity if the corresponding contexts of the target word are similar.

For learning the graph-based neural network, optimization of the loss function of class label prediction is repeated for 11,000 iterations, and optimization of the loss function of graph context prediction is repeated for 1,000 iterations. Then, the obtained model is used to classify new examples of the target word into semantic classes.

In our experiments, we considered five types of features as follows:

- ipadicBF : word segmentation using dictionary "ipadic" for extracting BF features
- UniDicBF : word segmentation using dictionary "UniDic" for extracting BF features
- UniDicBF+IWA : UniDicBF and additional examples from Iwanami's dictionary
- UniDicBF+LC : UniDicBF and additional local collocation features
- UniDicBF+LC+IWA : UniDicBF, additional local collocation features and additional examples from Iwanami's dictionary

For the Japanese lexical sample WSD task, we compared our method with two previous methods. Firstly, we compared our method with the supervised SVM classifier approach [15]. Secondly, we compared our method with the semi-supervised WSD method that combines automatically labelled data expansion and semi-supervised learning [8].

## V. RESULTS

Table I shows the results of the experiments of applying the proposed method and the two existing methods described in the previous section. The best result per column is printed in bold. As shown in Table I, the proposed method is the highest precision when UniDicBF is used as features. When UniDicBF is used as features, the proposed method is higher than the SVM classifier. However, when we use UniDicBF+IWA, it performs worse than the SVM classifier.

TABLE I. EXPERIMENTAL RESULTS APPLYING THE PROPOSED METHOD AND THE TWO EXISTING METHODS

| Features | Proposed Method | SVM | (Fujita et al., 2011) |
|---|---|---|---|
| ipadicBF | 77.24 | 77.28 | - |
| UniDicBF | **77.76** | 76.8 | 76.56 |
| UniDicBF+IWA | 76.68 | **77.84** | **76.76** |
| UniDicBF+LC | 75.88 | 75.72 | 74.92 |
| UniDicBF+LC+IWA | 76.28 | 77.36 | 76.52 |

TABLE II. EXPERIMENTAL RESULTS WHEN CHANGING THE GRAPH MAKING METHOD

| Jaccard Coefficient | Cosine Similarity | Mutual k-Nearest Neighbour graph |
|---|---|---|
| 77.76 | 77.24 | 77.56 |

TABLE III. CLASSIFICATION PRECISION IN SEMI-SUPERVISED NN AND MAXIMUM ENTROPY AND (FUJITA ET AL., 2011)

| Proposed Method | Maximum Entropy | (Fujita et al., 2011) |
|---|---|---|
| 77.76 | 76.52 | 79.2 |

Table II shows the results of precision among three measurements, the cosine similarity, the Jaccard coefficient, and the mutual k-nearest neighbour using the proposed method with UniDicBF. The results indicate that the Jaccard coefficient measure is the most effective one among all similarity measures with 77.76% precision.

Table III shows the experimental results of both the proposed method with the highest precision and the conventional semi-supervised method [8].

As shown in Table III, the proposed method performs worse than a previous semi-supervised method because the previous method uses the Hinoki Sensebank with UniDicBF+IWA to train a classifier. The Hinoki Sensebank consists of the Lexeed Semantic Database of Japanese [15] and corpora annotated with syntactic and semantic information. Therefore, for a fair comparison, we employed the UniDicBF+IWA features for both methods. As shown in Table I, the proposed method performs better than the previous method.

## VI. DISCUSSIONS

Experimental results show that the proposed method performs better than the SVM classifier. This result was

 ISBN: 978-1-61208-813-6 

obtained by using the proposed method based on the graph-based semi-supervised learning in addition to the conventional supervised method. Therefore, we consider that the proposed method is efficient because it can cope with the lack of labelled data for WSD.

When we use UniDicBF+IWA, the proposed method performs worse than SVM classifier. Example sentences of the Iwanami's Japanese dictionary tend to be connected to short example sentences in the corpus. Therefore, examples of Iwanami's Japanese dictionary tend not to be effective in constructing a graph structure. However, using the SVM classifier, examples of Iwanami's Japanese dictionary are effective for WSD. When we construct a graph structure, we develop a method to utilize the example sentences of the Iwanami's Japanese dictionary effectively in the future.

As shown in Table I, the proposed method using the UniDicBF+LC+IWA performs worse than that using UniDicBF+IWA. The SVM classifier using the UniDicBF+LC+IWA also performs worse. Many examples of the Iwanami's Japanese dictionary are short so that the LC features are not so effective for both methods.

Comparing the features of ipadicBF and the features of UniDicBF, the features of UniDicBF are more effective than the features of ipadicBF. By using UniDic, it is possible to obtain more consistent word segmentation for Japanese sentences of many genres than using ipadic. Therefore, we consider that it is possible to construct an effective graph structure with the UniDic features.

Among the three measurements, the cosine similarity, the Jaccard coefficient, and the mutual k-nearest neighbour, the Jaccard coefficient measure is the most effective of all similarity measures. Thus, if available features are small and dense, the Jaccard coefficient is considered to be suitable for the construction of the graph structure.

Comparing the proposed method with the previous semi-supervised method [8], the proposed method performs worse than the previous method. The previous method uses the basic form (lemma) of the word and the Hinoki Sensebank in addition to the BF features without thesaurus IDs. However, the proposed method does not use the basic form of the word as features (word segmentation) and the Hinoki Sensebank that has 35,838 sentences in 158 senses. Because the features used in the proposed method differ from those used in the previous method, we consider that the features used in the previous method are more effective in comparison to the features used in the proposed method. Therefore, using the UniDicBF+IWA features for both methods for a fair comparison, the proposed method performs better than the previous method. From these results, we consider that the proposed method is more effective in terms of semi-supervised learning for the WSD task.

For the target word "教える (oshieru)," there exist five examples that have similar context, but they have different meanings in the test data. Using the SVM classifier, the classifier could not classify these examples correctly. However, the proposed method was able to classify one test example correctly out of the five examples. To construct the graph structure, the proposed method connects these five examples by the edge. We consider that it is possible to distinguish two examples because the edge between these two examples has been deleted by repeating training with the training examples.

## VII. CONCLUSION

In this paper, we proposed a semi-supervised method using a graph convolutional neural network for the WSD task. The efficiency of the proposed method was evaluated on the Semeval-2010 Japanese WSD task dataset. Experimental results show that the proposed method performs better than the previous supervised method and the morphological features obtained by the UniDic short-unit dictionary is effective for the semi-supervised WSD method. Moreover, the Jaccard coefficient is the most effective measure among three measures to construct a graph structure. Moreover, for the problem with small difference such as examples that have similar context but have different meanings, the proposed method improved the performance of WSD. When the difference between two meanings is small, it is difficult to classify them properly using the existing method for examples that have similar context but have different meanings. Therefore, if we can distinguish such example sentences, we consider the performance of WSD systems improved.

In the future, we would like to explore methods to construct an effective graph structure by using paraphrase information, and the dependency analysis technique, the effective filtering method for unlabelled data. In addition, we would like to develop a method to use the example sentences of the Iwanami's Japanese dictionary effectively.

## ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number 18K11422.

## REFERENCES

[1] R. Navigli, “Word sense disambiguation: A survey”, ACM Comput. Surv. vol. 41, no. 2, article 10, pp. 1–69, February 2009.

[2] H. Shinnou et al., “Classification of Word Sense Disambiguation Errors Using a Clustering Method”, Journal of Natural Language Processing vol. 22, no. 5, pp. 319–362, 2015.

[3] M. Kågebäck and H. Salomonsson, “Word Sense Disambiguation using a Bidirectional LSTM”, Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex-V), pp. 51–56, 2016.

[4] D. Yuan, J. Richardson, R. Doherty, C. Evans, and E. Altendorf, “Semi-supervised Word Sense Disambiguation with Neural Models”. Proceedings of the 26th International Conference on Computational Linguistics (COLING2016), pp. 1374–1385, 2016.

[5] A. Raganato, C. Delli Bovi, and R. Navigli, “Neural sequence learning models for word sense disambiguation”, Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1156–1167, 2017.

 ISBN: 978-1-61208-813-6 

[6] D. Yarowsky, “Unsupervised Word Sense Disambiguation Rivaling Supervised Methods”, Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics pp. 189–196, 1995.

[7] K. Taghipour and H. T. Ng, “Semi-Supervised Word Sense Disambiguation Using Word Embeddings in General and Specific Domains”, Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT-NAACL2015), pp. 314–323, 2015.

[8] S. Fujita and A. Fujino, “Word Sense Disambiguation by Combining Labeled Data Expansion and Semi-Supervised Learning Method”, ACM Transactions on Asian Language Information Processing, vol. 12, no. 2, article 7, pp. 676–685, June 2013.

[9] Z. Niu, D. Ji, and C. L. Tan, “Word Sense Disambiguation Using Label Propagation Based Semi-Supervised Learning”, Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 395—402, 2005.

[10] M. Nishio, E. Iwabuchi and S. Mizutani, “Iwanami Kokugo Jiten Dai Go Han”, Iwanami Publisher, 1994.

[11] Z. Yang, W. W. Cohen and R. Salakhutdinov, “Revisiting Semi-Supervised Learning with Graph Embeddings”, Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48 (ICML'16), pp. 40–48, 2016.

[12] K. Maekawa et al., “Balanced Corpus of Contemporary Written Japanese”, Language Resources and Evaluation (LREC2014), pp. 345–371, 2014.

[13] National Institute for Japanese Language, “Bunrui Goi Hyo (enlarged and revised version)”, Dainippon Tosho, 2004.

[14] Z. Zhong and H. T. Ng, “It Makes Sense: A Wide-Coverage Word Sense Disambiguation System for Free Text”, Proceedings of the ACL 2010 System Demonstrations, pp.78–83, 2010.

[15] M. Okumura, K. Shirai, K. Komiya and H. Yokono, “SemEval-2010 task: Japanese WSD”, Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval ’10), Association for Computational Linguistics, pp. 69–74, 2010.

[16] K. Kasahara et al., “Construction of a Japanese semantic lexicon: Lexeed”. SIG-NL-159, IPSJ, Japan, pp. 75–82, 2004.

 ISBN: 978-1-61208-813-6 