

# Word Sense Disambiguation Based on Word Similarity Calculation Using Word Vector Representation from a Knowledge-based Graph

Dongsuk O<sup>\*1</sup>, Sunjae Kwon<sup>\*2</sup>, Kyungsun Kim<sup>1</sup> and Youngjoong Ko<sup>3</sup>

<sup>1</sup>R&D Center, Diquest Inc., Seoul 08390, Republic of Korea

<sup>2</sup>School of Elec. & Comp. Engineering, UNIST, Ulsan 44919, Republic of Korea

<sup>3</sup>Dept. of Computer Engineering, Dong-A University, Busan 49315, Republic of Korea

<sup>1</sup>ods@diquest.com, <sup>2</sup>soon91jae@unist.ac.kr, <sup>1</sup>kksun@diquest.com,

<sup>3</sup>youngjoong.ko@gmail.com

## Abstract

Word sense disambiguation (WSD) is the task to determine the sense of an ambiguous word according to its context. Many existing WSD studies have been using an external knowledge-based unsupervised approach because it has fewer word set constraints than supervised approaches requiring training data. In this paper, we propose a new WSD method to generate the context of an ambiguous word by using similarities between an ambiguous word and words in the input document. In addition, to leverage our WSD method, we further propose a new word similarity calculation method based on the semantic network structure of BabelNet. We evaluate the proposed methods on the SemEval-2013 and SemEval-2015 for English WSD dataset. Experimental results demonstrate that the proposed WSD method significantly improves the baseline WSD method. Furthermore, our WSD system outperforms the state-of-the-art WSD systems in the Semeval-13 dataset. Finally, it has higher performance than the state-of-the-art unsupervised knowledge-based WSD system in the average performance of both datasets.

## 1 Introduction

In natural language text, it is very common for a word to have more than one sense. For example, in a sentence “An airline hires new cabin crews” of Figure 1, the words ‘airline,’ ‘hires,’ ‘new,’ ‘cabin’ and ‘crews’ have more than two senses. In this case, we can map ‘airline’ to ‘airline Noun#2,’ ‘hires’ to ‘hire Verb#1,’ ‘new’ to ‘new Adjective#6,’ ‘cabin’ to ‘cabin Noun#3’ and ‘crews’ to ‘crew Noun #7’ by the context of this sentence. The word sense disambiguation (WSD) is the task to determine the correct meaning of an ambiguous word in a given context. WSD is being used as a key step for performance improvement in many natural language processing tasks such as machine translation (Vickrey et al., 2005; Chan et al., 2007), information retrieval (Sanderson 1994; Stokoe et al., 2003) and so on.

WSD can be divided into supervised approaches and knowledge-based unsupervised approaches. In the supervised approach, machine learning models are trained by a corpus, in which the correct senses of ambiguous words are already annotated by human annotator (Weissenborn et al., 2015; Melamud et al., 2016; Raganato et al., 2017). However, constructing training corpus for all languages and words is tremendously expensive, so the supervised approaches generally have some limitations on the set of the words that can be disambiguated. On the other hand, the knowledge-based unsupervised approaches utilize lexical knowledge bases (LKBs) such as a **Wordnet** (Banerjee and Pederson., 2003; Chaplot et al., 2015). These approaches have performed WSD by combining **contextual information and semantic knowledge on the LKBs**. Thus, there is much number of words that can be disambiguated when compared to the supervised approaches. **For this reason, it is known that the knowledge-based approach is more suitable than supervised approach for the practical WSD systems (Moro et al., 2014; Chaplot and Salakhutdinov 2018).**

---

\* Both authors contributed equally to this work.

This work is licensed under a Creative Commons Attribution 4.0 International License. License details: <http://creativecommons.org/licenses/by/4.0/>

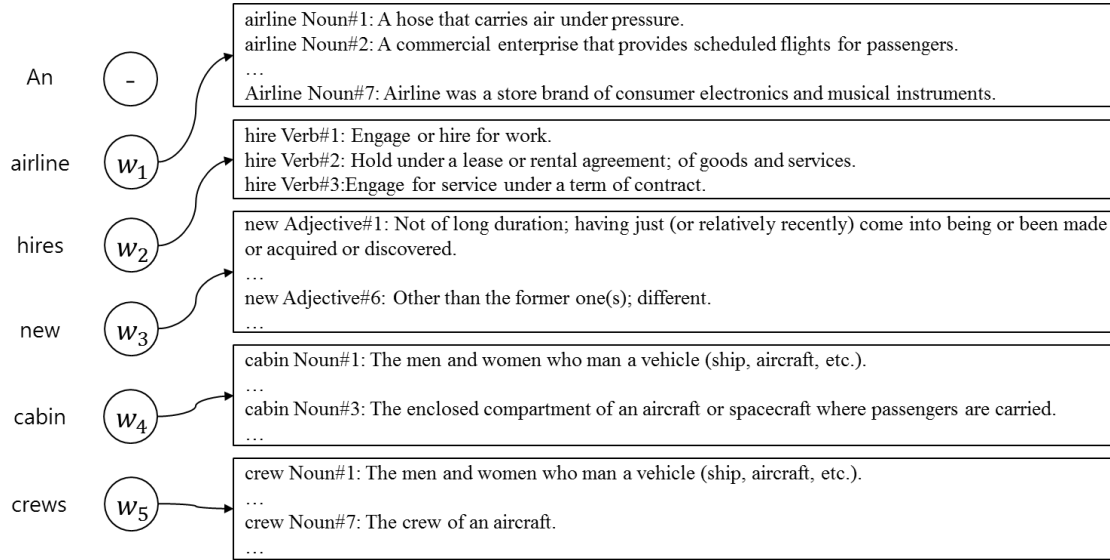


Figure 1: Example for the WSD task.

The most popular approach for the knowledge-based unsupervised ones is the graph-based WSD approach determining the **answer sense** by **linking the senses and context of the ambiguous words on the LKBs to generate a semantic subgraph and measure the connectivity of the senses corresponding to a node in the subgraph**. Therefore, establishing an efficient strategy for generating subgraphs of those senses directly affects the performance of WSD. Conventional knowledge-based unsupervised approaches tried to construct the subgraph of all the words that appear in the document and concurrently disambiguate all ambiguous words in the document (Navigli and Lapata, 2007; Navigli and Lapata, 2010). This joint optimization strategy has the advantage of being able to derive the optimal sense set for ambiguous words, but there is a limitation in that the computational complexity increases exponentially as the number of ambiguous words is increasing. To ameliorate above problem, Manion et al. (2014) proposed a subgraph construction strategy using the iterative subgraph reconstruction approach that greedily analyzes the ambiguous words according to a specific priority. However, the iterative subgraph uses entire words in the document as a context to determine the sense of each ambiguous word and it makes sometimes the subgraph of a word overcomplicated with unnecessary information.

Based on this iterative subgraph reconstruction approach, we propose a new subgraph construction strategy to avoid the abovementioned problems by **selectively restricting the context when constructing a subgraph of an ambiguous word**. In our proposed approach, contextual words of an ambiguous word for constructing the subgraph are selected by thresholding the word similarities to the ambiguous word. In addition, we propose a new word similarity measure method by using word vector representations, generated from the knowledge graph of LKBs and a neural network, for the efficient contextual word selection. In the experiments, we prepared the three publicly accessible English WSD datasets of SemEval-2007 (Pradhan et al., 2007), SemEval-2013 (Navigli et al., 2013) and SemEval-2015 (Moro and Navigli, 2015). We used the SemEval-2007 dataset as a development set for parameter tuning and other datasets as test sets. Experimental results show that our proposed WSD approach with the new subgraph construction strategy has significantly better performance than the baseline iterative subgraph reconstruction approach of Manion et al. (2014). Moreover, when we apply our proposed word similarity method to our WSD approach, it achieved better performance than the WSD systems using other existing word similarity methods. Eventually, the final proposed WSD system with all of the new subgraph construction and word similarity methods achieved higher performance than the state-of-the-art unsupervised knowledge-based WSD systems.

The remainder of the paper organized as follows. In section 2, we introduce previous studies for the WSD system and our sense repository BabelNet (Navigli and Ponzetto, 2012). Section 3 is allocated to introduce our proposed WSD system in detail. Experimental environments and results are described in section 4. Finally, conclusion and future work are discussed in section 5.

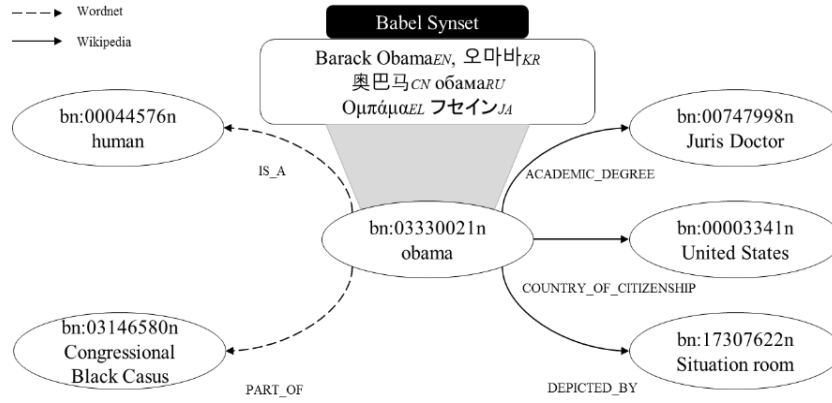


Figure 2: Babelnet example displaying the Babel synsets and the semantic relations (edges) between senses (nodes).

## 2 Related Work

### 2.1 Word Sense Disambiguation

Recently, the graph-based WSD method become the most popular a method for the knowledge-based WSD (Navigli and Velardi, 2005). The graph-based method selects the answer sense of the ambiguous word based on the semantic structure of LKBs. Generally, the answer sense is chosen from the semantic subgraph that connects the senses of the words in the input document using the semantic relationships defined in LKBs. Navigli and Lapata (2007) built a semantic subgraph of the entire words including senses and then used graph connectivity measures to determine the combination of answer senses. Agirre et al. (2014) suggested a knowledge-based WSD approach used personalized page rank (PPR) over the semantic subgraph. They calculated the relative importance of senses using PPR and the sense with the highest score was chosen as an answer sense. Babelfy (Moro et al., 2014) presented another graph-based approach that jointly selects answer of WSD and entity linking (Xiao et al., 2015). Utilizing the random walk algorithm with a restart, it extracted a dense subgraph and reweighted the edges of a BabelNet. They iteratively disambiguate words by reconstructing a semantic subgraph at each word. Based on the assumption that word with a minimum sense is an easiest word among the entire ambiguous words, Manion et al. (2014) disambiguated the ambiguous words in order of the number of their senses. Chaplot et al. (2015) maximized the joint probability of whole senses in the context using WordNet and dependency. Tripodi and Pelillo (2017) suggested to apply the idea of the evolutionary game theory to their WSD system. By exploiting the semantic similarity of the words, they formulated WSD as a constraint satisfaction problem and derived it utilizing game theorem tools.

In addition to these abovementioned methods, several methods for WSD have been proposed. Chaplot and Salakhutdinov (2018) proposed a topic modeling based WSD approach to ameliorate computational complexity of graph-based WSD. Zhong and Ng (2010) tried to disambiguate words using support vector machine (Suykens and Vandewalle, 1999) with rich linguistic features such as part-of-speech, local collocations and surrounding contextual words. Weissenborn et al. (2015) jointly optimized WSD and entity linking model in an extensible multi-objective optimization. Pasini and Navigli (2017) built a large-scale training corpus for WSD from scratch using Wordnet. Raganato et al. (2017) suggested a supervised WSD approach using bidirectional long short-term memory and attention mechanism.

Our WSD method is based on the iterative subgraph reconstruction approach (Manion et al., 2014). However, our WSD approach is crucially different in that it selectively constructs subgraphs by thresholding the contents words of the input document based on the similarity with the ambiguous words.

### 2.2 BabelNet

Most unsupervised WSD systems utilize LKBs such as Wordnet<sup>1</sup> to obtain a set of possible senses for each ambiguous word. BabelNet<sup>2</sup> is a multi-lingual lexicalized semantic network and ontology. It pro-

<sup>1</sup> <https://wordnet.princeton.edu/>

<sup>2</sup> <http://babelnet.org>

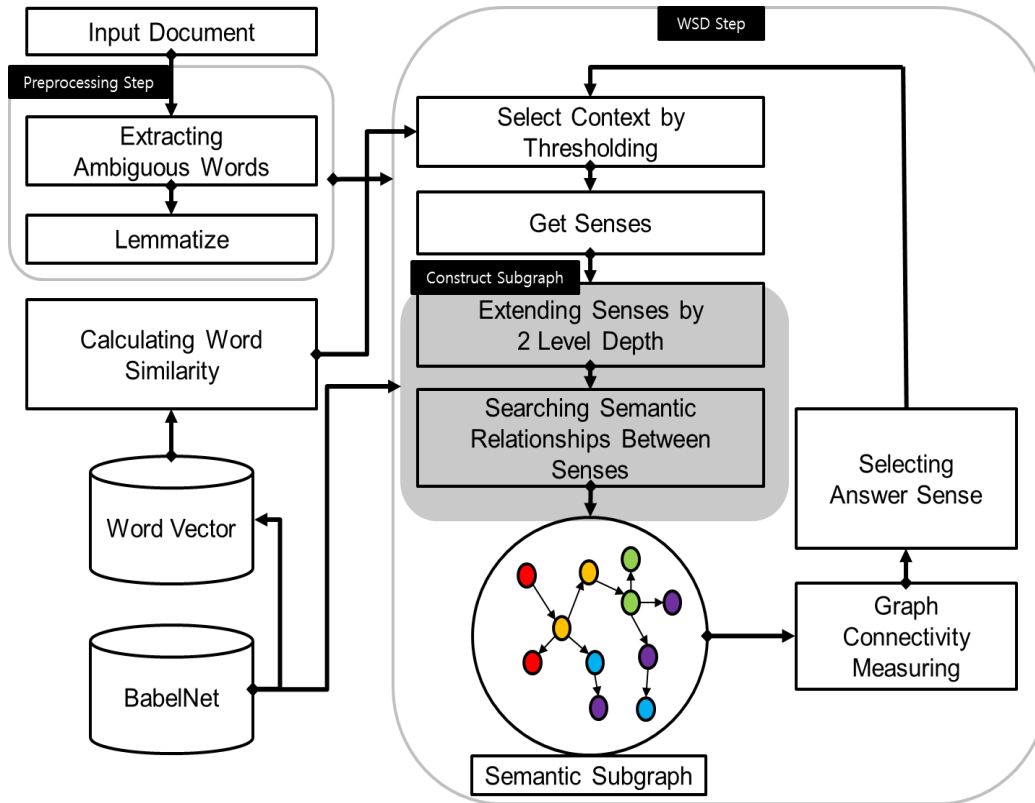


Figure 3: Overall processes of the proposed system.

vides the senses of content words<sup>3</sup>, semantic relationship between the senses and the set of synonyms of the sense.

As shown in Figure. 2, BabelNet has a graph structure that consists of nodes and edges. A node indicates the sense of a word and an edge denotes the semantic relationship between the senses. The synonym information for a sense is accessible from Babel synset, which provides multi-lingual synonyms. Semantic relationship between senses contains the relationship defined by both Wikipedia (ACADEMIC\_DEGREE, COUNTRY\_OF\_CITIZENSHIP, DEPICTED\_BY, etc) and Wordnet (IS\_A, PART\_OF, etc). For example, a word ‘Obama’ contains six possible noun senses such as “bn:03330021n: ‘Barack Hussein Obama II is an American politician who served as the 44th President of the United States from 2009 to 2017.’,” and an adjective sense of “bn:13705874a: ‘Of or pertaining to the political figure and 44th president of the United States of America Barack Obama.’” In addition, ‘Obama’ with a noun sense of ‘bn: 03330021n’ has an ‘IS-A’ semantic relationship with ‘Human’ with a noun sense of ‘bn: 00044576n’ and has a semantic relationship of ‘COUNTRY\_OF\_CITIZENSHIP’ with ‘United States’ with a noun sense of ‘bn: 00003341n.’

As mentioned above, BabelNet provides the senses and their semantic relationships to multi-lingual, it is advantageous to extend a WSD system to other language. This makes many recent studies for WSD systems choosing BabelNet as a sense repository (Moro et al., 2014; Manion et al., 2014; Tripodi and Pelillo 2017).

### 3 Proposed WSD System

This section describes our fundamental ideas to improve the performance of WSD and how they are integrated into our WSD system as illustrated in Figure 3. The subsection 3.1 is allocated to introduces a novel word similarity calculation method for the contextual word selection. Next, we explain our new iterative subgraph construction method that combine contextual word selection in the traditional WSD approach.

<sup>3</sup> Verbs, adverbs, nouns and adjectives

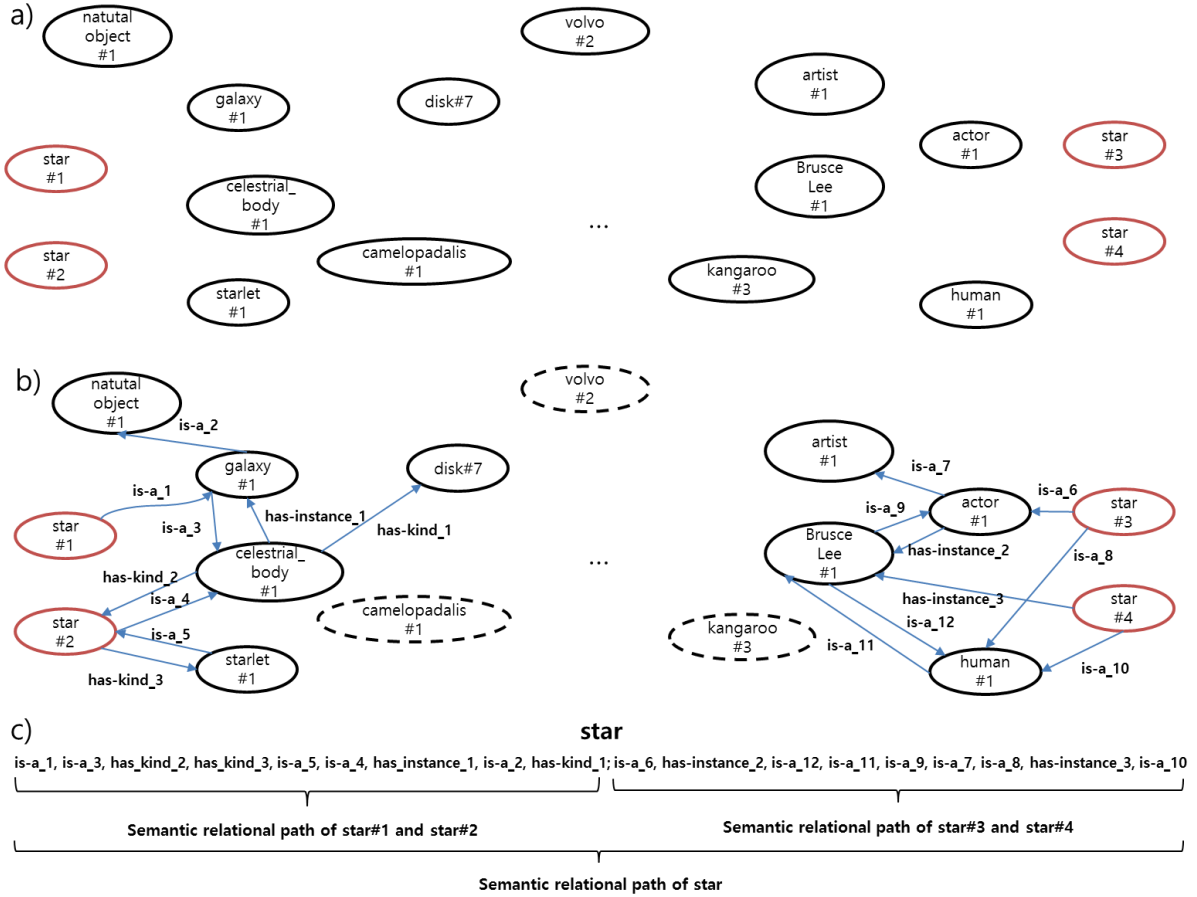


Figure 4: Example of the semantic relational path extracting process of a word 'star'. An ellipse means a sense and an arrow indicates the semantic relationship between the senses. a) Initial state, b) Extending the senses by 2 level depth and c) Generating Semantic relational path of 'star' using DFS algorithm.

### 3.1 Word Similarity Calculation Through the Word Vector Representation from the BabelNet Graph Structure for the Contextual Words Selection

In order to determine the correct sense of the ambiguous words in the WSD, it is very important to use the context around to find the sense of the ambiguous word. However, not all the contextual words are equally important for WSD (Karov and Edelman, 1998). If we can choose more important contextual words for each ambiguous word in the document, then we can more accurately disambiguate the sense of an ambiguous word by reducing unnecessary information. From this point of view, we assume that the higher a word has similarity to the ambiguous word, the more it can contribute to determining the sense of an ambiguous word.

In WSD, we determine the sense of a word based on context. At this time, the context in which we are interested is the theme of the entire document or sentence. For this reason, semantic information such as theme words will have a much more impact on WSD than syntactic information such as part-of-speech or sentence component information. Under this assumption, we propose a novel method of generating the word vector representations for the semantic information using knowledge graph structure as follows.

The senses of a word can be connected by a series of the semantic relationships in the semantic network. Figure 4 illustrates an example of representing a word as the sequence of the semantic relationships by connecting the senses of the word on the BabelNet knowledge graph. In Figure 4 (a), a noun word 'star' has four different senses; star#1: "A celestial body of hot gases", star#2: "Any celestial body visible from the Earth at night.", star#3: "An actor who plays a principal role" and star#4: "A widely known person." To connect these senses, we extend the senses by  $L$  level depth around each sense of the word as shown in Figure 4 (b). In this example, there are two connected graphs are generated. The



first one connects the extended senses of star#1 and star#2 and the second one does those of star#3 and star#4. In this example, **each subgraph is related to the common theme of the connected senses**. A graph connecting star#1 and star#2 contains common senses associated with astronomical phenomena. On the other hand, a graph connecting star#3 and star#4 consists of senses related to human or occupation. As shown in the Figure 4 (c), **the DFS algorithm is applied to easily handle subgraphs that make up the meaning of a word**. By the DFS algorithm, we can represent each graph connecting the sense of a word as a sequence of semantic relationships, which we refer to the semantic relational path. Finally, the concatenated semantic relational path of all subgraphs is considered as the overall representation of the word (see Figure 4 (c)).

The semantic relational path of a word from the previous paragraph consists of three structures: relations that connect senses, subgraph that is connected with the senses and sharing a theme and words that are represented as a set of subgraphs. **If we match a relation to a word and a subgraph to a sentence in this structure, we can regard the semantic relation path of word as a kind of a pseudo-document**. To effectively encode information of the semantic relational path of word, we used Doc2vec (Le and Mikolov, 2014). Doc2vec is an unsupervised learning algorithm that generates a document vector based on words contained in the document. The vectors of documents having a similar meaning are projected into the similar vector space. In our case, the semantic relational path, as a pseudo-document, of the word is an input and the word vector representation of the word is an output of the Doc2vec. Thus, if words sharing semantic relationships will be projected into the similar vector space, otherwise they will be projected in the totally different vector space. To measure the similarity of the words by regarding both distance and direction, the cosine similarity measure of Eq.1 is used to calculate the word similarity of the vector representation of the words  $w_1$  and  $w_2$ .

$$word\_similarity(w_1, w_2) = \frac{w_1 \cdot w_2}{||w_1|| \times ||w_2||} \quad (1)$$

### 3.2 Iterative Subgraph Reconstruction Approach with Word Similarity-based Context Words Selection

In order to determine the correct sense of ambiguous words in a graph-based WSD approach, it is essential to establish an efficient strategy for constructing a subgraph that connects the correct senses by taking into account the structure of the semantic network and words in the context. In our study, we suggest a new WSD strategy that constructs a set of context words that have a similarity value to the ambiguous word beyond a threshold.

---

#### Proposed Word Sense Disambiguation System

---

**Input:** input document (*Input*)

**Output:** disambiguated sense (*Answer*)

- 1:  $I \leftarrow \text{Extracting\_ambiguous\_words}(\text{Input})$
  - 2:  $\mathcal{L} \leftarrow \text{Lemmatize}(I)$
  - 3:  $\text{Answers} \leftarrow \emptyset$
  - 4: **For**  $l_i$  in  $\mathcal{L}$  **do**
  - 5:    $C_i \leftarrow \text{SelectContext}(l_i, \text{Other\_Words})$
  - 6:    $S_{l_i} \leftarrow \text{GetSenseSet}(l_i)$
  - 7:    $S_{C_i} \leftarrow \text{GetSenseSet}(C_i)$
  - 8:    $G_i \leftarrow \text{ConstructSubGraph}(S_{l_i}, S_{C_i}, \text{Answer})$
  - 9:    $\hat{s}_* \leftarrow \text{argmax}_{s_j \in S_{l_i}} \phi(G_i, S_{l_i})$ , where  $\phi$  indicates graph connectivity
  - 10:   put  $\hat{s}_*$  in *Answer*
  - 11: **Return** *Answer*
- 

Our WSD system is made up of two steps: the preprocessing step (line 1 to 3) and the WSD step (line 4 to 11). In the preprocessing step, we extract the sequence of the ambiguous words  $I = \{w_1 \dots w_m\}$  in an input document, *Input*, and the order of sequence follows to the occurrence order of the ambiguous

words in the input document. The sequence of ambiguous words,  $I$ , is mapped to their lemmatized form,  $\mathcal{L} = \{l_1 \dots l_m\}$  and the answer set of disambiguated senses from our WSD system,  $Answer$ , is initialized by a null set. In the WSD step, our proposed WSD method iteratively determine the answer sense in order of  $\mathcal{L}$ . To do this, it first selects the contextual words,  $C_i$ , of an ambiguous word,  $l_i$ , by measuring similarities between the ambiguous word and other words in the document. To do this, our system calculates all the similarities between  $l_i$  and other words. Words whose similarity to  $l_i$  exceeds a threshold are selected as  $C_i$ . If there is no word exceed the threshold, its context is created by choosing a word that has the highest similarity (line 5). Then the senses ( $S_{l_i}$  and  $S_{C_i}$ ) of  $l_i$  and  $C_i$  are extracted from BabelNet (line 6 & 7). Next, the whole senses of  $S_{l_i}$ ,  $S_{C_i}$  and  $Answer$  are extended by the depth of level  $L$  and they are connected as the semantic relation by the depth-first search (DFS) algorithm (line 8). Finally, the graph connectivity of the sense,  $s_j$ , is calculated by the PPR algorithm (Gutiérrez et al., 2013) and the sense with the highest connectivity is selected as answer sense (line 9 and 10). This process is repeated until no more ambiguous words remained in the set of  $\mathcal{L}$  (line 4 to 11).

## 4 Experiments

### 4.1 Datasets

We evaluated our WSD system on the three publicly available English WSD corpora: SemEval-2007, SemEval-2013 and SemEval-2015.

- The SemEval-2007 dataset consists of three documents with 465 noun and verb words annotated with Wordnet entries. In our experiment, 414 words in the BabelNet entry were selected and this dataset was used for the development set.
- The SemEval-2013 dataset consists of 13 news articles, including various domains from 2010 to 2012. All the noun words were annotated and there are 1,931 words to be disambiguated.
- The SemEval-2015 dataset consists of four documents from several heterogeneous domains. This dataset annotated WSD and entity linking tasks at the same time. The answer senses were annotated for all content words in the dataset and there are 1,261 words to be disambiguated.

### 4.2 Experimental Settings

Gensim Doc2vec library<sup>4</sup> was used to generate word vector representation introduced in the subsection 3.2. The dimension of the word vector was allocated to 200 and window size is set to 3. In addition, we set the initial learning rate of Doc2vec to 0.5. All the other Doc2vec parameters were set to default. Finally, threshold for the contextual words was set to 0.5. The resources for our word vector revector representation is available at <https://github.com/nlpbank/SRP2Vec>. Furthermore, we set up the hyperparameters of our system at the highest score on the SemEval-2007 dataset as the development set.

As a performance evaluation measure of the WSD systems, we used a  $F_1$ -score criteria of Eq.2 that is a harmonic mean of precision of Eq.3 and recall of Eq.4. Besides, to determine statistically significant difference between the performance of the system, we carried out macro student  $t$ -test (Yang and Liu, 1999).

$$Precision = \frac{\# \text{ of correctly disambiguated answers}}{\# \text{ of words that outcome is positive}} \quad (2)$$

$$Recall = \frac{\# \text{ of correctly disambiguated answers}}{\# \text{ of true answers}} \quad (3)$$

$$F_1 - \text{score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

<sup>4</sup> <https://radimrehurek.com/gensim/models/doc2vec.html>

### 4.3 Experimental Results

To verify the effectiveness of our proposed WSD system, we compared our WSD approach (*Wordsim\_iter*) to a baseline WSD approach (*Sudoku\_iter*) (Manion et al., 2014). The results of the SemEval-2013 and SemEval-2015 datasets are the same as those reported by Manion et al, (2014). On the other hand, because Manion et al, (2014) have not reported score on the SemEval-2007 dataset, we re-implemented and evaluated *Sudoku\_iter* on the SemEval-2007 dataset. Furthermore, in order to verify our hypothesis that semantically similar words are more important than syntactically similar words in the WSD task, we compare our word similarity (*SRP2vSim*) calculation method with an existing word vector representation-based similarity calculation method as follows:

- *W2vSim*: It is based on the Word2vec that an unsupervised algorithm to generate word vector representation (Mikolov et al., 2013). We obtained word vector representations in the official word2vec website<sup>5</sup>. The word vector representation pretrained in the general text corpus has both the semantic and syntactic features of the words (Mikolov et al., 2013).

WSD Approach	Word Similarity Measurement	Dev			Test					
		SemEval-2007			SemEval-2013			SemEval-2015		
		Prec.	Rec.	$F_1$	Prec.	Rec.	$F_1$	Prec.	Rec.	$F_1$
<i>Sudoku_iter</i>	—	52.9	52.9	52.9	67.2	67.2	67.2	60.8	59.1	59.9
<i>Wordsim_iter</i>	<i>W2vSim</i>	48.5	48.5	48.5	72.1	72.1	72.1	57.8	51.1	54.2
	<i>SRP2vSim</i>	<b>56.1</b>	<b>56.1</b>	<b>56.1</b>	<b>75.0</b>	<b>75.0</b>	<b>75.0</b>	<b>69.2</b>	<b>62.6</b>	<b>65.8</b>

Table 1: Performance comparison of our WSD systems with the baseline WSD system (*Sudoku\_iter*).

Table 1 shows that our proposed WSD system, *Wordsim\_iter*, achieved the significantly improved performance compared to our baseline system, *Sudoku\_iter* ( $p$ -value  $< 0.01$ ), in all of the datasets. From these results, we are able to confirm that WSD performance can be improved by ignoring the words that are less similar with the ambiguous word. It also proved that WSD performance is determined by the criteria for selecting the contextual words. In case of word similarity measurement, our WSD system using *SRP2vSim* achieved significant improvement in performance compared to one using *W2vSim* ( $p$ -value  $< 0.05$ ). Meanwhile, when we used *W2vSim* in the SemEval-2015 dataset, its result was significantly lower than the baseline ( $p$ -value  $< 0.01$ ). On the other hand, *Wordsim\_iter* with *SRP2vSim* has higher performances (4.5 %p and 10 %p) than *Sudoku\_iter* and *Wordsim\_iter* with *W2vSim*, respectively. Through these results, we think that the word vector representation from the knowledge-based graph is more adaptable than the traditional word vector representation for WSD tasks because the semantic information is more important than the syntactic information of words in the WSD tasks.

Approach	System	Semeval-2013	Semeval-2015	Macro Avg $F_1$
Unsupervised (Knowledge-based)	Moro 14	66.4	<b>70.3</b>	68.4
	Agirre 14	62.9	63.3	63.1
	Apidianaki 15	—	64.7	—
	Tripodi 17	70.8	—	—
	<i>Wordsim_iter</i> <sub><i>SRP2vSim</i></sub>	<b>75.0</b>	65.8	<b>70.4</b>
Supervised	Zhong 10	66.3	69.7	68.0
	Weissenborn 15	71.5	<b>75.4</b>	<b>73.5</b>
	Raganato 17	66.9	71.5	69.2
	Pasini 17	65.5	68.6	67.1

Table 2: Performance comparison of our WSD system with state-of-the-art BabelNet-based unsupervised and supervised WSD systems.

In Table 2, we can compare our WSD approach using the *Wordsim\_iter* method and *SRP2vSim* word similarity measurement (*Wordsim\_iter*<sub>*SRP2vSim*</sub>) to other Knowledge-based WSD systems introduced in

<sup>5</sup> <https://github.com/mmihaltz/word2vec-GoogleNews-vectors>



"[Alimta] is a [powder] that is [made\_up] into a [solution] for [infusion] (drip into a [vein])."

<p>Alimta: Chemotherapy drug manufactured and marketed by Eli Lilly and Company.  powder: A solid substance in the form of tiny loose particles.  make up: Form or compose  solution: A homogeneous mixture of two or more substances  infusion: The passive introduction of a substance into a vein or between tissues  vein: A blood vessel that carries blood from the capillaries toward the heart</p>
--

Figure 5: An example sentence and definitions of the correct word senses. The term ‘[]’ denotes an ambiguous word.

the Section 2, such as Moro 14 (Moro et al., 2014), Agirre 14 (Agirre et al., 2014), Apidianaki 15 (Apidianaki and Gong, 2015) and Tripodi 17 (Tripodi and Pelillo, 2017). In addition, we compared Wordsim\_iter<sub>SCP2vSim</sub> to several supervised WSD systems, such as Zhong 10 (Zhong and Ng, 2010), Weissenborn 15 (Weissenborn et al., 2015), Raganato 17 (Raganato et al., 2017) and Pasini 17 (Pasini and Navigli., 2017). The results show that our WSD system surpassed all other state-of-the-art WSD systems with a large margin in the SemEval-2013 dataset. In the SemEval-2015 dataset, our WSD system has similar performance to the Agirre 14 and Apidianaki 15 and it has somewhat less performance than the state-of-the-art knowledge-based WSD system, Moro 14. This is due to the nature of the SemEval-2015 data, Moro 14 is designed to simultaneously analyzes WSD and entity linking. However, in terms of the macro average score of SemEval-2013 and SemEval-2015, Wordsim\_iter<sub>SCP2vSim</sub> shows higher performance than the Moro 14. On the other hands, unsupervised knowledge-based approaches, including our system, generally has poorer performance than supervised approaches in the SemEval-2015 dataset. Especially, Weissenborn 15, a hybrid supervised WSD system that jointly has trained the WSD model and the entity linking model, achieved higher performance on macro average than our WSD system. Nevertheless, fewer limitation on the analyzable word set of our model makes it relatively more competitive than its counterpart state-of-the-art supervised-based WSD models.

#### 4.4 Error Analysis

Despite the competitiveness of our system, the greedy algorithmic characteristic of the iterative subgraph-based algorithm has a negative effect on its performance. In particular, some previously analyzed words can affect other words and it makes them mis-disambiguated. For example, in the sentence of Figure 5, there are 6 ambiguous words: ‘Alimta’, ‘powder’, ‘made up’, ‘solution’, ‘infusion’ and ‘vein’. Our WSD system wrongly determined ‘made up’ as "Apply make-up or cosmetics to one's face to appear prettier" because this wrong sense is more related with a previously determined sense ‘powder’ than correct sense in Figure 5. In addition, the mis-disambiguated ‘made up’ and previously analyzed words of ‘powder’ and ‘solution’ leads the meaning of ‘infusion’ as a wrong sense "A solution obtained by steeping a substance." If we decide the meaning of ‘made up’ and ‘infusion’ by regarding a word ‘vein,’ then we can disambiguate the words ‘made up’ and ‘infusion’ correctly.

### 5 Conclusions and Future Work

In this paper, we propose a knowledge-based WSD method that restricts contextual words based on the similarities between the ambiguous words and content words. We first measure the similarities of the words in an input document and ambiguous word and selectively create context of the ambiguous word with the words over the certain threshold. In addition, we further suggest a novel similarity calculating method suitable for our WSD method. Our WSD system significantly improves a baseline WSD system and has a higher performance than the state-of-the-art unsupervised knowledge-based WSD systems.

Our WSD system is based on an iterative subgraph reconstruction approach that determines the sense of a word in order. This method has been proposed to solve the computational complexity problem of finding the optimal combination among all possible set of senses. However, due to the nature of the greedy search, sometimes it makes hard to inference correct sense of the ambiguous word because of the error propagation from a previous result can determined answer.

For the future work, we plan to extend our WSD system to the multi-lingual system. In particular, we are going to research to generate of multi-lingual word vector representation using the Babel synset information. Another possible future work is to use the Beam search (Ow and Morton, 1988) to compensate for the drawbacks of the iterative subgraph reconstruction’s greedy algorithmic characteristics.

## Acknowledgements

This work was supported by the Korea Evaluation Institute of Industrial Technology grant funded by the Korea government (No. 10080681, Technical development of Korean speech recognition system in vehicle), Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (2017-0-00255, Autonomous digital companion framework and application), and this research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (NRF-2015R1D1A1A01056907).

## References

- Marianna Apidianaki, and Li Gong. 2015. LIMSI: Translations as Source of Indirect Supervision for Multilingual All-Words Sense Disambiguation and Entity Linking. *In the proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pp. 298-302.
- Eneko Agirre, Oier López de Lacalle, and Aitor Soroa. 2014. Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, Vol. 40, No. 1, pp. 57-84.
- Satanjeev Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. *In proceedings of the IJCAI*, pp. 805–810.
- Yee Seng Chan, Hwee Tou Ng, and David Chiang. 2007. Word sense disambiguation improves statistical machine translation. *In proceedings of the 45th annual meeting of the association of computational linguistics*, pp. 33-40.
- Devendra Singh Chaplot, Pushpak Bhattacharyya, and Ashwin Paranjape. 2015. Unsupervised Word Sense Disambiguation Using Markov Random Field and Dependency Parser. *In proceedings of the AAAI*, pp. 2217-2223.
- Devendra Singh Chaplot, and Ruslan Salakhutdinov. 2018. Knowledge-based Word Sense Disambiguation using Topic Models. *arXiv preprint arXiv:1801.01900*.
- Yoan Gutiérrez, et al. 2013. UMCC\_DLSI: reinforcing a ranking algorithm with sense frequencies and multidimensional semantic resources to solve multilingual word sense disambiguation. *In proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pp. 241-249.
- Yael Karov, and Shimon Edelman. 1998. Similarity-based Word Sense Disambiguation. *Computational Linguistics*, Vol. 24, No. 1, pp. 41-59.
- Quoc Le, and Tomas Mikolov. 2014. Distributed representations of sentences and documents. *In proceedings of the International Conference on Machine Learning*, pp. 1188-1196.
- Xiao Ling, Sameer Singh, and Daniel S. Weld. 2015. Design challenges for entity linking. *Transactions of the Association for Computational Linguistics*, Vol. 3, pp. 315-328.
- Steve L. Manion, and Raazesh Sainudiin. 2014. An Iterative ‘Sudoku Style’ Approach to Subgraph-based Word Sense Disambiguation. *In proceedings of the Third Joint Conference on Lexical and Computational Semantics*, pp. 40-50.
- Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional lstm. *In proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pp. 51-61.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeffery Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. *In proceedings of the 26<sup>th</sup> Advances in Neural Information Processing Systems*, pp. 3111-3119.
- Andrea Moro, Alessandro Raganato. and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, Vol. 2, pp. 231-244.

- Andrea Moro, and Roberto Navigli. 2015. Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. *In proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pp. 288-297.
- Roberto Navigli, and Paola Velardi. 2005. Structural semantic interconnections: a knowledge-based approach to word sense disambiguation. *IEEE transactions on pattern analysis and machine intelligence*, Vol. 27, No. 7, pp. 1075-1086.
- Roberto Navigli, and Mirella Lapata. 2007. Graph Connectivity Measures for Unsupervised Word Sense Disambiguation. *In proceedings of the IJCAI*, pp.1683-1688.
- Roberto Navigli, and Mirella Lapata. 2010. An experimental study of graph connectivity for unsupervised word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.32, No.4, pp.678-692.
- Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, Vol. 193, pp. 217-250.
- Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. Semeval-2013 task 12: Multilingual word sense disambiguation. *In proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pp. 222-231.
- Tommaso Pasini, and Roberto Navigli. 2017. Train-O-Matic: Large-Scale Supervised Word Sense Disambiguation in Multiple Languages without Manual Training Data. *In proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 78-88.
- Sameer S. Pradhan, Edward Loper, Dmitriy Dligach and Martha Palmer. 2007. SemEval-2007 Task 17: English Lexical Sample, SRL and All Words. *In proceedings of the 4th International Workshop on Semantic Evaluations*, pp. 87-92.
- Alessandro Raganato, Claudio Delli Bovi, and Roberto Navigli. 2017. Neural sequence learning models for word sense disambiguation. *In proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1156-1167.
- Mark Sanderson. 1994. Word sense disambiguation and information retrieval. *In proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 142-151.
- Volker Steinbiss, Bach-Hiep Tran, and Hermann Ney. 1995. Improvements in beam search. *In proceedings of the third International Conference on Spoken Language Processing*, pp. 2143-2146.
- Christopher Stokoe, Michael P. Oakes, and John Tait. 2003. Word sense disambiguation in information retrieval revisited. *In proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 159-166.
- Johan AK Suykens, and Joos Vandewalle. 1999. Least squares support vector machine classifiers. *Neural processing letters*, Vol. 9, No. 3, pp. 293-300.
- Egidio Terra, and Charles LA Clarke. 2003. Frequency estimates for statistical word similarity measures. *In proceedings of the 2003 NAACL-HLT*, pp. 165-172.
- Rocco Tripodi, and Marcello Pelillo. 2017. A game-theoretic approach to word sense disambiguation. *Computational Linguistics*, Vol. 9, No. 3, pp. 31-70.
- David Vickrey, Luke Biewald., Marc Teyssier, and Daphne Koller. 2005. Word-sense disambiguation for machine translation. *In proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 771-778.
- Dirk Weissenborn, Leonhard Hennig, Feiyu Xu and Hans Uszkoreit. 2015. Multi-objective optimization for the joint disambiguation of nouns and named entities. *In proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 596-605.
- Yiming Yang and Xin Liu. 1999. A re-examination of text categorization methods, *In proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 42-49.
- Zhi Zhong, and Hwee Tou Ng. 2010. It makes sense: A wide-coverage word sense disambiguation system for free text. *In proceedings of the ACL 2010 system demonstrations*, pp. 78-83.