Unsupervised Graph-based Word Sense Disambiguation Using lexical relation of WordNet

Ehsan Hessami[1], Faribourz Mahmoudi[2] and Amir Hossien Jadidinejad[3]

[1] Islamic Azad University, Qazvin Branch,
Qazvin, Iran

[2] Islamic Azad University, Qazvin Branch,
Qazvin, Iran

[3] Islamic Azad University, Qazvin Branch,
Qazvin, Iran

Abstract

Word Sense Disambiguation (WSD) is one of tasks in the Natural Language Processing that uses to identifying the sense of words in context. To select the correct sense, we can use many approach. This paper uses a tree and graph-connectivity structure for finding the correct senses. This algorithm has few parameters and does not require sense-annotated data for training. Performance evaluation on standard datasets showed it has the better accuracy than many previous graph base algorithms and decreases elapsed time.

Keywords: *word sense disambiguation, tree, Graph connectivity.*

1. Introduction

The objective of word sense disambiguation is identifying the correct sense of word. Since Human language includes many ambiguity words. WSD is one of the essential tasks in the most Natural Language Processing (NLP), including information retrieval, information extraction, question answering and machine translation. For instance, the term of *bank* has two senses: *finance* and *shore*. The correct sense of an ambiguous word can be selected based on the context where it occurs. The problem is defined as the task of automatically assigning the appropriate sense to polysemous word at given context.

The methods of word sense disambiguation can be classified in Supervised and Unsupervised. The supervised approaches have the better performance than unsupervised approaches [1,2], the supervised systems accuracy are between 60 and 70 percent and the unsupervised systems are between 45 and 60 percent [2]. But often require large amounts of training data to yield reliable results and their coverage is typically limited to the some words. Unfortunately, creating a suitable train-data which is including all the human language words and sense are too difficult, expensive and must be reiterated for new domains, Words, and sense inventories. Therefore, these approaches have many problems. As an alternative to supervised systems, knowledge-based WSD systems extract the suitable information and present in a lexical knowledge base to perform WSD, without using any further corpus evidence. The unsupervised methods can be used this lexical knowledge-based to WSD.

In the field of WSD, the unsupervised approaches are used to methods that perform sense disambiguation without need to train data. The unsupervised approaches divided in two classes: graph-based [6,11,14,15,16] and similarity-based [3,8,12]. Graph-based algorithms often have two steps. First, construct semantic graphs from words of context, and then process the graph in order to select the best sense for each of words. Similarity-based algorithms assign a sense to an ambiguous word by comparing each of its senses with those of the surrounding context, then select the sense has highest similarity. Experimental comparisons between the two algorithm types indicate that graph-based algorithms have better performance than similarity-based [5].

This paper, describes a different graph-base algorithm for unsupervised word sense disambiguation, builds a tree and finds the best Edges. Then builds a graph with the edges and uses the connectivity measure methods for extract the best sense of each word. Also uses the WordNet efficiently, performing significantly better that previously published approaches in English all-words datasets. Show that the algorithm has good results, also present some condition for receiving the better result, performance and time consuming.

The paper is organized as follows. We first describe Related work and followed by WordNet. Section 4 describes proposed algorithm. Section 5 shows the experimental setting and the main results, finally we conclude with a discussion of the conclusion and future works.

2. Related Work

In this section, briefly describe some graph-based methods for knowledge-based WSD. All the methods rely on the information represented on some lexical knowledge base, which typically is some version of WordNet, sometimes enriched with proprietary relations. The results on datasets show in Table 2.

Mihalcea [13] presented an approach that used the PageRank algorithm to identify sense which is relevant in context. Initially, builds a graph from the possible senses of words in a text and interconnects pairs of senses with meaningful relations by WordNet. Graph edges have weight. The weight of the links joining two synsets is calculated by executing Lesk's algorithm between them. Then, use the application of PageRank for selecting the best sense of each word. The PageRank computations require several alternatives through the graph to achieve the suitable ranking for sense of word.

Navigli and Velardi [4] presented the *Structural Semantic Interconnections*(SSI) algorithm, that offered method for development of lexical chain base on the encoding of a context free grammar of valid semantic interconnection patterns. To find the meaning of the words in WordNet glosses used, but can be used for English-all words, though has the weakly accuracy. Given a text sequence, first identifies ambiguity words and builds a sub graph of the WordNet lexicon which includes all the senses of words. Then, select the senses for the words which maximize the degree of connectivity of the induced sub graph.

Navigli and lapata [5] presented a method for build a graph, that had few parameters and did not require sense-annotated data for training. First, added the sense of words in a set, then for the all of sense perform a Depth-First Search (DFS) of the WordNet graph. If appear the node is a member of set, will add all the intermediate nodes and edges on the path in the set. Finally, uses the graph connectivity measures for selecting the best sense for each of words. Also present a study of graph connectivity measures for unsupervised WSD and indicated that the local measures performance is better than global measures. The best local measures are Degree and PageRank.

Sinha and Mihalcea [6] extend their previous work on unsupervised graph-based method for word sense disambiguation by using a collection of semantic similarity measures when assigning a weight to the links across synsets. Also presents and performs this system with all the measures of word semantic similarity and graph connectivity measures. Also Showed that the right combination of word similarity metrics and graph centrality algorithms can significantly outperform methods proposed in the past for this problem, therefore reduces 5–8% of error rate.

Agirre and Soroa [11] proposed a new graph-based method that uses lexical knowledge base in order to perform unsupervised word sense disambiguation. They create a sub graph of WordNet which connects the senses of the words in the input text, and then use Personalize PageRank. Performance is better than previous approaches that used PageRank in English all-words datasets. Also show that the algorithm can be easily ported to other languages with good results. The good choice of WordNet versions and appropriate relations are fundamental to the system performance.

3.WordNet

WordNet is an ontology of lexical which created and maintained at Princeton University. The WordNet lexicon contains nouns, verbs, adjectives, and adverbs. Senses of lexical have relation with together. The words that have similar sense encodes in synonym sets (henceforth synsets). Wordnet 3 is the latest version, contains approximately 155,000 words that organized in 117,000 synsets [1,4].

Relations have been organized in two sets, Lexical and semantic relations. The lexical relations are used to connect between the lexical and the semantic relations for the synsets. For example Antonymy, Pertainymy and Nominalization are lexical relations and Hypernymy, Holonymy, Similarity are semantic relations. Also provide a textual definition of the synset possibly with a set of usage examples, that's called gloss. Figure 1, shows the WordNet semantic networks of car_n^1 synset[1].

4. Proposed Method

This section, describes a proposed algorithm. Algorithm proceeds incrementally on a sentence by sentence basis. When given a sentence, is a sequence of words W = $\{w_1, w_2, w_N\}$, assumed the sentences are part-of-speech tagged, so considers content of words only (i.e., nouns, verbs, adjectives, and adverbs).

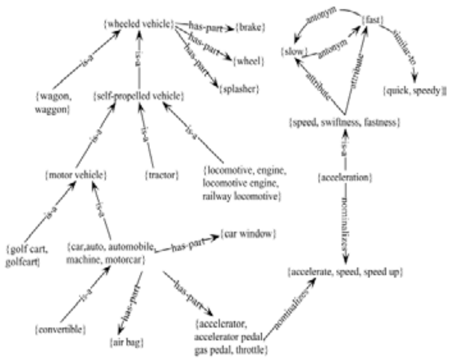

Figure 1.The WordNet semantic networks

Algorithm has two base sections, In First section, to enhance the algorithm performance, omits the stop words[1]. Then, for each of w_i, must extract the senses from WordNet that have specified part-of-speech and Tag-Count is greater than zero, S_{w_i}= $\{S_{w_i}^1, S_{w_i}^2, \ldots, S_{w_i}^n,\}$. Tag-Count is frequency of this word sense measured against a text corpus. After that, add the senses in set G . G uses for graph G= (V , E). V is a set of nodes and E is a set of edges respectively, V = { $S_{w_i}|i$ = *1..N*} and E = Ø.

For each of S_i in G must build a tree**.** This tree builds from the relations of WordNet. To improve performance of the algorithm only use relevance lexicalizes and words. Furthermore, the lexical will be added within the tree when it does not appear in the previous levels of tree. Thus, the nodes of tree are lexical and the edges are lexical relations. The depths of tree is denote with *maxlevel.* The *maxlevel* is a maximum level of the tree.

In second step, search the tree to find a node that is a member of G. If it found, an edge would be added from the root's tree to specific node. Finally, use the connectivity measure method to select the best sense for each of words. Sometimes none of word senses are a member of the graph. For these words select the sense that has the highest probability (the first sense) which is the common sense.

For example, assume we have the sentence "*he drinks some milk*". Initially omit the stop words are (*he, some*). Then, extract the senses of *Drink* and *Milk* from WordNet. *Drink* in this sentence is verb and *Milk* is noun. WordNet for *Drink* has five senses and four senses for *Milk*. But, only four senses for *Drink* and two senses for *Milk* have the Tag-Count greater than zero. Add these senses in set G. Therefore, must build the tree for all of members$G$. Figure 2 shows the tree of $drink_v^1$.

$$G = drink_v^1, \ldots, drink_v^4, milk_n^1, milk_n^2$$

After completing the tree, search in the tree for finding the nodes that are a member of G.In figure 2 shows target nodes denote with green. If target nodes found, the edge would be added in set G**.** Figure 3 shows the finally graph.

Now use the one kind of connectivity measure for select the best sense. Here first sense is better sense for *Drink* and *Milk*. Duo to, have the most arrival connectivity.

Algorithm 1. Propose method For Word Sense Disambiguation.

Input: Sequence W = {w_i |*i* = 1..N}

Extract senses
for each of words **do**
extract the senses that have Tag-count > 0.
add s_i in G.
end for

Build Tree and Graph
for each of s_i in G **do**
While level of tree <= *maxlevel* **do**
for all nodes of tree (v_i) **do**
for all the WordNet lexical relations of v_i **do**
if lexical not exist in the tree **then**
add the lexical in the tree.
end if
end for
end for
end while
If find the nodes are member of G **then**
add edge form s_i to nodes in G.
end if
Delete Tree.
end for

Score vertices in G
for all vertices in G **do**
Score(v) ← Degree Centrality(v).
end for

Sense assignment
for each of words **do**
sense of word ← max (Score(v)).
end for
If words don't have the sense in G **Then**
sense of word ← first sense.
end if

[1] http://www.webconfs.com/stop-words.php

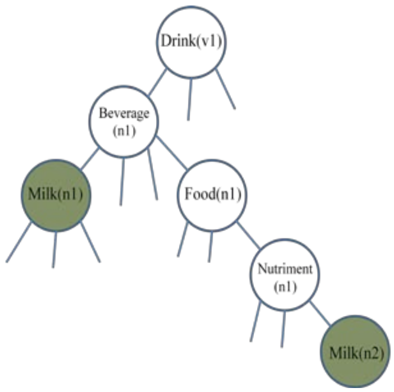

Figure2. Tree of Drink(v1)

5. Experiments And Result

In order to speed up and enhance accuracy the word sense disambiguation, we use the tree structure and prune some relations. Moreover, all paths connecting pairs of senses in WordNet were exhaustively enumerated and stored in a database. Also determine the best maximum value for depth of the tree experimentally. Run WSD algorithm on the Sensaval-3 data set using the Degree connectivity measure and the WordNet sense inventory while varying the depth length from 3 to 6. The length 3 isn't very good. The length 5 and 6 are very time consuming and their accuracy are not better than 4.

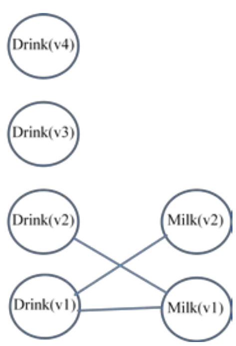

Figure 3.Graph for the sentence he drank some milk (Drink, Milk).

Therefore, we choose 4 for the depth path of the tree.
In order to select the best sense for the words in graph can use local and global measure methods. Local measures of graph connectivity determine the degree of relevance of a single vertex in a graph. But, Global connectivity measures are concerned with the structure of the graph as a whole rather than with individual nodes. Navigli in [5] indicated that local measures yield better performance than global ones, and the degree centrality that is local measure had the best result for the graph. Degree centrality is the simplest way to measure node, it is the degree of node that normalized with maximum degree [7]. This paper used the degree centrality.

5.1. Data

Evaluation and comparing the word sense disambiguation systems is very difficult, because each other use the different data set, knowledge resources and sense inventory. *Senseval*[2] (now renamed *Semeval*) is an international word sense disambiguation competition. The objective is to perform a comparative evaluation of WSD systems in several kinds of tasks, include all-words and lexical sample WSD for different languages. The Senseval workshops are the best reference to study the recent trends of WSD.
This paper evaluated the experiments on the Sensaval-2 [9] and Sensaval-3 [10] English-all words data sets. These data sets were manually annotated with the correct senses by human and use for competitions and evaluation the different systems. The sensaval-3 is difficultly to disambiguate than sensaval-2, but the Senseval-2 data set is meaningfully than the Senseval-3 data set, thus more appropriate as a test set [6]. These data-sets labeled withWordNet1.7 tags. These were normalized to WordNet 3.0 using publicly available sense mappings[3]. Table 1, shows the statistics of those data sets.

Table 1.Occurrences of noun (N), verb (V), adjective (Adj.) and adverb (Adv.) words of Wordnet 3 in Senseval 2 and Senseval 3.

Sensaval-2				Sensaval-3			
N	*V*	*Adj*	*Adv*	*N*	*V*	*Adj*	*Adv*
1136	581	457	299	951	751	364	15

5.2. Results

This section provides an evaluation the tasks that Described in the previous section. The base algorithm, uses the all relation of WordNet and all the words in a sentence (denote AT-A), Also extracts the senses from WordNet that have the Tag-Count are greater than zero. With these conditions in Senaval-3 accuracy and recall are 52.67% and Senaval-2 is 58.67%. The AT-A problem is time consuming, due to using the all words in the sentence. If omit the stop words, in the Sensaval-3 accuracy is 56.52% and recall is 44.16 %, also the Sensaval-2 accuracy and recall are 62.18% and 47.23 % respectively. This method Denotes with WT-A. When omit the stop word, the accuracy and system performance are improved, but reduced the recall.

2 http://www.senseval.org.
3 http://www.cse.unt.edu/~rada/downloads.html

Therefore, the stop words don't have specific sense and may add the noisy edge in the graph.

In order to reduce time and enhance the performance and accuracy, use only lexical relation for building the tree and omit some nodes and extract the senses from WordNet that have the Tag-Count are greater than zero. It's our proposed algorithm (denote WT-R). With this condition in Sensaval-3 accuracy is 63.28% and recall is 49.45% and in Sensaval-2 accuracy and recall are 65.00% and 49.41% respectively. This has very good time and accuracy, because use only lexical relation and prune some nodes and senses.

Table 2 compares the accuracy of the best graph-based method with our methods. As discussed in Sec 2. Mihalcea et al. [13] (Mih05), the method of Agirre and Soroa [11] (Agi09), the results from the work of Navigli and Lapata [7] (Nav07), the method of Navigli and Velardi [4] (SSI), the method of Navigli and Lapata [5] (Nav10) and the method of Sinha and Mihalcea [6] (Sinha07) are well-known methods in the literature.

Table 2. Comparison with related work

	Sensaval-2	Sensaval-3
	Accuracy	*Accuracy*
Mih05	54.2	52.2
Agi09	59.5	57.4
Nav07	n/a	52.5
SSI	n/a	60.4
Nav10	n/a	52.9
Sinha07	56.4	52.4
AT-A	58.67	52.67
WT-A	62.18	56.52
WT-R	**65**	**63.28**
FS	63.7	61.3

Whenever results were not available, due to they were not reported in the literature, an entry *n/a* exists in the respective cell. Finally, added in the comparison a simple heuristic method (*FS*) that always selects the first sense of the target word from WordNet (i.e., the most frequent) to conduct the disambiguation.

Figure 4, Show Elapsed time (in minutes) of our algorithms when applied to the Senseval-3 dataset. The proposed method has very good time, use the some relations of WordNet. Also, with omit stop word the performance of system is better than when use the all words. These times acquired by a computer with processor 2.50GHZ Core 2 Duo and 4GB RAM.

6. Conclusion And Future Work

This paper has proposed a new method for word sense disambiguation. First builds a tree for some of the senses of ambiguity words which there are in the sentence and detects the best path. Then with these paths builds a Graph and uses the connectivity measure for choosing the best sense of words. Here, we used the degree centrality, because Navigli [5] proved it's the best connectivity measure. When we are building the tree, uses some relations of WordNet to improve the accuracy and performance system together. The previous methods used the all relation and senses of word for WSD. But, we use only lexical relation and the senses that have the Tag-Count are greater than zero. With this condition our

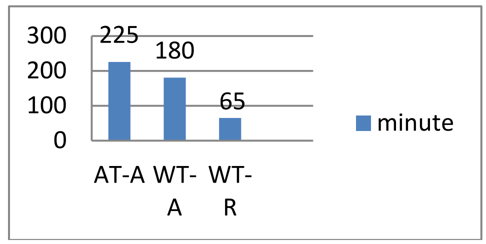

Figure 4. Elapsed time (in minutes) of the algorithm when applied to the Senseval-3 dataset

graph is less than other methods and compute is easier. Also the result is better than other graph-base and other unsupervised method. Performance our proposed method (WT-R) is greater than 20 percent better than base method (AT-A) and accuracy is 63.28 percent in sensaval-3 and 65.00 percent in sensaval-2 dataset. The algorithm can be applied easily to sense inventories and knowledge bases different from WordNet.

In the future, we are interested in applying the proposed method to weight graphs. For this purpose we can use the measures of word semantic similarity or Navigli proposed graph [6] with other conditions and calculate the probability of nodes in graph connectivity.

Refrences

[1] R.Navigli," Word sense disambiguation: A survey", *ACM Comput. Surv*, 41, 2, Article 10 ,February 2009.

[2] G.Tsatsaronis, I.Varlamis, K.Nørvag," An Experimental Study on Unsupervised Graph-based Word Sense Disambiguation", *CICLing - Conference on Intelligent Text Processing and Computational Linguistics*2010.

[3] M. Lesk." Automated sense disambiguation using machine-readable dictionaries: How to tell a pine cone from an ice cream cone". In *Proc. of the SIGDOC Conference*, pages 24–26,1986.

[4] R.Navigli, P.Velardi. "Structural semantic interconnections: A knowledge-based approach toword sense disambiguation," *IEEE Trans. Pattern Anal. Mach. Intell., 27(7):1075–1086*, 2005.

[5] R.Navigli, M.Lapata, ," An Experimental Study of Graph Connectivity for Unsupervised Word Sense Disambiguation," *Pattern Analysis and Machine Intelligence*, IEEE Transactions on, IEEE Computer Society, April 2010.

[6] R.Sinha, R.Mihalcea," Unsupervised Graph-based Word Sense Disambiguation Using Measures of Word Semantic Similarity," In *IEEE International Conference on Semantic Computing,* ICSC 2007.

[7] R.Navigli, M.Lapata." Graph connectivity measures for unsupervised word sense disambiguation," *IJCAI'07 Proceedings of the 20th international joint conference on Artifical intelligence*,2007.

[8] S.Banerjee, T.Pedersen," An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet ," *CICLing '02 Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*,2002.

[9] M. Palmer, C. Fellbaum, S. Cotton, L. Delfs, and H. Dang. "English tasks: all-words and verb lexical sample". In *Proceedings of ACL/SIGLEX Senseval-2,* Toulouse, France,2001.

[10] B. Snyder and M. Palmer. "The English all-words task". In *Proceedings of ACL/SIGLEX Senseval-3*, Barcelona, Spain,July 2004.

[11] E. Agirre and A. Soroa. "Personalizing pagerank for word sense disambiguation". In *Proc. Of EACL*, pages 33–41, 2009.

[12] Vasilescu, Florentina, P.Langlais, G.Lapalme," Evaluating variants of the Lesk approach for disambiguating words", *Proceedings of the Conference on Language Resources and Evaluation (LREC),* Lisbon, Portugal,633–636. 2004

[13] R. Mihalcea. "Large vocabulary unsupervised word sense disambiguation with graph-based algorithms for sequence data labeling". In *Proceedings of the Human Language Technology / Empirical Methods in Natural Language Processing conference*, Vancouver, 2005.

[14] R. Mihalcea, P. Tarau, and E. Figa. "Pagerank on semantic networks with application to word sense disambiguation". In *Proc. of COLING*, 2004.

[15] R. Navigli. "Online word sense disambiguation with structural semantic interconnections". In *Proc. of EACL,* 2006.

[16] G. Tsatsaronis, M. Vazirgiannis, and I. Androutsopoulos. "Word sense disambiguation with spreading activation networks generated from thesauri". In *Proc. of IJCAI*, pages 1725–1730,2007.