CS2309 Survey Paper
Word Sense Disambiguation

Heng Low Wee
U096901R

1 Introduction

Human languages are often ambiguous. For instance, in English, there are many words that possess multiple meanings depending on the context. These meanings are referred to as *senses*. Let's look this two examples:

1. She is interested in the *interest* rates of the bank.
2. He developed an *interest* in art.

We can observe that the word *interest* in the two sentences clearly have different meanings. It is easy enough for us, humans, to tell the meanings of the word in each sentence. This is because humans have that "knowledge" to tell the difference. In the first sentence, we saw words like *bank* and *rates* and we were able to relate accordingly. This thought process is just like how Weaver [8] said, that if we only examine some text one word at a time, it is impossible to determine the meaning of it. To decide its meaning we must also look at its surrounding words. It is almost an unconscious act for a human to interpret the senses of words in a sentence. Machines, on the other hand, are not conscious, and don't have that "knowledge". Therefore, machines need to run through a series of analysis before they can tell the senses of the ambiguous word. This process is called *Word Sense Disambiguation* (WSD).

Word Sense Disambiguation is one of the basic tasks in Natural Language Processing. Other tasks of NLP would include Word Segmentation, especially for languages that do not use spaces as a delimiter, for example Chinese. WSD has many applications in computational linguistics, like Machine Learning, Text Mining and Information Retrieval. It can also be used by search engines like Google, that being able to identify the senses, the search results would probably be more relevant to what the user had in mind.

1.1 Types of Word Sense Disambiguation

There are four conventional types of WSD, they are:

1. Dictionary-based & knowledge-based
   This method uses formal dictionaries and lexical knowledge bases to disambiguate senses. These dictionaries provide the definitions of the possible senses of an ambiguous word, and then the definitions could be used in WSD algorithms. One example would be the Lesk Algorithm [1]. The general assumption of Lesk algorithm is that words in a given environment (sentence, paragraph etc) will tend to share a common topic. Santanjeev & Ted, instead of using standard dictionaries, adapted the Lesk Algorithm to use WordNet [3] as a source of word senses. They conducted testings, and from their results they found that the adapted implementation outperformed the traditional Lesk approach with an accuracy double of the traditional one. Not only that they achieved better results with their algorithm, they also demonstrated that by integrating with a lexical database like WordNet, it would increase the rate of accuracy of WSD.

2. Supervised
   Supervised WSD uses manually sense-tagged corpora. By doing so, these methods achieve a higher accuracy as compared to most unsupervised methods. However, supervised methods are subject to *knowledge acquisition bottleneck* since they rely on manually sense-tagged corpora for training. Furthermore, these corpora are both labour-intensive and costly to create.

3. Semi-supervised
   This method makes use of both labeled and unlabeled data. Similarly, it refers to using multiple untagged corpora to provide concurrent information to supplement a tagged corpus.

4. Unsupervised
   Most challenging approach among all, this method may also be related to *Word Sense Induction*, where senses could be induced by analyzing words in a given text. Unsupervised methods perform WSD without using any resource. To determine the senses, algorithms map the words to a collection of senses (e.g. WordNet). Mihalcea [6] introduced an approach that uses the articles on Wikipedia, together with mapping of senses with a WordNet resource. It addressed the *knowledge acquisition bottleneck* problem, as Wikipedia is updated by users all around the world on a regular basis. Also, it addresses the case where languages are evolving with new words and senses surfacing.

1.2 Motivation

I picked this topic mainly because of its benefits when applied into bilingual translation, for instance English-Chinese. Let's look at the example sentence, *He developed an interest in art.* In this context, it is more related to the word "hobby". So when we translate the word *interest* from English to Chinese, we want the output to be 兴趣 (human interest), instead of 利息 (simple interest). Being able to translate while retaining the original context would mean greater accuracy and relevancy in the translated text. Ultimately, this would encourage language learning as it is simpler and more straightforward when a language learner is aware of which is the correct sense to be used in a given context.

On the other hand, we are all aware that it is not uncommon to find new words or senses appearing on a regular basis. These words are not some made-up words, they are words that do, eventually, end up in Oxford Dictionary. For instance, some time ago, the word *tweet* refers to the chirp of a young bird. Today, Twitter has given *tweet* a brand new meaning. That is, a *tweet* is a post on the social networking service Twitter. This goes to show that there is a need for WSD systems to be able to keep up to new words and senses.

1.3 Outline

In this survey, specifically, we will look at Unsupervised WSD. We will also look at a disambiguation process's flexibility to handle new words and senses, as it is very common for new words and senses to be surfacing occasionally. Section 2 will cover three works related to WSD, and for each work some comments about the pros and cons. We will then conclude in Section 3.

2 Literature

2.1 Word Sense Disambiguation using Dependency Knowledge

This approach described by Chen et al [2] begins with the construction of a corpus. In the paper considered two possible text sources for corpus building. The two options are Electronic Text Collection and Web documents. The first option, even though professionally created and accurate, is old and lack of new words and senses used in modern English. Hence the authors picked the second option, even though the major concern is with data inconsistency. The procedure briefly, they sent ambiguous words to Web search engines to retrieve the relevant pages. These pages are cleaned, segmented and then parsed with a *dependency parser*, Minipar [5], to retrieve the parsing trees. These

trees are merged to form the *context knowledge base* [2]. Formulated into *weighted directed graphs*, it is effective in telling the dependencies between the words in a given text, simply by computing the weights of the nodes in the graphs. Chen et al introduced the *TreeMatching* function, which handles the weight assignments and score computations. A target sentence is passed into the WSD algorithm together with WordNet sense inventory and the context knowledge base built earlier. *TreeMatching* then assigns weights to the nodes based on rules and dependency relation instances, and returns the score of a WordNet gloss that an ambiguous word was compared to. Subsequently, either the sense with the best score, or the first sense will be decided as the correct sense.

The algorithm is effective, and accurate in matching the dependency relations to determine the correct senses, as shown in the evaluation results [2]. Even though it is an unsupervised method, its performance was approaching some supervised ones. However, before *TreeMatching* can be performed, all the glosses and sentences have to be preprocessed into parsing trees, which takes a lot of time according to Chen et al. According to [5], Minipar was able to achieve 89% precision for parsing sentences. One "missing piece" of the paper is, Chen et al mentioned that impact of the erroneous relations from the Minipar's parsing would be minimized with their WSD algorithm, but it was not explicitly defined how it actually did it.

2.2 Words Sense Disambiguation using Wikipedia

Manually created by users, Wikipedia articles are generally correct in terms of its contents. Wikipedia links, in the *MediaWiki* syntax, on an article provide navigation to related pages. Some examples of these links are [[**bar(law)**|**bar**]] and [[**bar(counter)**|**bar**]]. The senses of the word *bar*, namely *law* and *counter*, can be derived by extracting them from the annotated links. Mihalcea [6] made use of this information and introduced a method to build a sense-tagged corpora using the occurrences of a given words. Mihalcea's approach begins with first extracting all paragraphs from Wikipedia that contain the occurrences of a given word. Then, the senses of each word are extracted, then mapped onto their corresponding WordNet senses. Then, in the disambiguation algorithm [6], a target text is tokenized, and each token is tagged with its POS information. Collocations are also identified. Then, local and topical features are extracted form the context of the ambiguous word. This set of features is similar to the one used by Ng & Lee [7].

This method would address the *knowledge acquisition bottleneck* issue. This is because the dynamic nature of Wikipedia makes it extensible. Hence, using Wikipedia as a corpus source can ensure that it will be up-to-date and contain any new words or new senses. However, there might be data inconsistency. Various users might use different names for

the same object or entity. For instance, *handphone* and *mobile phone*. This is also partly because of the fact that there are new words and senses emerging rapidly in today's modern languages.

One can observe that this approach would have the flexibility to handle new words and senses. For example, considering how popular Wikipedia is, any new word used by people, say *tweet* (a post on Twitter), would most likely appear on Wikipedia much faster than any other formal dictionaries or lexical databases. We must, however, consider the fact that as the number of Wikipedia pages grow, it is necessary to re-construct the sense-tagged corpus in order to capture these new words and senses. Hence the design considerations should include *expandability*.

2.3 Word Sense Disambiguation using the Noisy Channel Model

In this paper, the authors noted that the term "unsupervised" had been slightly misused. Unsupervised systems are supposed to be systems that do not directly use sense-tagged corpora for training. However, many unsupervised systems do use sense ordering and sense frequencies from the lexical database WordNet. According to [9], these systems should be classified as weakly-supervised or semi-supervised instead.

This approach for unsupervised WSD was based on the Noisy Channel Model. This model is a framework commonly used in spell checkers, machine translation and speech recognition. It can be used whenever a signal received does not uniquely identify the message being sent. Bayes' Law is used to identify the most probable intended message within the channel. The authors modeled each context, C, as a distinct channel. Then the senses, S, are modeled as the intended messages in the channel. The words, W, received would be the signal that is received. With that, they adapted the Bayes' formula, as follows:

$$P(S|W,C) = \frac{P(W|S,C)P(S|C)}{P(W|C)} \tag{1}$$

So essentially, to maximize the value of $P(S|W,C)$, $P(W|S,C)$ and $P(S|C)$ must be increased. To estimate the values of $P(W|S,C)$ and $P(S|C)$, they assumed that the distribution of words used to express a given sense is the same for all context, such that $P(W|S,C) = P(W|S)$. They used the WordNet sense frequencies to estimate $P(W|S)$, and a statistical language model, 5-gram model, to estimate $P(W|C)$.

In their experiments the authors compared their system to some of the best supervised and unsupervised systems. It was found that this probabilistic approach of theirs was able to outperform all the unsupervised systems that were compared with. Recall that the authors had noted that some of all the unsupervised systems available should actually

be considered as "semi-supervised" because of their reliance on sense-tagged corpora for training. In other words, their unsupervised WSD had outperformed some of the "semi-supervised" ones, which should be considered a remarkable feat. Also, the main contribution of this method is the reduction of the WSD problem into the estimation of two distributions: the distribution of words that can be used in a given sense $[P(W|S)]$ and in a given context $[P(W|C)]$.

3 Conclusion

In this survey we touched on the field of *Word Sense Disambiguation* (*WSD*). WSD is a very challenging task because it involves working with the complexity of human languages. First formulated as a distinct computational task during the 1940s, WSD is one of the oldest problems in computational linguistics. Weaver [8] wrote in his memorandum, that if we examine text *one* word at a time, it is impossible to determine the meaning of it. Only by looking at its surrounding words then can one decide its meaning. Related to our interest to improve bilingual translations by WSD, it is notable that Lefever & Hoste [4] had demonstrated Cross Lingual WSD using parallel corpora from Europarl[1]. However, the corpora used were mainly in European languages, not applicable when the languages we are focusing on are English and Chinese.

In general, the characteristics mentioned in this survey are *accuracy*, *flexible* (to handle new words & senses) and *unsupervised*. For that, we have studied some approaches and techniques that may give rise to these characteristics in WSD. Chen et al [2] demonstrated remarkable *accuracy* in unsupervised WSD by using dependency knowledge. Despite being unsupervised, their method was competitive to the supervised methods, with performance approaching the supervised ones. To be *flexible*, the general idea is to adapt a Web-based resource for sense-extraction. Mihalcea described using Wikipedia for WSD [6]. Advantages include it being publicly accurate in its contents, and it being exposed to new words and senses added by articles' authors regularly. While data inconsistency might be an issue here, it is flexible enough to handle ambiguous words that may not yet appear on formal dictionaries. As for *unsupervised*, Yuret et al [9] noted that the term "unsupervised" was probably misused. "Unsupervised" was meant to refer to system that do not directly rely on sense-tagged corpora for training. However, most of the unsupervised systems available do so. They introduced a WSD method that was based on the Noisy Channel Model. In the end, they were able to reduce the WSD problem into a probabilistic one, performing estimations of two distributions.

[1] http://www.statmt.org/europarl/

Perhaps the next idea to consider is to go real-time for WSD. So far the above mentioned literatures did not touch on this idea, probably because of performance issues, for it can be easily inferred that the construction of large corpora could not possibly be achieved in a matter of seconds. While this poses yet another challenge not exclusive to the field of Word Sense Disambiguation, but as systems' performance are reaching new heights regularly, it should not be impossible in the near future.

References

[1] Banerjee, S., and Pedersen, T. An adapted lesk algorithm for word sense disambiguation using wordnet. In *CICLing '02: Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing* (London, UK, 2002), Springer-Verlag, pp. 136–145.

[2] Chen, P., Ding, W., Bowes, C., and Brown, D. A fully unsupervised word sense disambiguation method using dependency knowledge. *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics on - NAACL '09*, June (2009), 28.

[3] Fellbaum, C., Ed. *WordNet An Electronic Lexical Database*. The MIT Press, Cambridge, MA ; London, May 1998.

[4] Lefever, E., and Hoste, V. Semeval-2010 task 3: cross-lingual word sense disambiguation. In *DEW '09: Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions* (Morristown, NJ, USA, 2009), Association for Computational Linguistics, pp. 82–87.

[5] Lin, D. A dependency-based method for evaluating broad-coverage parsers. *Nat. Lang. Eng. 4*, 2 (1998), 97–114.

[6] Mihalcea, R. Using wikipedia for automatic word sense disambiguation.

[7] Ng, H. T., and Lee, H. B. Integrating multiple knowledge sources to disambiguate word sense: an exemplar-based approach. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics* (Morristown, NJ, USA, 1996), Association for Computational Linguistics, pp. 40–47.

[8] Weaver, W. Translation. In *Mimeographed* (1949), MIT Press, pp. 15–23.

[9] Yuret, D., and Yatbaz, M. A. The Noisy Channel Model for Unsupervised Word Sense Disambiguation. *Computational Linguistics 36*, 1 (Mar. 2010), 111–127.