UROP I Progress Report
Word Sense Disambiguation

Heng Low Wee
U096901R

1 Introduction

In any human language, there are some words, or even phrases, that have multiple interpretations depending on the context in which a word is used. These words are called homographs, and have multiple meanings (or *senses*), making language itself ambiguous. Here are some examples:

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

Observe that the word *interest* in the two sentences clearly have different meanings. This is the ambiguity we are talking about. Paraphrasing Weaver [10], if we examine some text only *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. It is almost an unconscious act for a human to interpret the word's correct sense. But because human languages are complex, a machine need to run through a series of analytical processes before it can guess the best answer. In general, this process is *Word Sense Disambiguation* (*WSD*). More formally, WSD is the process of deciphering the intended meaning of a homographic word(s) in a given context. There are four conventional methods to WSD, namely:

1. Dictionary-based & knowledge-based
   This method relies mainly on dictionaries and lexical knowledge bases to disambiguate senses. The Lesk Algorithm [5] is such a method. Satanjeev & Ted adapted the Lesk Algorithm but instead of using a standard dictionary, they used WordNet [4] as a source of words' senses (for details, refer to [1]). After evaluation testings, it was found that their implementation outperformed a more traditional Lesk approach with an accuracy rate of 32%, double of the traditional approach. This demonstrated the fact that an approach that integrates WordNet, or other lexical databases, would be able to achieve a higher rate of accuracy in WSD.
2. Supervised
   Generally, this method relies on manually sense-tagged corpora. Using manually sense-tagged corpora allows supervised methods to achieve a higher accuracy as compared to most unsupervised methods. The downside of using manually sense-tagged corpora, however, is its high costs and labour intensiveness 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
   This is probably the most challenging among all the approaches to WSD. This method may also be closely related to *Word Sense Induction*, where senses can be induced from analyzing the words in a given text. Schutze [9] described a disambiguation algorithm based on clustering words, and then senses are interpreted as a cluster of similar context of an ambiguous word. To determine the senses, some algorithms may also map the words to a collection of senses. Mihalcea [7] described an unsupervised approach by using the articles on Wikipedia, together with mapping of senses with a WordNet resource. It addressed the issue of *knowledge acquisition bottleneck*, as Wikipedia serves as a dynamic source for senses because it is updated by users all around the world regularly. More importantly, it addresses the case where languages are gaining new words, and that words are gaining new senses, a common thing in today's modern society.

There are many applications that are related to WSD. For instance, if search engines are able to identify the correct sense in the search words, search results will be more accurate and relevant. Another application would be language translations. During the translation process from English to Chinese and vice versa, words that are ambiguous would usually end up with an entirely wrong translation output. For instance, the translation for *interest*, as mentioned in the first example, might be 兴趣, which is more related to the term "hobby". Also, we wish for the system to be able to handle new words and new senses in human languages. In this paper, we introduce an existing translation tool, DiCE Translator, that allows users to translate text on a webpage from English to Chinese and vice versa. We also explore some methods to integrate WSD so as to improve the accuracy of bilingual translations by retaining the original context.

2 Related Work

2.1 Word Sense Tagging using Parallel Corpora

Supervised word sense disambiguation systems rely heavily on manually sense-tagged corpora, and to produce more reliable results they need high quality annotations. However, manual tagging is very labour intensive and costly. It is also very impractical as doubling the training corpora only reduces errors by 3 to 4% [11].

Diab & Resnik described a form of automated annotation and sense-tagging by analyzing an ambiguous word's translations in a second language [3]. Their approach involves two languages, for example English and French. The first step is to identify the target words and their corresponding translations in the source corpus. Word alignment is carried out and the corresponding positions of a target word in both languages are captured. Then, grouping the target words into *target sets*. Next, in each set consider all possible senses for each word and then tag a word with the sense most similar to the other words.

Last, they make use of the sense tags in the target sets and project them to the source corpus. (for details, refer to [3])

2.2 Using Wikipedia for Automatic Word Sense Disambiguation

Wikipedia articles, manually created by users, are generally correct in terms of its contents. Some examples of Wikipedia links (on an article), in the *MediaWiki* syntax, 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 [7] made use of this information and described an approach to build a sense-tagged corpora using Wikipedia articles. It begins with extracting all paragraphs from Wikipedia that contain the occurrences of a given word. It follows that the senses of each word are extracted, then mapped onto their corresponding WordNet senses. Then, the approach moves on into the disambiguation algorithm [7]. A target text is tokenized, and each token is tagged with its part-of-speech information. Then, collocations are identified. Next, local and topical features are extracted from the context of the ambiguous word. This set of features is similar to the one used by Ng & Lee [8]. (for details, refer to [7])

2.3 Word Sense Disambiguation using Dependency Knowledge

Similar to Section 2.2, this approach described by [2] begins with the construction of the corpus. In short, ambiguous words are sent to Web search engines to retrieve the relevant pages. These pages are cleaned, segmented, and then parsed with a *dependency parser*, Minipar [6] to retrieve the parsing trees, which 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 values of the weighted nodes. The weight assignments and score computations are handled by the *TreeMatching* function [2], which is the score calculator for the weights in the parsing trees. 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 with. Subsequently, either the sense with the best score or the first sense will be determined as the correct sense. (for details, refer to [2])

3 Method

We intend to focus on, and implement two methods of WSD, namely the *It Makes Sense* [12] and Mihalcea's method as mentioned in Section 2.2 [7], so as to achieve better translations. But before we discuss the two methods, we shall briefly describe DiCE, and provide some updates of it.

DiCE Translator[1] is a Firefox extension that allows users to translate text on a webpage from English to Chinese and vice versa. Currently it supports only the English and Chinese languages. When triggered, a tooltip (see Figure 1) that contains various information is displayed. These information include pronunciation, translation, and meanings. Previous work done on this extension include generating database tables that maps one language to another, and also the pronunciation for the source language. Other than these database tables, DiCE also uses Google Translate API to translate multiple words, though with limited number of words due to limits in the length of the query string.

In order to make learning languages easier, it was necessary to include other information like a word's definition, and possibly some example sentences so a user could know the word's usage. Wiktionary was opted to be the source for such information. Briefly, the extraction process is as follow:

1. Wiktionary XML dumps are first downloaded from the site. We downloaded both English and Chinese dumps.
2. We parsed the dumps to extract the words, pronunciations, senses and sample sentences
3. The extracted information are inserted into our database for our tool to access. For each word, there can be multiple part-of-speech (POS). For each POS, there can be multiple senses. The database schema, in general, is illustrated in Figure 2.

Figure 1: DiCE Translator tooltip

From the extraction process, we extracted more than 47,000 English words and more than 3,000 Chinese words. Yet, it was all only about retrieving dictionary information, that from these information, we are unable to differentiate the senses so that they match the context in the sentence on a webpage. In other words, like the example mentioned earlier, we do not want *interest* in "interest rate" to be interpreted as 兴趣.

What we want to do for DiCE is to make it "smarter" by being able to translate relevantly in terms of senses and context. For that, we intend to implement the following two methods of WSD.

[1]https://addons.mozilla.org/en-US/firefox/addon/12443/

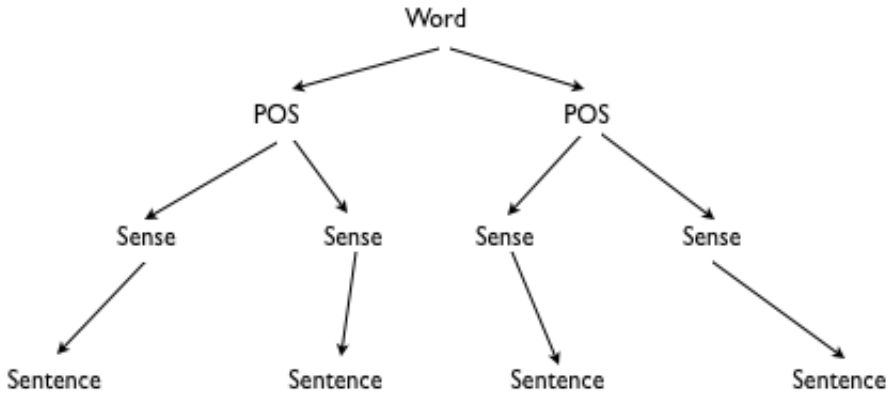

Figure 2: Database Schema

3.1 Supervised English All-words Word Sense Disambiguation

It Makes Sense (IMS) is a supervised English all-words WSD System introduced by Zhong & Ng [12]. It is a flexible system that allows users to customize it by integrating different preprocessing tools and additional features. There are generally 3 modules in IMS: Preprocessing, Feature & Instance Extraction and Classification. The system accepts any input text. In the Preprocessing module, the OpenNLP toolkit is used by default. Sentences in the input are detected and split using a sentence splitter, and then tokenized into words, before POS tags are assigned to all tokens using OpenNLP's POS tagger. Then, the lemma form of each token is determined using a lemmatizer that is based on using the WordNet thesaurus. In the Extraction module, it makes use of 3 knowledge sources namely, POS Tags of Surrounding Words, Surrounding Words and Local Collocations. In general, the purpose of using these knowledges is to increase the accuracy of the the WSD. Finally in the Classification module, the IMS's classifier trains a model for each word type which has training data during the training process. The classifier as used by IMS is LIBLINEAR. In the testing process, the trained classification models will be applied to the test instances of the corresponding word types. If a test instance is not seen, IMS outputs the first sense found in the WordNet sense inventory.

We intend to utilize IMS as a back-end system for DiCE to disambiguate words selected by users on a webpage. The advantage of IMS is being able to handle longer text inputs, in sentences or paragraphs. However, we are limited to only English inputs, and might be required to use the database tables and Google Translate concurrently to translate to Chinese language.

3.2 Word Sense Disambiguation using Wikipedia

We make reference to Mihalcea's work [7] on using Wikipedia for WSD. As mentioned in Section 2.2, we can utilize the annotations created by authors of Wikipedia to determine the sense of ambiguous words. Most importantly, for a word like *interest*, a Wikipedia page would contain translations in various context. This will address the limitation highlighted in Section 3.1, that we can have more accurate English-to-Chinese translations

that preserve context.

4 Conclusion

Word Sense Disambiguation is a very challenging task due the complexity of the human languages. There are various approaches, either by using formal dictionarys and lexical databases, or to induce the word senses by analyzing the words in a given text. One of the applications for WSD is making translations more accurate and relevant by retaining the original context, which is related to the goals of the DiCE Translator, a Firefox extension that allows users to translate text on a webpage from English to Chinese and vice versa. With DiCE, we hope to make language learning easier by providing translations for the language to be learnt. Other than providing definitions and sample sentence, we wish to make translations more meaningful and accurate by being able to retain the context of the text selected by users. For that we looked at a few WSD-related works. Notably, the approach introduced by Mihalcea [7], and the *It Makes Sense* [12] system have been considered to be integrated with the DiCE Translator. No formal evaluation has been performed and so what follows would be the implementation of these WSD systems, in order for us to move on into the Evaluation phase.

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] Diab, M., and Resnik, P. An unsupervised method for word sense tagging using parallel corpora. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (Morristown, NJ, USA, 2002), Association for Computational Linguistics, pp. 255–262.

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

[5] Lesk, M. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *SIGDOC '86: Proceedings of the 5th annual international conference on Systems documentation* (New York, NY, USA, 1986), ACM, pp. 24–26.

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

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

[8] 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.

[9] Schütze, H. Automatic word sense discrimination. *Comput. Linguist. 24*, 1 (1998), 97–123.

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

[11] Yarowsky, D., and Florian, R. Evaluating sense disambiguation across diverse parameter spaces. *Nat. Lang. Eng. 8*, 4 (2002), 293–310.

[12] Zhong, Z., and Ng, H. T. It makes sense: A wide-coverage word sense disambiguation system for free text. In *Proceedings of the ACL 2010 System Demonstrations* (Morristown, NJ, USA, 2010), ACL '10, Association for Computational Linguistics, pp. 78–83.