Implementing Machine Learning Techniques for Word Sense Disambiguation

Khooshi Asmi, 2022114006

December 3, 2023

Abstract

Word sense disambiguation (WSD) is a fundamental task in natural language processing aimed at determining the correct meaning of a word within a given context. This paper explores the application of various machine learning algorithms, including Naive Bayes, K-Nearest Neighbors (KNN), and Maximum Entropy (MaxEnt), for WSD using the SemCor and OMSTI datasets. Additionally, it investigates the impact of different features such as part-of-speech (POS), bag-of-words (BoW), and word embeddings on the accuracy of the disambiguation process.

1 Introduction

Word Sense Disambiguation (WSD) is pivotal in natural language processing, resolving the ambiguity present in words with multiple meanings within diverse contexts. Its primary goal is to accurately decipher the intended meaning of words, significantly enhancing language understanding and precision in various applications. By determining the appropriate sense of ambiguous words, WSD ensures more accurate translations, improves search engine results by better understanding user queries, and enhances sentiment analysis by contextualizing meanings within sentences. Approaches to WSD include supervised learning using labeled data, unsupervised methods relying on statistical techniques, and knowledge-based approaches utilizing semantic networks and dictionaries. Machine learning techniques, such as Naive Bayes, Support Vector Machines, and neural networks, are extensively employed to disambiguate word senses, leveraging linguistic features like part-of-speech tags and contextual information. Overall, WSD's role in NLP remains critical, with machine learning playing a key role in advancing language understanding and improving various text-centric applications.

2 Literature Review

Machine learning and word sense disambiguation in the biomedical domain: design and evaluation issues [2]:The study by Bakx, Gerard Escudero, L. M. Villodre, and G. R. Claramunt in their unpublished doctoral dissertation (2006) serves as a significant precursor in exploring the application of machine learning techniques for word sense disambiguation. Although the specific methodologies and findings from this dissertation may not be publicly accessible, the work contributes to the broader understanding of employing various machine learning algorithms, notably Naive Bayes, K-Nearest Neighbors (KNN), and Maximum Entropy (MaxEnt), within the context of word sense disambiguation.

2.1 Objectives

The primary objectives of this research encompass: 1. To assess the performance of Naive Bayes, KNN, and MaxEnt algorithms for WSD labeled datasets. 2. To analyze the influence of different features like POS tagging, BoW, POS of the neighbor word representations on the accuracy of word sense disambiguation.

3 Methodology

The methodology adopted in this research involves the following components:

3.1 Data Collection

The study utilizes the SemCor and OMSTI datasets for training and evaluation purposes.

3.1.1 Training data:

SemCor (Miller et al., 1994). SemCor is a manually sense-annotated corpus divided in 352 documents for a total of 226,040 sense annotations. SemCor is, to our knowledge, the largest corpus manually annotated with WordNet senses, and is the main corpus used in the literature to train supervised WSD systems (Agirre et al., 2010b; Zhong and Ng, 2010).

OMSTI (Taghipour and Ng, 2015a). OMSTI (One Million Sense-Tagged Instances) is a large corpus annotated with senses from the WordNet 3.0 inventory. It was automatically constructed by using an alignment-based WSD approach (Chan and Ng, 2005) on a large English-Chinese parallel corpus (Eisele and Chen, 2010, MultiUN corpus). OMSTI has already shown its potential as training corpus by improving the performance of supervised systems which add OM-STI as part of their training (Taghipour and Ng, 2015a; Iacobacci et al., 2016).

3.1.2 Testing Data:

Senseval-2 (Edmonds and Cotton, 2001). This dataset consists of 2283 sense annotations, including nouns, verbs, adverbs and adjectives. Senseval-3 task 1 (Snyder and Palmer, 2004). This datasets is divided in three documents from three different domains (editorial, news story and fiction), totaling 1850 sense annotations. SemEval-07 task 17 (Pradhan et al., 2007). This is the smallest among the five datasets, containing 455 sense annotations for nouns and verbs only. SemEval-13 task 12 (Navigli et al., 2013). This dataset includes thirteen documents from various domains. In this case the original sense inventory was WordNet 3.0, which is the same that we use for all datasets. The number of sense annotations is 1644, although only nouns are considered. SemEval-15 task 13 (Moro and Navigli, 2015). This is the most recent WSD dataset available to date. It consists of 1022 sense annotations in four documents coming from three heterogeneous domains: biomedical, mathematics/computing and social issues. The concatenations of all five above datasets as a single evaluation dataset (”ALL”) was used for evaluation. [1] Testing and training Data

3.2 Feature Engineering

The feature extraction process involved the utilization and combination of various linguistic elements from the provided dataset. The following features were extracted:

- **ID:** Unique identifiers assigned to each instance in the dataset.
- **Lemma:** Lemmatized forms of words capturing their base or dictionary form.
- **Sentence:** Contextual information denoting the sentence where the word appears.
- **POS:** Part-of-Speech tags indicating the grammatical category of the word.
- **POS Sent:** POS tags for the neighboring words in the sentence.
- **BoW:** Bag-of-Words representation capturing word frequency in a given context.
- **Word Form:** The original form or surface representation of the word.
- **POS Before:** Part-of-Speech tags of the word before in the sentence.
- **POS After:** Part-of-Speech tags of the word after in the sentence.

4 Results

The outcomes of the experiments conducted to evaluate different algorithms and features for word sense disambiguation are presented in this section. The baseline model employed a simple strategy of predicting the most frequent words in the dataset. This rudimentary approach yielded an accuracy of 46% for semcor+omsti data and 49% for semcor dataset

4.1 Algorithm Analysis

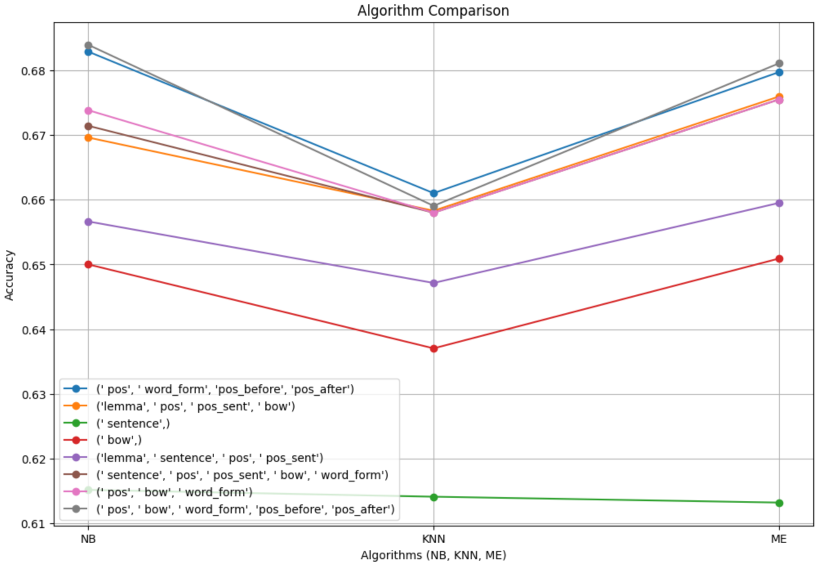
The performance of three prominent algorithms—Naive Bayes (NB), K-Nearest Neighbors (KNN), and Maximum Entropy (ME)—was evaluated across different feature representations for word sense disambiguation.

- **Naive Bayes (NB):**
  - Achieved accuracies ranged between approximately 61.5% to 68.4% across various feature combinations.
  - Notably consistent performance across different features, with a slight variation of about 6.9%.
- **K-Nearest Neighbors (KNN):**
  - Showed accuracies ranging from approximately 61.4% to 66.0% across different feature representations.
  - Slightly lesser accurate than NB, with a variation of approximately 4.6%.
- **Maximum Entropy (ME):**
  - Demonstrated accuracies ranging from around 65.9% to 68.5% across diverse feature sets.
  - Relatively more consistent and accurate performance with a variation of about 2.6%

4.2 Feature Analysis

The impact of different features and their combinations on word sense disambiguation accuracy was analyzed. Features extracted were The following combinations had the best accuracies.

- **POS, Word Form, POS Before, POS After:**
  - NB: 68.2%, KNN: 66.1%, MaxEnt: 67.9%
  - This feature set resulted in competitive accuracies for all algorithms.
- **Sentence, POS, POS Sent, BoW, Word Form:**



- NB: 68.4%, KNN: 65.9%, MaxEnt: 68.1%
- This combination of features gave the highest accuracies on using NB and MaxEnt Classifier

Through comparative analysis of all the accuracies of all the features, the results consistently showed the POS of the instance, the form in which it appears in the sentence, POS of the word before the instance, POS of the word after the instance and Bag of Words with a window of 3 were the most relevant features for Word Sense Disambiguation.

4.3 Dataset Analysis

The Above results are based only on the Semcor dataset. When Training the data on the Semcor+omsti dataset, the results are relatively similar concerning algorithms and features but overall the accuracy decreases.This could be because semcor is manually annotated while omsti is automatically generated Figure 2 represents the the comparative results of the feature and algorithms

4.4 Error Analysis

The misclassifications observed during the disambiguation process can be attributed to several factors that highlight the complexities in accurately determining word senses.

1. **Lack of Context:** Instances where the surrounding context failed to provide sufficient information for disambiguation led to misclassifications.

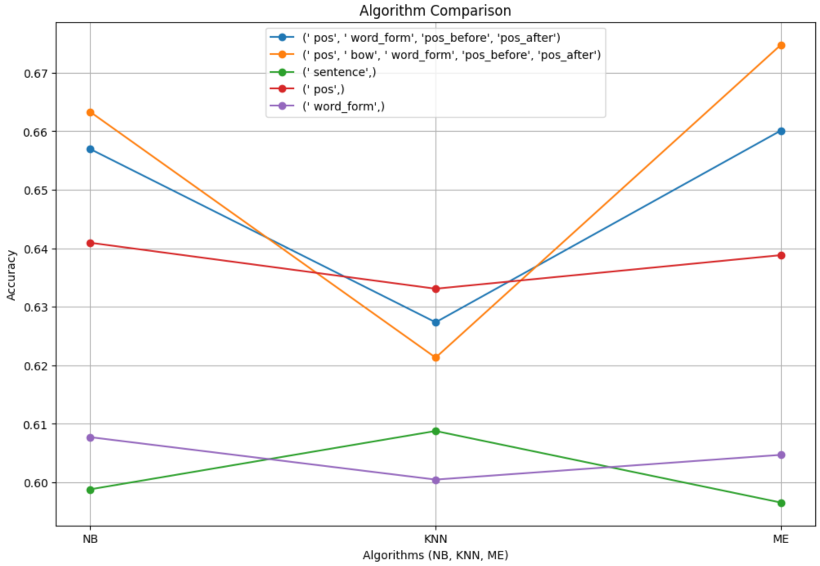


Figure 1: Semcor+omsti

Words in isolated or ambiguous contexts posed challenges for all algorithms across various feature representations. Example: I saw a bat. Here, bat can mean both the mammal and the sport equipment, without more context, it will be difficult even for a human to disambiguate it.

2. **Domain-Specific Words:** Words with domain-specific meanings or rare usages within the dataset context presented difficulties in accurate sense assignment. The algorithms struggled to discern less common word senses due to limited exposure within the training data. Example: ”The cell is dead.” Biology: could refer to a biological cell that is no longer living. Technology: it might imply a dead battery cell or a malfunctioning unit. It will be easy for a human to interpret the meaning of this instance given they are mostly aware of the domain. However, it could be difficult for computers to disambiguate this

3. **Polysemous vs Homograph Words:** Distinguishing between polysemous words (multiple related meanings) and homograph words (same spelling but unrelated meanings) revealed varying levels of complexity. Predictably, homographs were relatively easier to disambiguate compared to polysemous words. The distinct and unrelated meanings of homographs might have provided clearer contextual clues for classification given we use Bag of Words as a feature to disambiguate. Example: The leaves on the trees Here ’leaves’ has the root word ”leaf” noun and not ”leave” verb

5 Conclusion

This project examined the performance of Naive Bayes, K-Nearest Neighbors, and Maximum Entropy algorithms in word sense disambiguation across diverse feature representations. The obtained results shed light on the impact of algorithm choice and feature engineering on disambiguation accuracy.

The significance of algorithm selection was evident, with Maximum Entropy and Naive Bayes exhibiting more consistent and competitive performance compared to K-Nearest Neighbors.

However, feature representation played a crucial role in influencing disambiguation accuracy. The comprehensive set encompassing POS, BoW, Word Form, POS Before, and POS After yielded notably high accuracies across all algorithms, highlighting the importance of feature engineering in enhancing disambiguation performance.

References

[1] Alessandro Raganato, Jose Camacho-Collados, Roberto Navigli, et al. Word sense disambiguation: a uinified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, volume 1, pages 99–110, 2017.

[2] Hua Xu, Marianthi Markatou, Rositsa Dimova, Hongfang Liu, and Carol Friedman. Machine learning and word sense disambiguation in the biomedical domain: design and evaluation issues. *BMC bioinformatics*, 7(1):1–16, 2006.