Word Sense Disambiguation

Prof Nirali Arora

Word Sense Disambiguation

Word Sense Disambiguation (WSD) is a key task in Natural Language Processing (NLP) that involves determining which sense (or meaning) of a word is used in a given context. Many words have multiple meanings depending on how they are used, and WSD helps in distinguishing between these different senses.

Example on Word Sense Disambiguation

For example, the word "bank" could refer to a financial institution or the side of a river. In the sentence "She went to the bank to withdraw money," the sense of "bank" is related to finance. In the sentence "The boat was tied to the riverbank," the sense of "bank" is related to the land beside a river.

Word Sense Disambiguation Importance

WSD is important for various NLP applications like machine translation, information retrieval, and text understanding, because accurately interpreting the meaning of words is crucial for comprehending and processing text correctly.

Approaches of WSD

Knowledge based methods

Supervised learning methods

Unsupervised and semi supervised methods

Contextual Embeddings

Knowledge based methods

Dictionary and knowledge based methods: These avoid the use of corpus based evidence in favour of dictionaries ,thesauri and lexical knowledge bases. Supervised techniques : They use corpora with sense annotations as training data.

These use resources like dictionaries, thesauri, or ontologies to identify the correct sense of a word based on its context. For example, algorithms might compare the context of a word with definitions in a lexical database like WordNet.

Semi supervised or minimally supervised methods

Methods that are semi supervised or minimally supervised use a secondary source of information,such as a word-aligned bilingual corpus or a short annotated corpus used as seed data in a bootstrapping process.

These involve training machine learning models on labeled datasets where the correct sense of words is annotated. The model learns to predict the sense based on features extracted from the context.

Unsupervised and semi supervised methods

These approaches do not rely on labeled data. Instead, they use clustering or statistical methods to infer the senses based on patterns in large corpora of text.

These forsake (nearly entirely) external data in favour of working directly with unannotated raw corpora. Word sense discrimination is another name for techniques.

Contextual Embeddings

Modern approaches often use contextual embeddings from models like BERT or GPT, which understand word meanings in context by analyzing surrounding words. These models can dynamically adapt to different senses of a word based on the sentence they appear in.

Lesk algorithm

The Lesk algorithm was proposed by Michael Lesk in his 1986 paper titled *"Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone."* Michael Lesk, an early researcher in the field of computational linguistics, developed this algorithm as a method for Word Sense Disambiguation (WSD) by leveraging the definitions and examples available in machine-readable dictionaries.

Lesk Algorithm

The Lesk algorithm relies on the notion that words with similar meanings tend to appear in similar contexts. The basic idea is to choose the sense of a word that has the greatest overlap with the context in which the word appears.

Simplified outline :Lesk Algorithm

Context Extraction: Extract the context surrounding the ambiguous word. This context typically includes a few words before and after the target word in the sentence.

Sense Definitions: For each possible sense of the ambiguous word, retrieve its definition (and optionally, examples) from a lexical resource like WordNet.

Overlap Calculation: Compute the overlap between the context of the ambiguous word and each sense's definition. This can be done by comparing the words in the context with the words in each sense's definition.

Sense Selection: Choose the sense with the highest degree of overlap. The idea is that the sense whose definition shares the most words with the context is likely the correct one.

Example of Lesk Algorithm

Consider the ambiguous word "bank" in the sentence "She went to the bank to deposit money."

1. **Context Extraction: The context might be "She went to the bank to deposit money."**
2. **Sense Definitions: Suppose we have two senses for "bank":**
   - **Sense 1: A financial institution.**
   - **Sense 2: The side of a river.**
3. **Overlap Calculation:**
   - **For Sense 1: The definition might include terms like "financial institution" and "money." The overlap with the context ("deposit money") is high.**
   - **For Sense 2: The definition might include terms like "river" and "shore." The overlap with the context is low.**
4. **Sense Selection: Sense 1 would be chosen because it has a higher overlap with the context.**

Lesk Algorithm Limitations

Limitations

- **Dependence on Definitions: The Lesk algorithm depends heavily on the quality and granularity of the sense definitions available in the lexical resource. If definitions are too vague or too specific, the algorithm might not perform well.**
- **Limited Context: The algorithm can struggle with very limited context or when the definitions do not adequately capture the nuances of word meanings.**
- **Performance: While it provides a baseline for WSD, more modern methods often outperform the Lesk algorithm, particularly those that use advanced techniques like contextual embeddings.**