# ****photo_2025-05-31_01-03-33****

**Woldia University Institute of Technology**

**School of Computing**

**Department of Computer Science**

# Natural Language Processing Group Assignment

**Group Members:**

**No. Name ID**

**Submitted to: Mr. Girma. B**

### Submission Date: May 2025

**6.3. Lexical Semantics and Word-Sense Disambiguation**

1. Definition of Lexical Semantics and Explanation of Word Sense

**Lexical semantics** is a branch of linguistics and computational linguistics that focuses on the meaning of words and their relationships within a language. It examines how words convey meaning, how they relate to one another, and how they contribute to the meaning of larger linguistic units, such as phrases and sentences. Key aspects of lexical semantics include:

**Word Meanings**: The semantic properties of individual words, including their denotation (literal meaning) and connotation (implied or associated meanings).

**Sense Relations**: Relationships between words, such as:

* Synonymy: Words with similar meanings (e.g., "big" and "large").
* Antonymy: Words with opposite meanings (e.g., "hot" and "cold").
* Hyponymy: Hierarchical relationships where one word is a specific type of another (e.g., "rose" is a hyponym of "flower").
* Meronymy: Part-whole relationships (e.g., "wheel" is a meronym of "car").
* Compositionality: The principle that the meaning of a sentence or phrase is derived from the meanings of its constituent words and their syntactic arrangement.

**Concept of Word Sense**

A word sense refers to a specific, distinct meaning of a word. Many words are polysemous (having multiple related meanings) or homonymous (having multiple unrelated meanings). The correct sense of a word depends on the context in which it is used. For example:

- The word "bank" can refer to:

- A financial institution (e.g., "I deposited money in the bank.").

- The side of a river (e.g., "We sat by the bank of the river.").

- Other senses, such as a slope or a collection (e.g., "a bank of clouds").

- Word-sense disambiguation (WSD) is the task of identifying the intended sense of a word in a given context, which is critical for natural language processing (NLP) tasks like machine translation, information retrieval, and question answering.

**2. Methods for Word-Sense Disambiguation (WSD)**

WSD is the process of determining the correct sense of a word in a specific context. Below are two common methods for WSD: the Lesk algorithm and the machine learning-based approach.

**(A) Lesk Algorithm (Knowledge-Based Approach)**

The Lesk algorithm, proposed by Michael Lesk in 1986, is a knowledge-based method that uses dictionary definitions (glosses) to disambiguate word senses without requiring training data.

How it Works:

1. For a target word in a sentence, retrieve all possible senses and their corresponding dictionary definitions (glosses).

2. Retrieve the glosses of surrounding words (context words) in the sentence.

3. Compute the overlap by counting the number of shared words between the target word’s gloss and the glosses of context words.

4. Select the sense with the highest overlap as the most likely sense.

Example:

Sentence: "He went to the bank to withdraw money."

- Possible senses of "bank":

- Sense 1 (Financial Institution): "A financial establishment that invests money deposited by customers."

- Sense 2 (River Bank): "The land alongside a body of water."

- Context word: "money" (gloss: "A medium of exchange in the form of coins or bills.").

- Overlap:

- "Money" appears in the gloss of Sense 1 but not in Sense 2.

- Decision: The algorithm selects Sense 1 (financial institution)

**Advantages:**

- Simple and interpretable, requiring only a dictionary or lexical resource like WordNet.

- Does not require labeled training data, making it unsupervised.

**Limitations:**

- Performance depends heavily on the quality and richness of dictionary glosses.

- May fail when glosses are short or when there is insufficient overlap between glosses and context.

- Struggles with rare senses or domain-specific terms not well-covered in dictionaries.

- Variants of the Lesk algorithm, such as Simplified Lesk and Adapted Lesk, improve performance by incorporating additional information (e.g., example sentences or related words from lexical resources like WordNet).

- The algorithm is often used as a baseline in WSD research due to its simplicity but is outperformed by more advanced methods in most cases.

**(B) Machine Learning-Based Approach (Supervised WSD)**

Supervised WSD uses machine learning models trained on labeled datasets to predict the correct sense of a word based on contextual features.

How it Works:

- Training Phase:

- A dataset with words annotated with their correct senses (e.g., SemCor, a corpus where words are tagged with WordNet senses) is used.

- Features are extracted from the context, such as:

- Surrounding words (n-grams, e.g., words before and after the target word).

- Part-of-speech (POS) tags (e.g., noun, verb).

- Syntactic dependencies (e.g., subject-verb relationships).

- Semantic features (e.g., named entities or topic information).

- A classifier (e.g., Support Vector Machine, Naïve Bayes, or Neural Networks) is trained to map features to word senses.

- Prediction Phase:

- For a new sentence, the same features are extracted.

- The trained model predicts the most likely sense based on these features.

Example:

Sentence: "The fisherman stood on the bank."

- Features:

- Previous word: "the" (POS: determiner).

- Next word: "fisherman" (POS: noun).

- Context: "fisherman" is associated with water-related activities.

- The model, trained on similar examples, predicts \*Sense 2 (river bank)\* due to the contextual clue provided by "fisherman."

**Advantages:**

- Can achieve high accuracy with sufficient high-quality training data.

- Captures complex contextual patterns that knowledge-based methods like Lesk cannot.

- Modern neural models (e.g., BERT, ELMo) leverage contextual embeddings to improve performance.

**Limitations:**

- Requires large, high-quality annotated datasets, which are expensive and time-consuming to create.

- May not generalize well to unseen words, senses, or domains not present in the training data.

- Computationally intensive compared to knowledge-based methods.

- Recent advances in WSD use contextualized word embeddings from transformer-based models like BERT, which encode the entire sentence to capture nuanced contextual information. These models have significantly improved WSD performance, often achieving near-human accuracy on benchmark datasets.

- Datasets like SemEval and Senseval provide standardized tasks for evaluating WSD systems.

- Semi-supervised and unsupervised WSD approaches (e.g., using clustering or knowledge graphs) are also explored to reduce reliance on labeled data.

**1. Applications of WSD:**

- WSD is critical for NLP applications such as:

- Machine Translation : Ensuring the correct translation of ambiguous words (e.g., translating "bank" to French as "banque" for financial institution or "rive" for river bank).

- Information Retrieval: Improving search engine results by disambiguating query terms.

- Question Answering: Understanding user queries to provide accurate responses.

- WSD also supports sentiment analysis, text summarization, and dialogue systems by ensuring precise interpretation of word meanings.

**2. Challenges in WSD:**

- Fine-Grained Senses: Some lexical resources (e.g., WordNet) define very fine-grained senses, making disambiguation harder (e.g., distinguishing subtle variations of "run" like "run a race" vs. "run a business").

- Domain Adaptation: WSD models trained on general corpora may struggle with domain-specific senses (e.g., medical or technical terms).

- Multilinguality : WSD must account for language-specific ambiguities and sense inventories.

**3. Modern Approaches:**

- Deep Learning: Transformer-based models like BERT, RoBERTa, and XLM-R use contextual embeddings to represent words in context, outperforming traditional supervised methods. These models implicitly perform WSD by encoding sense-specific information in their representations.

- Knowledge-Enhanced WSD: Combining knowledge-based approaches (e.g., WordNet) with neural models to leverage both structured lexical knowledge and contextual understanding.

- Few-Shot and Zero-Shot WSD : Recent research explores using large language models (LLMs) to disambiguate senses with minimal or no training data, leveraging their pre-trained knowledge.

**4. Evaluation Metrics:**

- WSD systems are evaluated using metrics like accuracy (percentage of correctly disambiguated words) or F1-score for fine-grained sense distinctions.

- Benchmark datasets include SemCor, Senseval, and SemEval tasks, with WordNet often serving as the sense inventory.

**5. Comparison of Lesk and ML-Based Approaches:**

- Lesk is lightweight and interpretable but less accurate, especially for complex contexts or sparse glosses.

- ML-based methods, particularly those using deep learning, are more robust but require significant computational resources and annotated data.

- Hybrid approaches combining Lesk’s simplicity with ML’s robustness are an active area of research.

**Conclusion**

Lexical semantics explores the meanings of words and their relationships, while word sense refers to the distinct meanings a word can have based on context. The Lesk algorithm uses dictionary gloss overlaps for WSD, offering simplicity but limited accuracy. In contrast, supervised machine learning approaches leverage labeled data and contextual features for higher accuracy but are data-dependent. Advances in deep learning and hybrid methods continue to enhance WSD, making it a cornerstone of natural language understanding in modern NLP systems. For further exploration, consider investigating transformer-based WSD models or experimenting with WSD datasets like SemEval to deepen understanding of practical implementations.