Word Sense Disambiguation in the Uzbek Language

Axmedova Xolisxon Ilxomovna[1,*], Gulyamova Shakhnoza Qakhramonovna[2], Murtazayeva Umida Isakulovna[3], Mavlonov Bokhodir Biloljonovich[4] and Alimova Shahnoza Maqsudovna[5]

[1] *International Islamic Academy of Uzbekistan, Tashkent, Uzbekistan*
[2] *Tashkent University of Applied Sciences, Tashkent, Uzbekistan*
[3] *Tashkent University of Information Technologies, Samarkand, Uzbekistan*
[4] *Mahalla and Family Research Institute, Tashkent, Uzbekistan*
[5] *Urgench State University, Urgench, Uzbekistan*

Abstract

In the field of Natural Language Processing (NLP), semantic analysis remains one of the most important and relevant challenges. Word Sense Disambiguation (WSD) is a key task within semantic analysis, and the automatic identification of homonymous words requires the application of modern algorithms. Machine learning (ML) algorithms and transformer-based models are among such approaches. In this study, the K-Nearest Neighbors (K-NN), Random Forest, and SenseBERT algorithms were applied to disambiguate homonymous words in the Uzbek language. A dataset containing sentences for different senses of Uzbek homonymous words was compiled. The collected data set was subjected to initial pre-processing procedures. After cleaning, the models were trained using Random Forest and Sense BERT algorithms. The trained models were tested, achieving an average accuracy of 92 %. To further improve accuracy, it is recommended to expand the size of the dataset and train the models on high-memory computing resources.

Keywords

Word sense disambiguation, Machine learning approaches, K-NN algorithm, Random forest algorithm, Sense Bert model

1. Introduction

Computational linguistics is one of the rapidly developing fields of the 21st century. This discipline encompasses a wide range of natural language processing (NLP) tasks, and solving these tasks across different natural languages is of significant importance. Among these tasks are machine translation (MT), the development of question-answering systems, automatic text analysis, and the identification of named entities (NER), to name a few. Each of these tasks consists of multiple subtasks. For example, text analysis generally involves three levels: morphological analysis [2], syntactic analysis [8], and semantic analysis. Each type of analysis follows its own hierarchical set of processes. This article presents an overview of semantic analysis and its components.

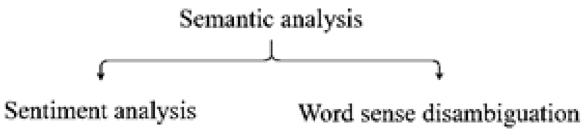


Figure 1: Types of semantic analysis

ICAIW 2025: Workshops at the 8th International Conference on Applied Informatics 2025, October 8–11, 2025, Ben Guerir, Morocco

*Corresponding author.
xolisa9029@mail.ru (A. X. Ilxomovna); bunnycham@mail.ru (G. S. Qakhramonovna); murtazayeva198202@gmail.com (M. U. Isakulovna); mavlonov@moti.uz (M. B. Biloljonovich); shaxnozaalimova103@urdu.uz (A. S. Maqsudovna)
0000-0002-9828-1650 (A. X. Ilxomovna); 0000-0002-9633-873X (G. S. Qakhramonovna); 0000-0003-2423-5149 (M. U. Isakulovna); 0009-0001-2073-9473 (M. B. Biloljonovich); 0009-0009-7837-8509 (A. S. Maqsudovna)

Semantic analysis is itself divided into two categories: sentiment analysis and word sense disambiguation. While sentiment analysis evaluates the emotional tone of a text, word sense disambiguation helps to determine the intended meaning of a word within its context. The task of word-sense disambiguation is considered one of the core areas of NLP and plays a crucial role in the field. Solving the problem of identifying the correct meaning of a word in a natural language is achieved by semantically distinguishing between different types of words.

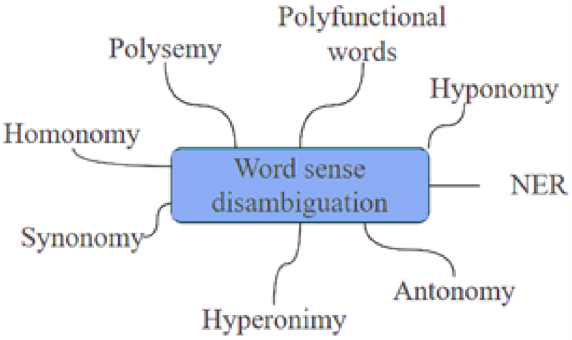


Figure 2: Elements of word sense disambiguation

In numerous natural languages around the world, this problem has been explored and addressed effectively. The commonly adopted approaches to word-sense disambiguation are presented below:

- Knowledge-based approaches
- Machine learning-based approaches
  - Supervised learning algorithms
  - Unsupervised learning algorithms
  - Semi-supervised learning algorithms
- Neural network models

Numerous studies have been conducted across a wide range of the world's languages to address the problem of word sense disambiguation. In recent years, modern approaches have been applied to tackle this task. For example, in Russian, neural network models have been utilized to identify homonyms [9], while in Kazakh [10], Arabic [11], Hindi [12], English [13], and again in Russian [14], various machine learning algorithms and deep learning techniques have been employed to detect homonymous and polysemous words.

Similarly, in the Uzbek language, a number of studies have been conducted to determine the meaning of words. The problem of word-sense disambiguation can be addressed using knowledge-based approaches, though their accuracy is relatively limited. Such approaches are typically applied to determine the contextual meaning of ambiguous words within a specific domain [1, 2, 3]. However, to disambiguate the meaning of a polysemous word in any given context, the use of machine learning-based algorithms has proven to be more effective [6].

Supervised learning algorithms require a dataset consisting of semantically tagged data that belongs to specific categories $(k_1 ... k_n)$. Each data point within the dataset is represented by a set of features $(f_1 ... f_2)$. The goal of the algorithm is to learn and identify the relationships between these features and their corresponding categories, enabling it to accurately classify new, unseen data based on the learned patterns.

To determine word meaning in context, supervised learning requires a semantically tagged dataset or corpus. For instance, when disambiguating polysemous words in English, the Sem-Cor corpus is commonly used. Sem-Cor is a subset of the Brown Corpus, containing 234,000 words, where the lexical units in each sentence are annotated with their corresponding WordNet senses. Naturally, these WordNet sense annotations for the lexical units were manually labeled by human annotators [7].

When applying these algorithms, two main types of features are typically used, often combined in various ways: collocational features and co-occurrence features.

Collocational features refer to specific words (along with their part-of-speech (POS) tags) that appear at certain positions immediately to the left or right of the target word.

```
"Traktor matorining shovqini suruvdagi otlarni hurkitib yubordi"
```

If the target word in this sentence is "ot+larni", its feature vector - consisting of the two words to the left and two words to the right — would be as follows.

```
[shovqin, N, suruv, N, hurkitmoq, V, yubormoq, V].
```

The feature vector is made up of the lemmas of the two preceding and two following words relative to the current ambiguous word, along with their corresponding part-of-speech (POS) tags.

2. Main Part

Co-occurrence features rely on the words surrounding the target word. In this approach, the features are the surrounding words themselves, without considering their part-of-speech (POS) categories. The value of each characteristic indicates how frequently these words appear in the context of the target word.

To apply this method effectively, a small number of meaningful words that frequently occur near the target word are typically selected as features. For example, for the word "ot", the most frequently occurring surrounding words from sentences containing its different senses might include the following:

```
tog’, dara, chipor, toy, poyga, dala, ...
"Zotdor otlar orasida poyga uchun qabul davom etmoqda. . . "
```

If the window size is set to 10, this sentence can be represented as the following vector.

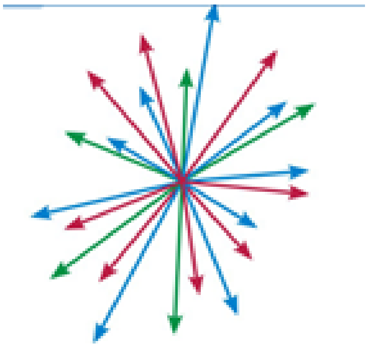


Figure 3: Mean vectors of sentence

2.1. Unsupervised learning algorithm.

In this approach, we start with an untagged training dataset — meaning we do not know which class each data point belongs to. Only the features are available, and the algorithm must independently identify which data points belong to the same class based on the patterns it discovers in the data.

Machine learning approaches and transformer-based models offer significant capabilities for identifying the contextual meaning of homonymous words. In this study, we utilize machine learning algorithms to semantically distinguish homonymous words within sentences in the Uzbek language.

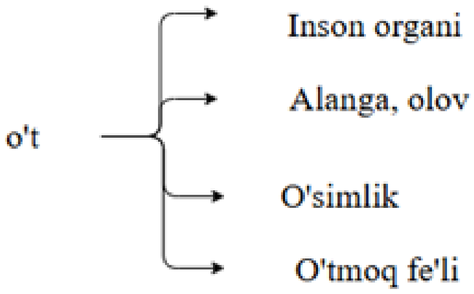


Figure 4: Senses of "o't" homonym word

2.1.1. K-NN algorithm

K-NN is a supervised machine learning algorithm that applies a classification method based on semantic similarity [4]. For an ambiguous word, it identifies the k nearest similar instances and determines the word's meaning based on these neighbors. The sequence of this process is illustrated in Figure 5.

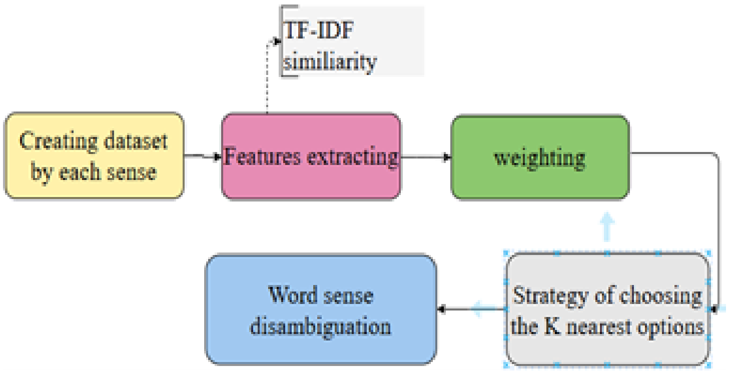

TF-IDF similiarity
Creating dataset by each sense
Features extracting
weighting
Strategy of choosing the K nearest options
Word sense disambiguation

Figure 5: Procuresses of K-NN classifier

Feature extraction refers to the process known in English as Feature Extraction [4]. In developing a WSD (Word Sense Disambiguation) system, correctly extracting relevant features is crucial. The set of features typically includes the following types:

2.1.2. Term Frequency (TF)

In this method, the most frequently occurring words in the text are selected. Based on TF, the words that appear most often in the context of a particular sense of an ambiguous word are identified, as they help determine the intended meaning of the ambiguous word [4]. When using the K-Nearest Neighbors (K-NN) algorithm, the following steps are performed:

1. Data Preparation: The available data consists of items separated by other words (for example, words in a text).
2. Distance Measurement: The similarity between words can be measured using a distance metric. In the K-NN algorithm, Euclidean distance is commonly used as a distance measure
3. Finding the k Nearest Neighbors: For a new word, the k closest neighboring words are selected
4. Learning and Classification: Based on these k neighbors, the meaning of the new word is determined. In other words, the sense of an ambiguous word in a newly provided sentence is identified by relating it to the overall meaning of its k neighboring words [5].

In word sense disambiguation, the position and frequency of words within a text play an important role. The weight of each word can be calculated based on the following formula:

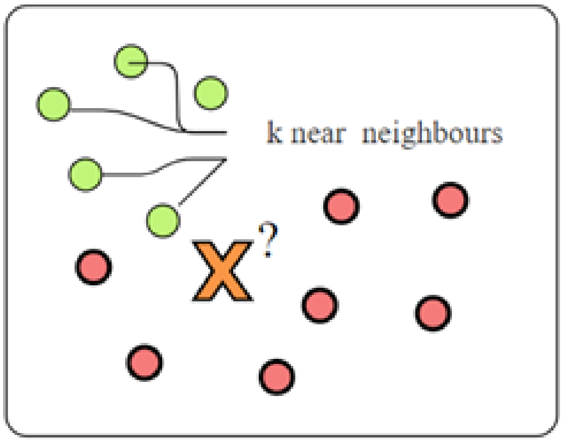


Figure 6: Identifying neighboring words

2.1.3. Weighting Strategy

The significance (weight) of a word is determined according to its frequency in the context and its relative position within the sentence. This weighting approach helps prioritize more contextually relevant words when identifying the meaning of an ambiguous word.

$$w(k, f_i) = \frac{N(k, f_i)}{N(k)} \tag{1}$$

Here: $N(k, f_i)$ – the number of feature words f_i that co-occur with the word in sense k, $N(k)$ – the total number of samples for sense k.

Table 1
Statistics of homonymous words

Words	With surrounding words (%)	With frequently occurring words (%)	When using both together (%)
bo‘ydor	90.7	89.7	90.7
bo‘sh	76.8	75.8	76.8
bo’yli	75.6	72.6	75.6
bog’li	78.6	82.1	78.6
sirli	61.2	56.9	63.8
belli	75.5	73.8	76.1

Average accuracy: 76.1

The proposed method outperformed previously developed approaches.

2.1.4. Random forest

This algorithm uses multiple decision trees and yields good results in identifying complex homonyms. Let’s review the process of determining the meanings of homonymous words in a text using the Random Forest algorithm. This process consists of the following main stages:

- Homonyms and their context-based sentences are identified through a dataset (Excel file).
- The model is trained and then used to determine word senses in new texts
- The model’s accuracy is evaluated through testing.

To build a model using the Random Forest algorithm, sentences are first collected for each meaning of the homonymous words. The dataset for this algorithm was compiled in the following format.

Libraries required for model development (Figure 8).

Model Training and Evaluation

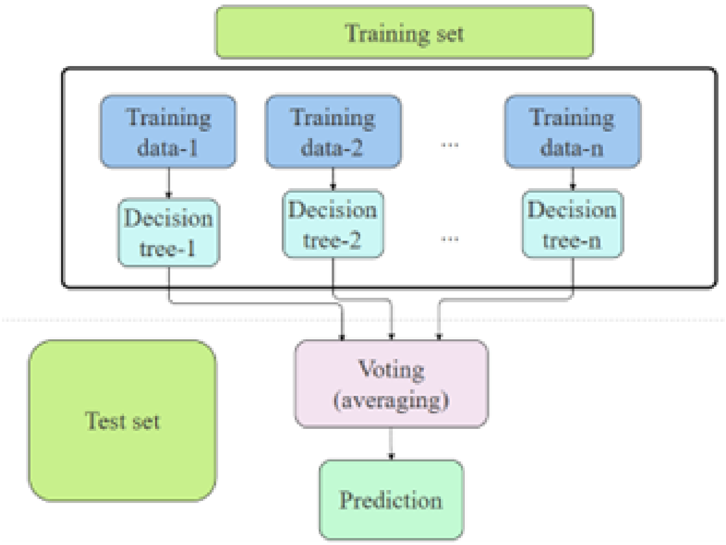
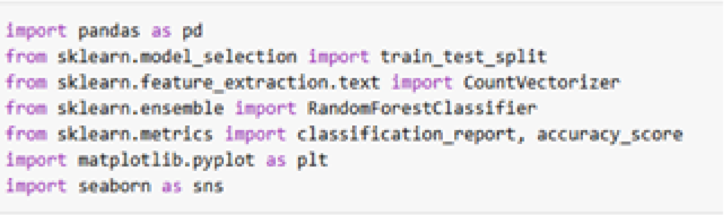


Figure 7: Processes of the Random Forest Algorithm

Table 2
Example of a dataset prepared for the model

Words	Context	Tag_sense
Kasr	Kasr sonlarni o‘nli kasrga aylantirish bo‘yicha dars o‘tdik	Siniq, parcha. Matematika termini; miqdor birligining qismi
Kasr	Bir litr sutning yarm kasri qaynatildi	Siniq, parcha. Matematika termini; miqdor birligining qismi.
Kasr	Bog‘ning uchdan bir kasri sabzavot ekish uchun ishlatildi.	Siniq, parcha. Matematika termini; miqdor birligining qismi
Kasr	U kasr tufayli yugurish musobaqasida qatnasha olmadi	Shikastalik, nuqsonlilik
Kun	Kun bulutlarni yorib, qishloqqa issiqlik berdi	Yerga issiqlik va nur taratib turuvchi planeta, quyosh, oftob
Kun	U kun chiqishini suratga olish uchun dron ishlatdi	Yerga issiqlik va nur taratib turuvchi planeta, quyosh, oftob.



```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, accuracy_score
import matplotlib.pyplot as plt
import seaborn as sns
```

Figure 8: Used libraries for creating models

1. Sentences and their corresponding senses were extracted from an Excel file.
2. The sentences were converted into vector representations using the CountVectorizer method.
3. The dataset was divided into training and testing subsets in an 80/20 ratio using the train-test-split function.
4. A RandomForestClassifier() was employed to train the model.

3. Results

Evaluation of the Trained Model

During the model development process, a total of 54,000 semantically annotated sentences were collected. Of these, 43,200 sentences were used to train the model. The resulting model was then tested on 10,800 instances. The test dataset contains 20 sentences for each meaning of Uzbek noun homonyms. In the test results, each homonym was analyzed separately. The results of the analysis can be seen in Table 3.

Table 3
F1 scores of homonyms according to their different meanings

Words	Sense-1	Sense-2
Kasr	0.64	0.53
Kun	0.72	0.52
Ado	0.62	0.61
Alam	0.68	0.42
Aks	0.69	0.41
...	...	...

In this article, the $F1$ scores are calculated for the words presented in Table 2. Based on the results of the testing phase, the $F1$ score was calculated to evaluate the performance of the model. The test results are summarized below:

- Total phrases recognized as valid expressions: 10800
- Correctly identified synonyms: 9950
- Incorrectly matched synonyms: 50

To assess the overall performance of the model, the $F1$ score was calculated based on these results.

$$F1 = 2 * \frac{Pricision * Recall}{Pricision + Recall} \tag{2}$$

$$Pricision = \frac{1000}{1000 + 50} \tag{3}$$

$$Recall = \frac{1000}{1000 + 9950} \tag{4}$$

Based on the calculated precision values (3) and recall (4), (2) was found to be 0.664.

The data was extracted from a corpus containing 460 million words, which includes news articles, scientific abstracts, spoken-language transcripts, and literary texts. In terms of size, this represents one of the largest training and testing datasets used in the field of word-sense disambiguation. The compiled dataset was used to train a modern transformer-based model, specifically the Sense BERT algorithm, and the resulting model was subsequently evaluated. The model trained with Sense BERT demonstrated significantly higher accuracy compared to traditional approaches. In particular, the same data set size was used for both the Random Forest and Sense BERT algorithms. The Sense BERT model achieved an accuracy score of 92 %.

4. Conclusion

Modern approaches to word-sense disambiguation rely on several empirically observed linguistic properties, particularly the principle that a word typically conveys a single meaning within a given collocational and discourse context. Algorithms aim to exploit these properties by modeling the diversity of collocational relationships as effectively as possible. In this study, the problem of identifying the contextual meanings of homonymous words in the Uzbek language was addressed using Random Forest, K-Nearest Neighbors (K-NN) and Sense BERT algorithms. Given that homonyms carry different meanings depending on the context in which they occur, automatically determining their intended sense is one of the essential challenges in natural language processing. The primary goal was to identify the meaning of a homonymous word in a user-provided text based on its context.

During the model development phase:

- A dataset in Excel format was compiled, containing columns for the homonym, the sentence, and its corresponding meaning
- The texts were cleaned and lemmatized.
- Sentences were converted into numerical representations using vectorization techniques such as TF-IDF.
- The Random Forest algorithm was selected for model training because:
  - It achieves high accuracy in classification tasks.
  - It is resistant to overfitting.
  - It provides interpretable results.

The final model successfully identified the contextual meaning of homonymous words in user-submitted texts.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] Elov, B., Abdurakhmonova, N., Axmedova, X., Xusainova, Z., Iskandarova, A., and Fattaxova, D. Designing Processes and Models of Semantical Differentiation for Polyfunctional Words in the Uzbek Contexts. In: Nguyen, N.T., et al. Recent Challenges in Intelligent Information and Database Systems. ACIIDS 2024, Communications in Computer and Information Science, vol. 2145. Springer, Singapore (2024).

[2] Boltayevich, E.B., Mirdjonovna, H.S., and Ilxomovna, A.X. (2023). Methods for Creating a Morphological Analyzer. In: Zaynidinov, H., Singh, M., Tiwary, U.S., and Singh, D. (Eds.), Intelligent Human Computer Interaction. IHCI 2022, Lecture Notes in Computer Science, vol. 13741. Springer, Cham.

[3] Boltayevich, E.B., Ilxomovna, A.X., Hakim Qizi, P.M., and Uktambay O'g'li, K.N. (2023). Semantic Differentiation of Uzbek Homonyms Using the Lesk Algorithm. 2023 8th International Conference on Computer Science and Engineering (UBMK), Burdur, Turkiye, pp. 137–140.

[4] Comparable Corpora to improve SMT performance Sadaf. 12th Conference of the European Chapter of the ACL, pages 16–23, Athens, Greece, 30 March – 3 April 2009.pp 16-23

[5] Gale, K. Church, and D. Yarowsky.: A Method for Disambiguating Word Senses in a Large Corpus. Computers and Humanities, vol. 26, pp. 415-439 (1992).

[6] Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, pp. 2825–2830

[7] Mihalcea, R., and Moldovan, D. (2002). Word sense disambiguation with pattern learning and automatic feature selection. Natural Language Engineering, 8(4), 349–370.

[8] Oqila Abdullayeva. O'zbek tili matnlarida sintaktik teg va teglash masalasi. O'zbekiston: til va madaniyat. Kompyuter lingvistikasi. 2024 Vol. 1 (6) B. 46-58.

[9] Ayazbayev, D.; Bogdanchikov, A.; Orynbekova, K.; Varlamis, I. Defining Semantically Close Words of Kazakh Language with Distributed System Apache Spark. Big Data Cogn. Comput. 2023, 7, 160.

[10] El-Razzaz, M., Fakhr, M. W., and Maghraby, F. A. (2021). Arabic Gloss WSD Using BERT. Applied Sciences, 11(6), 2567

[11] Zhong, Z., Ng, H.T.: It makes sense: a wide-coverage word sense disambiguation system for free text. In: ACL 2010—48th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 78–83 (2010)

[12] Zaripova D.A., Loukachevitch N.V. (2023) Approaches to Automatic Word Sense Disambiguation Based on Uneven Distribution of Word Senses in Corpus. Lomonosov Philology Journal. Series 9. Philology, no. 6, pp. 40–51

[13] F. Meng, Graph and Word Similarity for Word Sense Disambiguation, 2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Chengdu, China, 2020, pp. 1114-1118

[14] D. Sumanathilaka, N. Micallef and J. Hough, Exploring the Impact of Temperature on Large Language Models: A Case Study for Classification Task Based on Word Sense Disambiguation, 2025 7th International Conference on Natural Language Processing (ICNLP), Guangzhou, China, 2025, pp. 178-182

[15] U. R. Dhungana, S. Shakya, K. Baral and B. Sharma, Word Sense Disambiguation using WSD specific WordNet of polysemy words, Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015), Anaheim, CA, USA, 2015, pp. 148-152