Contextual Embeddings for Ukrainian: A Large Language Model Approach to Word Sense Disambiguation

Yurii Laba
The Machine Learning Lab at
Ukrainian Catholic University,
Lviv, Ukraine
laba@ucu.edu.ua

Volodymyr Mudryi
Independent Researcher,
Lviv, Ukraine
vova.mudruy@gmail.com

Dmytro Chaplynskyi
Lang-uk
Kyiv, Ukraine
chaplinsky.dmitry@gmail.com

Mariana Romanyshyn
Grammarly
Kyiv, Ukraine
mariana.scorp@gmail.com

Oles Dobosevych
The Machine Learning Lab at
Ukrainian Catholic University,
Lviv, Ukraine
dobosevych@ucu.edu.ua

Abstract

This research proposes a novel approach to the Word Sense Disambiguation (WSD) task in the Ukrainian language based on supervised fine-tuning of a pre-trained Large Language Model (LLM) on the dataset generated in an unsupervised way to obtain better contextual embeddings for words with multiple senses. The paper presents a method for generating a new dataset for WSD evaluation in the Ukrainian language based on the SUM dictionary. We developed a comprehensive framework that facilitates the generation of WSD evaluation datasets, enables the use of different prediction strategies, LLMs, and pooling strategies, and generates multiple performance reports. Our approach shows 77,9% accuracy for lexical meaning prediction for homonyms.

1 Introduction

Word Sense Disambiguation (WSD) task involves identifying a polysemic word's correct meaning in a given context. A task of WSD is applicable in various NLP fields Sharma and Niranjan (2015), such as information retrieval, machine translation Neale et al. (2016), and question answering. For well-resourced languages, this problem has many different approaches for solving that demonstrate competitive results Navigli (2009).

However, this task has received relatively little attention in the Ukrainian language due to the absence of sense-annotated datasets. To address this issue, we propose a novel approach to the WSD task based on fine-tuning a pre-trained Large Language Model (LLM) to obtain better contextual embeddings for words with multiple senses.

In this research, we present a method for generating a new dataset for WSD evaluation in the Ukrainian language, which includes lemmas, example sentences, and lexical meanings based on the SUM dictionary, of NAS of Ukraine (ULIF-NASU). This dataset is used to evaluate the effectiveness of our proposed method. For supervised LLM fine-tuning, we generate the dataset in an unsupervised way based on UberText Chaplynskyi (2023).

Additionally, we have developed a comprehensive framework [1] that facilitates the generation of WSD evaluation datasets, enables the use of different prediction strategies, LLMs, or pooling strategies, and generates multiple performance reports.

2 Related works

Early approaches in WSD utilized the concept of word embeddings, which were generated using pre-trained algorithms such as Word2Vec Mikolov et al. (2013) or Glove Pennington et al. (2014). However, these static word embeddings have a notable problem that all senses of a homonym word must share a single vector. To address this issue, several researchers have proposed techniques for capturing polysemy and generating more informative embeddings Faruqui et al. (2014) or Speer et al. (2017). Recently, there has been a trend toward utilizing contextual embeddings generated by LLMs instead of pre-trained word embeddings. These contextual embeddings provide a more nuanced representation of words, capturing context-specific information. As a result, a simple approach such as kNN can be used in combination with these embeddings to predict word senses in Word Sense Disambiguation tasks accurately Wiedemann et al. (2019).

WSD can be approached as a binary classifica-

[1] More details in Appendix A

Proceedings of the Second Ukrainian Natural Language Processing Workshop (UNLP), pages 11–19
May 5, 2023 ©2023 Association for Computational Linguistics

tion problem. One such approach was proposed by Huang et al. (2019), which involved adding a classification head to the BERT model Devlin et al. (2018). The model takes a pair of sentences as input, with one sentence containing the target word and the other providing one of the possible definitions of the target word. The model then predicts whether the target word in the sentence has the same meaning as the definition.

Another noteworthy approach to Word Sense Disambiguation is the one presented by Barba et al. (2021), where the model not only takes into account the contextual information of the target word, but also the explicit senses assigned to neighboring words.

Despite the high performance of the previously mentioned supervised approaches for Word Sense Disambiguation, their reliance on a large amount of annotated sense data can pose a challenge for their application to under-resourced languages. In contrast, unsupervised methods can also be applied to WSD tasks. One of the earliest and most well-known solutions is using sense definitions and semantic relations from lexical graph databases such as Babelfy Moro et al. (2014). However, recent works such as Huang et al. (2019) have shown that LLM-based solutions outperform those methods.

Given the limitations of prior research, particularly the shortage of annotated corpora in the Ukrainian language, we present our proposed solution of supervised fine-tuning of an LLM on a dataset generated in an unsupervised way. Additionally, we have prepared a validation dataset for the Ukrainian WSD task, derived from the SUM (Dictionary of Ukrainian Language) dictionary of NAS of Ukraine (ULIF-NASU).

Our approach will enhance the model's understanding of semantic word meaning and improve the performance of the Word Sense Disambiguation task in the Ukrainian language.

3 Evaluation Dataset

To assess the efficacy of our methodology for addressing the Ukrainian WSD task, we have established a validation dataset based on the SUM dictionary. The SUM dictionary is an appropriate resource as it employs componential analysis, a linguistic methodology used to differentiate common language phenomena such as polysemy and homonymy, by evaluating the presence or absence of shared semantic features among compared units. Therefore, the dataset derived from the SUM dictionary is well-suited for evaluating the performance of our approach. According to Ukrainian Lingua-Information Fund (2022), the examples in the SUM dictionary were taken from a broad selection of resources, including fiction (from the end of the 18th century to the present day), Ukrainian translations of the Bible, folklore, publicistic, scientific, and popular scientific works, the language of the mass media, the language of the Internet, etc. Unfortunately, at the moment of publication, there is only part of the dictionary available (until word ПІДКУРЮВАЧ (en: lighter, translit: pidkuryuvach)).

The dataset was constructed by extracting each lemma, its lexical meaning, and examples of usage related to that meaning. While building the evaluation dataset, the lemmas with single possible lexical senses were filtered out, and the resulting dataset consisted of 78,000 samples. Further data cleaning was performed to remove lemmas with a length of fewer than three characters, lemmas with missing senses or examples, lemmas that belong to functional parts of speech, and lemmas which lexical meaning reference for another lemma. After cleaning, the dataset consisted of 42,000 samples, with each sample consisting of a lemma, one of the possible lexical meanings of the lemma, and examples of this meaning. Assembling the dataset involved part-of-speech (POS) detection for each lemma using the Stanza library Qi et al. (2020), and this information was utilized in the subsequent evaluation table.

During our experiments, we observed that many lemmas in the Ukrainian language have multiple similar lexical meanings, which significantly complicates the task, the examples presented in Table 1. To address this issue, we built a dataset focusing on homonymy rather than polysemy.

Homonyms are unrelated words with the same written and spelling form but different lexical meanings. To construct a dataset of homonyms, we first filtered out lemmas with fewer than two entries in the SUM dictionary. Then, for each remaining lemma, we concatenated all the lexical meanings and examples of usage of each separate homonym. The resulting dataset consisted of 2,882 homonym samples, each sample including the lemma, its possible meanings, and examples for each meaning (see Table 2). We used this dataset for further model evaluation.

Lemma	Meaning	Example
КОСА (en: braid, transl: kosa)	Заплетене волосся (en: Braided hair)	Очі в неї були великі, дві чорні коси, перекинуті наперед, обрамляли лице. (en: Her eyes were large, two black braids, thrown forward, framed her face.)
КОСА (en: braid, transl: kosa)	Довге волосся (en: Long hair)	Густі, золото-жовті коси буйними хвилями спадали на її груди і плечі. (en: Thick, golden-yellow braids fell in wild waves on her chest and shoulders.)

Table 1: Examples from polysemy dataset (similar lexical meanings)

4 Approach

4.1 Task Definition

In our approach to Word Sense Disambiguation, for each homonym l (target word), we have identified a set of possible lexical meaning groups, denoted as

$$G_l = \{g_{l_1}, ..., g_{l_n}\}$$

Each lexical meaning group g_{l_i}, comprises all the possible lexical meanings of a particular lemma corresponding to the homonym. Our objective is to predict the correct lexical meaning group g_{l_i}, from all the possible lexical meaning groups of the lemma G_l, based on a list of examples of the lemma's usage.

To accomplish this, we first calculate embeddings for the sentence example and obtain the target word embedding from it using various pooling strategies, which will be described later. Subsequently, we measure the cosine similarity between the obtained embedding of the target word and the embeddings of each lexical meaning group. The lexical meaning group with the highest cosine similarity is considered to be the predicted context. Figure 1 demonstrates an example of the single lemma prediction process utilizing our approach.

4.2 Evaluation

In order to evaluate the performance of our WSD approach, we have chosen to utilize the accuracy metric. Specifically, for each sample in the dataset, we compare the predicted context of the lemma (see Figure 1) with the ground truth context derived from the corresponding example. Any instances where the predicted context matches the ground truth context are considered correct predictions, and the overall accuracy is calculated based on the total number of correct predictions.

4.3 Embedding calculation

In the context of natural language processing (NLP), word embeddings have emerged as a powerful technique to represent words in a numerical form, which can then be leveraged to perform various NLP tasks, including Word Sense Disambiguation. Each word is mapped to a high-dimensional vector of real numbers in word embeddings, which encodes its semantic and syntactic information based on its context in a given corpus. By capturing words' intrinsic meaning and contextual usage, word embeddings have demonstrated their effectiveness in various NLP applications, including WSD Huang et al. (2019).

In NLP, one of the most effective approaches for generating high-quality contextualized word embeddings is leveraging pre-trained LLMs such as RoBERTa Liu et al. (2019) or GPT-2 Radford et al. (2019). LLMs allow the calculation of word embeddings for individual words or entire sentences. For instance, the BERT (Bidirectional Encoder Representations from Transformers) base model Devlin et al. (2018) employs 12 layers of transformer encoders, which utilize a multi-head attention mechanism to learn context-dependent representations of input tokens. The resultant output vector of each token from each layer of the BERT model can be used as a word embedding.

Various pooling strategies can be applied to generate embeddings for individual words or entire sentences, but determining the most effective strategy for a particular task requires experimental investigation. In this study, we conducted experiments to compare the performance of different pooling methods, including:

1. Mean pooling - computes the average of the embeddings for each token from the last hidden state of the model. The last hidden state

Lemma	Meaning	Example
КОСА (en: braid, transl: kosa)	[Заплетене волосся (en: Braided hair), Довге волосся (en: Long hair)]	[Очі в неї були великі, дві чорні коси, перекинуті наперед, обрамляли лице. (en: Her eyes were large, two black braids, thrown forward, framed her face.); Густі, золото-жовті коси буйними хвилями спадали на її груди і плечі. (en: Thick, golden-yellow braids fell in wild waves on her chest and shoulders.)]
КОСА (en: scythe, transl: kosa)	[Сільськогосподарське знаряддя для косіння трави, збіжжя тощо, що має вигляд вузького зігнутого леза, прикріпленого до держака. (en: An agricultural tool for mowing grass, grain, etc., having the form of a narrow bent blade attached to a handle.)]	[Внук косу несе в росу. (en: A grandson carries a scythe into the dew.)]

Table 2: Examples from the homonym dataset

corresponds to the sequence of hidden states at the output of the model's final layer.

2. Max pooling - extracts the maximum value of the embeddings for each token from the last hidden state of the model.

3. Mean Max pooling - calculates the average and maximum values of the embeddings for each token from the last hidden state of the model and concatenates the resulting vector.

4. Concatenate pooling - concatenates the embeddings from the last four hidden states.

5. Last four or two pooling - sums the embeddings from the last four or two hidden states.

Based on our experiments, we concluded that the mean pooling shows the best results in the WSD task for the Ukrainian language (see Table 3).

Our research aimed to determine the most effective LLM for generating contextual embeddings. To achieve this, we conducted experiments using a range of multilingual LLMs and evaluated their performance without fine-tuning. Our results in Table 3 demonstrates that one of the SBERT models Reimers and Gurevych (2019), namely paraphrase-multilingual-mpnet-base-v2 (PMMBv2), produced the highest quality contextual embeddings for our WSD task on a homonym dataset. Interestingly, our findings suggest that the SBERT model, initially designed to improve the semantic representation of entire sentences, can also significantly enhance the semantic representation of individual words.

5 Embeddings improvement

5.1 Dataset for fine-tuning

In order to enhance the quality of embeddings and to achieve superior performance on words with multiple lexical senses, we opted to fine-tune our best model, PMMBv2, as a means to improve its efficiency. Typically, researchers rely on supervised datasets such as Semcor Miller et al. (1993) or SemEval-2007 Pradhan et al. (2007) to enhance WSD task performance, consisting of pairs of sentences and a sense for a particular lemma, along with binary labels indicating the usage of a lemma in that particular context. Unfortunately, no such dataset is available for the Ukrainian language, leading us to pursue fine-tuning our model using a dataset generated using our proposed unsupervised method.

Our dataset samples consist of an anchor, a positive, and a negative example. To define positive and

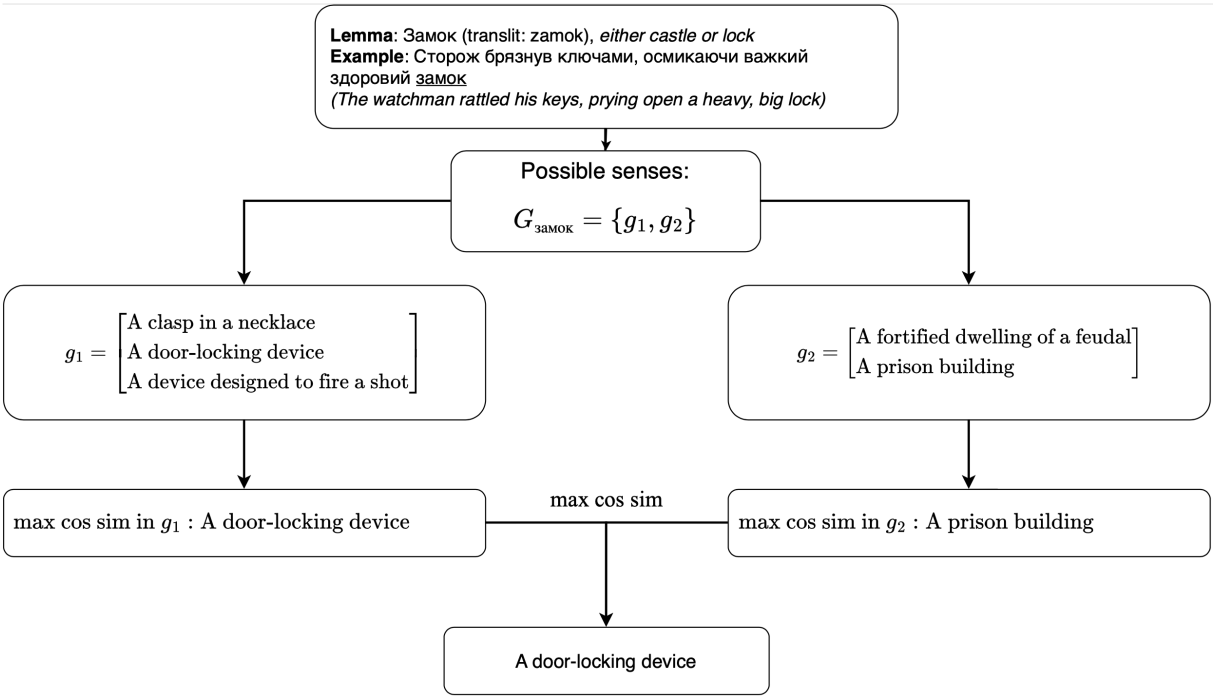
Lemma: Замок (translit: zamok), either castle or lock
Example: Сторож брязнув ключами, осмикаючи важкий здоровий замок
(The watchman rattled his keys, prying open a heavy, big lock)
Possible senses:
$G_{\text{замок}} = \{g_1, g_2\}$
$g_1 =$ A clasp in a necklace
A door-locking device
A device designed to fire a shot
$g_2 =$ A fortified dwelling of a feudal
A prison building
max cos sim in g_1 : A door-locking device
max cos sim
max cos sim in g_2 : A prison building
A door-locking device

Figure 1: Prediction Logic for Lemma "Замок" (translit: zamok). In Ukrainian, the lemma "zamok" has two possible meanings. The first one means a castle, and the second is a lock. In this figure, we depicted prediction logic given an example, lemma of interest, and possible senses.

negative examples relative to the anchor sentence for the WSD task, we determined that the positive sample should be a sentence with a lemma used in the same context as in the anchor sentence. In contrast, the negative sample should be a lemma used in a different context.

In order to acquire a suitable dataset, an entirely unsupervised methodology was employed. The Developers' preview of UberText 2.0 Chaplynskyi (2023), which comprises of texts from Ukrainian periodicals, was utilized to gather a vast number of Ukrainian language sentences. Subsequently, we filtered out sentences that did not contain any lemmas from our homonym evaluation dataset (Evaluation Dataset). We removed outliers based on criteria such as length and the presence of punctuation symbols or digits. We also employed langdetect Shuyo (2010) to remove non-Ukrainian language samples.

Each dataset sample was then represented as an embedding using ukr-roberta Radchenko (2020). We calculated the cosine distance between the anchor embedding and all other sentences in the dataset containing the required lemma. We then assumed that the sample with the highest cosine similarity would be the positive sample - containing a lemma used in the same context as in the anchor sentence and that the sample with the lowest cosine similarity would be the negative sample, containing a lemma used in a different context.

This dataset is available in two sizes, consisting of ~190,000 and ~1,200,000 triplet pairs obtained from UberText 2.0.

We assessed the suitability of our dataset for fine-tuning by selecting a subset of examples to determine if target lemmas in positive and negative instances have distinct lexical meanings. After sampling approximately 100 examples, we found that 13.1% of the samples constituted relevant triplets. In the Conclusion section of this paper, we will provide future works for enhancing the dataset's quality.

5.2 Loss

Given that we had access to a suitable dataset, we opted to employ the TripletMarginLoss Balntas et al. (2016) for fine-tuning our neural network.

The Triplet Margin Loss function is used to optimize a neural network by minimizing the distance between the embedding of an anchor sentence and that of a positive example while maximizing the distance between the anchor and a negative example. The loss function is defined as follows:

$$max(||a-p|| - ||a-n|| + M, 0)$$

Model	Mean	Max	Mean Max	Concatenat	Last four	Last two
bert-base-multilingual-cased	0.602	0.601	0.622	0.576	0.579	0.590
xlm-roberta-base	0.529	0.492	0.501	0.534	0.531	0.533
xlm-roberta-large	0.547	0.495	0.502	0.576	0.581	0.576
xlm-roberta-base-uk	0.528	0.491	0.501	0.535	0.533	0.535
ukr-roberta	0.580	0.559	0.570	0.572	0.570	0.582
paraphrase-multilingual-mpnet-base-v2	**0.735**	0.718	0.716	0.644	0.636	0.656

Table 3: Word Sense Disambiguation (WSD) Accuracy for Ukrainian language with Different Pooling Strategies and Pretrained Models without fine-tuning.

where a, p, and n are the embeddings of the anchor, positive, and negative sentences, respectively, and M is a margin hyperparameter that ensures that the positive example is at least closer to the anchor than the negative example. We used Euclidean distance as the distance metric in our experiments and set $M = 1$.

5.3 Training process

During the model's training, we monitored the performance of Word Sense Disambiguation accuracy on 20% of the SUM evaluation dataset to assess if it was being improved with the training process. We used 1% of a fine-tuning dataset to calculate training metrics and the rest 99% for training. We employed an early stopping mechanism based on the WSD accuracy on SUM based evaluation dataset. A batch size of 32 and the Adam optimizer with a learning rate of 2e-6 were used for the model optimization. Furthermore, we applied linear learning rate warm-up over the first 10% of the training data.

6 Results

The Table 4 presents the performance evaluation of our proposed method on the SUM evaluation dataset for homonyms.

We started with the Babelfy as a baseline, which was manually validated on 10% of the randomly sampled portion from the WSD evaluation dataset. Next, we tested a vanilla PMMBv2 model without fine-tuning, followed by a fine-tuned version of the PMMBv2 model using the proposed approach. The models fine-tuned by our approach outperform both Babelfy and vanilla PMMBv2 models. We observed that a larger dataset for fine-tuning led to better accuracy.

We assume that a model trained on a larger dataset, which also has a larger average distance between positive and negative examples, generates better homonym-specific embeddings. We also observed that the model PMMBv2 tuned on 1,2M triplets with filtering out pairs with a small difference (less than 0.3) between the cosine similarity of the anchor and positive examples and that of the anchor and negative examples, resulting in the best accuracy.

As the dataset used for training our model was constructed in an unsupervised manner, there existed a possibility of the model being biased towards the most frequently occurring senses of a given lemma. To assess this, we evaluated the model's accuracy based on the frequency of sense usage referring to the SUM dictionary (see Table 5). Our findings showed that the PMMBv2 model tuned on $\sim$1,2M triplets with filtering performed better for the less commonly occurring senses. Therefore, we can infer that the fine-tuned model not only considers the context but also makes predictions that are not solely based on the popularity of a sense.

We have evaluated our approach on the polysemy dataset to investigate the correlation between the performance of the model on homonyms and polysemous lemmas. The Table 6 shows the accuracy of the model on the polysemy dataset, where we have examined the model's ability to predict the first 2/3/all lexical meanings of each lemma. How-

Model	Overall acc.	Noun acc.	Verb acc.	Adj. acc.	Adv. acc.
Babelfy baseline	0.526	-	-	-	-
PMMBv2	0.735	0.767	0.668	0.752	0.593
PMMBv2 tuned on $\sim$190K triplets	0.77	0.819	0.685	0.743	0.562
PMMBv2 tuned on $\sim$1,2M triplets	0.778	**0.825**	**0.698**	**0.761**	0.531
PMMBv2 tuned on $\sim$1,2M triplets with filtering	**0.779**	0.824	0.693	0.759	**0.607**

Table 4: Accuracy on the WSD homonym evaluation dataset for Ukrainian Language using Babelfy, PMMBv2, and models fine-tuned by the proposed approach.

Frequency of sense usage	PMMBv2	PMMBv2 tuned on $\sim$1,2M triplets with filtering
1	0.76	0.799
2	0.703	0.754
3	0.666	0.773

Table 5: Accuracy on the WSD evaluation dataset for the Ukrainian Language based on the frequency of sense usage for the PMMBv2 baseline and fine-tuned version.

ever, we have observed a decrease in performance when evaluating the polysemy dataset, despite using better homonym-specific embeddings achieved through fine-tuning. We hypothesize that this may be due to the challenge of distinguishing between similar meanings for polysemous words (see Table 1). Furthermore, our observations indicate that the model PMMBv2, fine-tuned on 1,2M triplets with filtering out pairs, exhibits an even greater decrease in performance when applied to the polysemy dataset compared to PMMBv2 fine-tuned on 1,2M triplets without filtering.

7 Conclusion

Our research proposes a novel approach for solving the WSD task in under-resourced languages such as Ukrainian. We used a supervised approach to fine-tune LLMs on the unsupervised dataset generated by our method.

Furthermore, we built an evaluation dataset based on the SUM dictionary, which other researchers can use for evaluating the WSD task in the Ukrainian language.

We implemented the U-WSD framework during the research, which preprocess and generate evaluation and fine-tuning datasets, perform inference, and measure performance.

Our approach achieved 77.9% accuracy on the homonym dataset, surpassing graph-based methods such as Babelfy.

Future work aims to enhance the quality of the fine-tuning dataset by employing several measures. These measures include the removal of nearly identical anchor and positive examples, the exclusion of named entities detected as the target lemma, and the sampling of a more uniformly representative subset of examples for each lemma. We also want to improve the target lemma detection algorithm. Additionally, we plan to explore more advanced embedding comparison mechanisms beyond cosine similarity.

Limitations

The proposed approach has several limitations. Firstly, the approach is evaluated on a relatively small dataset of homonyms, which contains example from fiction, folklore, etc. Our dataset might not represent the entire Ukrainian language. Additionally, we focus only on homonymy, which may limit the approach's applicability to real-world scenarios where both homonymy and polysemy are present.

During our research on WSD, we discovered a lack of bias control in the SUM and UberText datasets. This deficiency presents a potential issue of such as gender, race, or socioeconomic status biases in our model.

Recreating the fine-tuning process requires a GPU with sufficient memory, such as the NVIDIA T4 GPU with 16 GB of memory on the AWS in-

Model	First 2 senses	First 3 senses	All senses
PMMBv2	0.682	0.637	0.608
PMMBv2 tuned on ∼190K triplets	**0.702**	**0.66**	**0.632**
PMMBv2 tuned on ∼1,2M triplets	0.7	0.656	0.629
PMMBv2 tuned on ∼1,2M triplets with filtering	0.689	0.646	0.618

Table 6: Accuracy on the WSD polysemy evaluation dataset for Ukrainian Language using, PMMBv2, and models fine-tuned by the proposed approach.

stance g4dn.xlarge.

To use the proposed approach for languages other than Ukrainian, a dictionary with lemmas and their lexical meanings, mechanisms to classify parts of speech, and a large dataset with sentences from various areas to cover lemmas with different meanings are needed.

Ethics Statement

Our objective is to increase the accessibility of NLP research by prioritizing under-resourced languages, with a particular focus on Ukrainian language research. Through the development of generalizable approaches, we hope to create solutions that can be applied to a variety of languages beyond Ukrainian. We are also mindful of the potential real-world impact of our research, and we strive to ensure that our work contributes to the advancement of society. Finally, we believe in the importance of engaging with the broader NLP community, particularly the global ACL community, to promote collaboration and knowledge-sharing.

References

Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. 2016. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Bmvc*, volume 1, page 3.

Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2021. Consec: Word sense disambiguation as continuous sense comprehension. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1492–1503.

Dmytro Chaplynskyi. 2023. Introducing UberText 2.0: a corpus of modern Ukrainian at scale. In *Proceedings of the Second Ukrainian Natural Language Processing Workshop*, pages 1–10, Dubrovnik, Croatia. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. 2014. Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. Glossbert: Bert for word sense disambiguation with gloss knowledge. *arXiv preprint arXiv:1908.07245*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

George A Miller, Claudia Leacock, Randee Tengi, and Ross T Bunker. 1993. A semantic concordance. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, 41(2):1–69.

Steven Neale, Luís Gomes, Eneko Agirre, Oier Lopez de Lacalle, and António Branco. 2016. Word sense-aware machine translation: Including senses as contextual features for improved translation models. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2777–2783.

Ukrainian Lingua-Information Fund of NAS of Ukraine (ULIF-NASU). 2010. Словник української мови [Dictionary of the Ukrainian language], volume 20 of Словники України [Dictionaries of Ukraine]. Наук. думка [Nauk. dumka], Kyiv.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 1532–1543.

Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. Semeval-2007 task-17: English lexical sample, srl and all words. In Proceedings of the fourth international workshop on semantic evaluations (SemEval-2007), pages 87–92.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. arXiv preprint arXiv:2003.07082.

Vitalii Radchenko. 2020. Youscan. https://youscan.io/blog/ukrainian-language-model/.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. OpenAI blog, 1(8):9.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084.

Neetu Sharma and Prof. S. Niranjan. 2015. Applications of word sense disambiguation: A historical perspective. INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH TECHNOLOGY (IJERT) NCETEMS-2015 (Volume 3-Issue 10).

Nakatani Shuyo. 2010. Language detection library for java. http://code.google.com/p/language-detection/.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In Proceedings of the AAAI conference on artificial intelligence, volume 31.

NAS of Ukraine Ukrainian Lingua-Information Fund. 2022. Ulif. https://en.ulif.org.ua/.

Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. Does bert make any sense? interpretable word sense disambiguation with contextualized embeddings. arXiv preprint arXiv:1909.10430.

A U-WSD framework

During our research, we implemented the framework to aid in working with models and evaluating their performance in the WSD task for the Ukrainian language, which is available at `https://github.com/YuriiLaba/U-WSD`. This framework consists of three main parts: (1) cleaning and generation of the SUM dataset, (2) embedding calculation and prediction running, and (3) performance metric evaluation.

The first part includes various dataset-cleaning techniques, such as filtering by the length of the lemma, selecting the first n senses or examples for each lemma, and more. Additionally, this part allows the generation of a dataset with lexical meanings for each lemma separately or grouping meanings at the homonym level.

The second part enables the selection of different models and pooling strategies for calculating embeddings for lexical meanings and examples. Finally, the third part generates a performance report based on the part of speech, lemma frequency which is obtained from the Ubertext dataset Chaplynskyi (2023), and different numbers of top n lexical senses of a lemma.