Word Sense Disambiguation with Large Language Models: Casing Bulgarian

Nikolay Paev, Kiril Simov, Petya Osenova
Artificial Intelligence and Language Technology
Institute of Information and Communication Technologies
Bulgarian Academy of Sciences
Bulgaria
nikolay.paev@iict.bas.bg, kivs@bultreebank.org, petya@bultreebank.org

Abstract

The paper presents two approaches to Word Sense Disambiguation in Bulgarian with the usage of Large Language Models (in our case - our own pre-trained BERT models). The knowledge from a Bulgarian WordNet (BTB-WordNet) has been used in the fine-tuning stages. As far as we know, the presented results are the SOTA in Word Sense Disambiguation task for Bulgarian. In addition, we experimented with various ways in dividing the data into training, development, and test datasets.

1 Introduction

The Word Sense Disambiguation (WSD) task is still an open NLP task, especially for the less-resourced and undertrained languages with respect to the semantic parsing and knowledge extraction. In this paper, we present an approach for WSD using Large Language Models (LLM). LLMs were fine-tuned on a dataset produced through compiled examples that are related to the WordNet synsets. We explored not only how well can LLMs be fine-tuned to the tasks of WSD, but also how well the knowledge learned by the model interacts with the knowledge encoded within the WordNet itself. The interaction between the two sources of knowledge was performed during the fine-tuning. It largely depends on the selection of elements in the training, development, and test datasets.

For example, let us consider the following Bulgarian sentence: "Нашата статия беше приета за конференцията Global WordNet в Италия." "Nashata statiya beshe prieta za konferenciyata Global WordNet v Italia." ("Our paper was accepted for the Global WordNet Conference in Italy.") The lemma 'статия' (statiya) is part of two synsets connected through a hyperonym relation: a) 'статия' (statiya) (as a paper, synset: oewn-06280609-n[1], gloss: a scholarly article describing the results of observations or stating hypotheses), and b) its hyponym 'статия' (statiya) (as an article, synset: oewn-06278749-n, gloss: nonfictional prose forming an independent part of a publication). Thus, when we annotate the occurrence of the word 'статия' (statiya) in the above sentence in the first sense, it is an exact annotation and thus classified as a positive example of a correct annotation. But using the second sense, which in the Bulgarian WordNet (which we use) is related to the same word 'статия' (statiya) as a negative example for the annotation in the same sentence, results in a false statement. Adding such examples within the dataset for WSD makes the dataset contain contradictory statements. LLMs very easily recognize such contradictions. Similar contradictions could also arise from more subtle interactions of relations within the WordNet. Thus, the selection of examples for the WSD dataset has to be done carefully.

For our experiments, Bulgarian BTB-WordNet (BGWN) is used (Simov and Osenova, 2023). The corpus of annotated examples of Bulgarian is composed of data selected to reflect the senses within BGWN. Each example contains at least one word form annotated with a sense of BGWN. Some examples might be related to more than one sense. Also, in some of the examples all non-functional words happen to be annotated. The number of different sentences is 86 755, the number of the annotations amounts to 111 272. The senses were assigned to 27 597 lemmas and these lemmas are part of 20 615 synsets. Thus, we consider the resulted corpus large enough for fine-tuning of LLMs with respect to Bulgarian WSD. The corpus allows us to perform experiments with different settings according to these characteristics.

[1]The Synset identifier was taken from the Open English Wordnet —- https://en-word.net/lemma/paper

To our knowledge, here we present the SOTA for Bulgarian in the WSD task.

The structure of the paper is as follows. In the next section, we present some focused related work. Section 3 describes the approaches that we implemented for fine-tuning of LLMs to the WSD task. The first is a binary classification of a context-gloss pair, where the context is the candidate word form within the text, and the gloss is the definition of the candidate sense. The second approach is the evaluation of all candidate senses with respect to their appropriateness for a given context. In Section 4 we report on the experiments with BGWN data. The last section concludes the paper and outlines some future directions of research.

2 Related Work

For the various types of WSD tasks, already LLMs started to be explored. For example, (Kibria et al., 2024) test four LLMs for English - ChatGPT-3.5, Mistral, Llama, Gemini Pro. The last one performed the best on the nine selected WSD datasets. However, no fine-tuning was performed in the experiments. The goal of the research reported in the paper is to evaluate the different LLMs with respect to their knowledge necessary for the task of WSD. The main difference with our work is that our aim is to train a WSD model using fine-tuning on our annotated corpus.

(Sumanathilaka et al., 2024) approach the improvement of WSD through the usage of LLMs and more precisely, of various prompt techniques. The authors suggest a method that combines a knowledge graph, a Part-of-Speech tagging and few-shot prompting as a guide to LLMs. Similarly to the above work this is not using fine-tuning. We plan to explore some of the proposed approaches in combination with the approach we present in this paper. Especially, combination with knowledge graphs, POS tagging, and Named Entity Recognition.

In (Bevilacqua et al., 2021) the recent trends in WSD have been described. Among other points, the authors note that 'different kinds of knowledge are orthogonal to each other and can be exploited in conjunction' and that adding more quality data improves the results. However, they also argue that at the moment the knowledge-based WSD is not so crucial as it was before due to the existence of many and multilingual pre-trained models.

Concerning WSD in other languages, in (Laba et al., 2023) a supervised fine-tuning of a pre-trained LLMs (mostly BERT-based) was employed on a dataset of Ukrainian, generated in an unsupervised way. The aim was to obtain better contextual embeddings for ambiguous words. We support authors' conclusion that 'WSD involves not just the knowledge of language but world knowledge and the capability of piecing together facts from multiple sources — in other words, functional competence.'

In our work, we started with a set-up very close to the one described in (Huang et al., 2019). Table 1 presents their construction methods for *context-gloss* pairs. In the first case, the target element (usually a word form) in the relevant context (usually a sentence) is coupled with all glosses for the target element. The glosses are taken from a Wordnet. For the context-gloss pairs, a binary classification has been performed (Yes/No) reflecting whether the gloss describes the correct sense for the target element in the context, or not. These pairs are called *Context-Gloss Pairs*. The specific examples of such pairs are given in the first part of the table. Additionally, the authors introduced a modification called *Context-Gloss Pairs with Weak Supervision*. In this case, the context part of the pair has been modified by highlighting the target element in quotation marks. The gloss part has also been modified by placing the lemma of the target element in front of the gloss. In Table 2 the prefix of the target element is highlighted by underlining. These modifications are shown in the second part of Table 1. Both types of context-gloss pairs have been constructed for all the senses related to the target element in the WordNet. Then these pairs were used for the fine-tuning of a BERT language model[2]. There are three fine-tuned models: (1) the model performs a classification of the target element in the context-gloss pair (the model is called **GlossBERT(Token-CLS)**); (2) the model performs a classification of [CLS] token in the context-gloss pair, and in this way the whole sentence is classified (the model is called **GlossBERT(Sent-CLS)**); and (3) the model performs a classification of [CLS] token in the context-gloss pair with weak supervision, and in this way the whole sentence is classified but with a stress on the target element (the model

[2]BERT model was introduced by Devlin et al. (2018)

Sentence with four targets:
Your research stopped when a convenient assertion could be made.

Context-Gloss Pairs of the target word [research]	Label	Sense Key
[CLS] Your research ... [SEP] systematic investigation to ... [SEP]	Yes	research%1:04:00::
[CLS] Your research ... [SEP] a search for knowledge [SEP]	No	research%1:09:00::
[CLS] Your research ... [SEP] inquire into [SEP]	No	research%2:31:00::
[CLS] Your research ... [SEP] attempt to find out in a ... [SEP]	No	research%2:32:00::
Context-Gloss Pairs with weak supervision of the target word [research]	Label	Sense Key
[CLS] Your "research" ... [SEP] research: systematic investigation to ... [SEP]	Yes	research%1:04:00::
[CLS] Your "research" ... [SEP] research: a search for knowledge [SEP]	No	research%1:09:00::
[CLS] Your "research" ... [SEP] research: inquire into [SEP]	No	research%2:31:00::
[CLS] Your "research" . . . [SEP] research: attempt to find out in a . . . [SEP]	No	research%2:32:00::

Table 1: The construction methods for training examples. The sentence is taken from SemEval-2007 WSD dataset. The ellipsis "..." indicates the remainder of the sentence or the gloss. The table is copied from (Huang et al., 2019), page 3510.

is called **GlossBERT(Sent-CLS-WS)**). The reported experiments show that the results improve from the first model to the last one (with some small exceptions).

In our own work, we start directly with the context-gloss pairs with weak supervision. Since in our case the context examples are directly included within the synsets of BGWN, the division of a context-gloss dataset in training, development, and test sets is not performed on the basis of the whole annotated texts, but on the individual context-gloss pairs. During the development and test evaluation phases, positive and negative cases were taken into account. We performed experiments by using different divisions of the context-gloss dataset in training, development, and test sets, and these divisions demonstrate different results, respectively.

Song et al. (2021) implemented an approach to semantically extend the sense representation in context-gloss pairs. This approach is called **Enhancing Sense Representations** (**ESR**). The idea is to add related words to the gloss of the synset. The motivation for this is the observed fact that the definitions within the synsets are usually short sentences, and thus do not provide enough information about the sense of the synset. This is especially true when the definitions were written for human understanding. The related words are constructed by first concatenating the words from the following three sources in order: (i) all the lemmas belonging to the synset (synonyms); (ii) WordNet example phrases or sentences of the synset; (iii) the hypernym gloss of the synset. All textual elements are concatenated and cleaned. The cleaning step is performed by deleting all the stop words and all the repeated words except the first occurrence. Another extension is on the context in the pair — the neighboring sentences of the context sentence are concatenated to it. In this way, the example consists of a larger context (more than one sentence) and a large sense description — the definition and related words. The paper demonstrates an improvement of the performance models trained over such extended context-gloss pairs.

In our work, we also enlarge the example, but only with lemmas from related synsets. We use more relations from WordNet, but we do not use the textual elements of the neighboring synsets.

In (Wahle et al., 2021) the authors propose some supervised methods that integrate WordNet knowledge for WSD in LM during the pre-training phase. In our case, we inject this knowledge into the fine-tuning phase. Interestingly, the authors also find that XLNet is more suitable for WSD than BERT. However, we used BERT in our current experiments.

3 LLM Word Sense Disambiguation Models

In this section, we present the models that were implemented in order to fine-tune LLMs to the task of Word Sense Disambiguation. We have performed experiments with three models - two binary classification models and one multiple choice model.

Pre-trained Language Models. For our experiments, we used our own pre-trained BERT models of two different sizes: 355M parameters (BERT-Large (**BERT-L**) with 24 layers and embedding dimension of 1024) and one with 657M parame-

Sentence with a single target:	
След приключването на курса има възможност за продължение ...	
... на работата с менторите, по същата схема на провеждане на занятията.	
Context-Gloss Pairs with weak supervision of the target word [продължение]	Label
[CLS] След приключването на курса има възможност за "продължение" ...	
... [SEP] продължение: Увеличаване на периода от време, в което нещо се ... [SEP]	**Yes**
[CLS] След приключването на курса има възможност за "продължение" ...	
... [SEP] продължение: Част от нещо (книга, пиеса, филм и други) ... [SEP]	**No**
Multiple choice disambiguation	
[CLS] След приключването на курса има възможност за продължение ...	
... [SEP] продължение: Увеличаване на периода от време, в което нещо се случва. ...	(1)
... [SEP] продължение: Част от нещо (книга, пиеса, филм и други) незавършено, ...	(2)
... [SEP] продължение: В спорта - допълнителното време, назначено за определяне ... [SEP]	(3)
Glosses with additional related lemmas (application of ESR)	
Увеличаване на периода от време, в което нещо се случва. ... [продължаване, продължение, ...]	
Част от нещо (книга, пиеса, филм и други) незавършено, ... [дял, клон, ...]	
В спорта - допълнителното време, назначено за определяне ... [времеви период, продължение, ...]	

Table 2: The construction methods for training examples in both setups. The first setup is similar to **GlossBERT**. The second setup presents multiple choice disambiguation where the embedding of the target word in the example is passed through a linear layer and compared with a dot product to the embeddings of the target word in each glosses passed through a separate linear layer. In this way the model assigns some probability to each candidate sense — (1), (2), or (3). The last part of the table demonstrates the addition of the list of lemmas by application of an ESR procedure.

ters (which we call BERT-ExtraLarge (**BERT-XL**) with 24 more layers — 48 in total). These BERT models were pre-trained on 20B Bulgarian tokens. Our pre-training dataset consists of mainly Web data, literature, administrative and scientific documents, as well as Wikipedia articles. The models were trained for 3 and 5 epochs, respectively, and a single epoch of pre-training took 23 and 60 hours on 16 Nvidia A100s. We plan to upload the models on Huggingface in the near future.

Fine-tuning Data Models. For our fine-tuning experiments with *Context-Gloss Pairs* we opted for the second setup (**GlossBERT(Sent-CLS)**) where the [CLS] token is passed through a classification layer. The model is trained on the binary classification task of predicting whether the gloss matches the target word. We explored the idea for *Context-Gloss Pairs with Weak Supervision* - the target word is enclosed in special tokens. Another modification which we studied is enriching the glosses with a list of lemmas related to the same sense of the gloss. (similar to **Enhancing Sense Representations** Song et al. (2021)). The lemmas are concatenated to the end of the gloss. We suppose that expanding the context can leverage the ability of the pre-trained model to recognize similar words.

We also explored a second setup — multiple choice disambiguation. We suppose that providing more glosses as options helps the model select the best one by excluding the others. The context contains the example and an arbitrary number of glosses prepended by the target word and separated with [SEP] tokens. The embedding of the first token of the target word in the example as well as the embedding of the first token of the target word in each gloss are passed through separate linear layers and then the dot product score is calculated between the results. The model is trained with *Cross Entropy Loss* to assign a high score to the correct glosses and low scores to the wrong ones. We considered two setups: (i) a list of glosses containing a single correct one, and (ii) a setup where a part of the inputs have no correct glosses, and in that case the model must assign the highest score to the last [SEP] token.

Examples of these setups are given in Table 2. The upper part of the table depicts two examples for binary classification — one positive (Yes) and one negative (No). The lower part of the table shows an example for the multiple choice classification, which contains three senses related to the same lemma.

Task 01:	Binary classification of context-gloss pairs with weak supervision			
	BERT Model	**ESR**	**Size Training**	**Accuracy**
Split: by examples				
01.	BERT-L	ESR -	88996	87.57
02.	BERT-L	ESR +	88996	89.22
03.	BERT-XL	ESR -	88996	89.57
04.	BERT-XL	ESR +	88996	**91.17**
Split: by synsets				
05.	BERT-L	ESR -	85254	72.77
06.	BERT-L	ESR +	85254	78.05
07.	BERT-XL	ESR -	85254	73.64
08.	BERT-XL	ESR +	85254	**81.88**
Task 02:	**Binary classification of context-gloss pairs with weak supervision Gloss with target word prefix: *target word:gloss***			
Split: by examples				
09.	BERT-L	ESR -	88996	87.97
10.	BERT-L	ESR +	88996	90.31
11.	BERT-XL	ESR -	88996	89.60
12.	BERT-XL	ESR +	88996	**91.83**
Split: by synsets				
13.	BERT-L	ESR -	84840	73.26
14.	BERT-L	ESR +	84840	81.62
15.	BERT-XL	ESR -	84840	77.53
16.	BERT-XL	ESR +	84840	**82.75**

Table 3: The results from the experiments with binary classification.

Selection of Datasets. For the binary classification task, both positive and negative context-gloss pairs are necessary. Consider the synset s. A positive pair is made by taking a single example e and pairing it with the definition def_s of s — (e, def_s). Negative pairs are made in a more complex way. Consider all lemmas related to the synset. Some of them are related to more than one synset. Let l be one such lemma related to both s and $\hat{s}$ and $s \neq \hat{s}$. Then the negative pair is taken to be $(e, def_{\hat{s}})$.

As mentioned previously in the introduction section, a problem with negative examples could arise if we use a definition from a synset that shares the same lemma as another synset whose definition is used for a positive example and the two synsets are semantically related. (If s and $\hat{s}$ are connected by the *hypernym* relation, for example.) In the sentence discussed above, the contradiction follows from the fact that the positive example claims that the sense for the target word is 'paper' as a scientific paper, while the negative example states that it is not the case, and that the same target word is an article being any type of non-fictional prose and thus forming an independent part of a publication (in Bulgarian the lemma is the same). Obviously, such a negative claim is not true for the target word. Thus, if the negations are selected completely random the resulting dataset could contains contradictions. Therefore, we have to control the selection of negative examples in order to escape from such contradictions.

Thus, the GlossBERT method as described above can lead to undesired data elements in the dataset for the fine-tuning step. The hyperonimy relation between the synsets is not the only relation in a wordnet that could couse introduction of such contradictory examples. Other relations in BGWN include also *causes, entails, mero_member, mero_part, mero_substance, instance_hypernym, similar, sem-derives-to, sem-derives-to-p, sem-derives-to-v, sem-derived-from-adj, sem-derived-from, sem-derives-to-adj, sem-derived-from-v*. In addition, the related synsets are not necessarily directly connected by some of these relations. Thus, we add one more constraint to the selection of the negative synset $\hat{s}$ — not only it must share a lemma with s but it must not belong to the transitive closure starting from the synset s over some of these relations: *causes, entails, hypernym, mero_member, mero_part, mero_substance, instance_hypernym, similar, sem-derives-to, sem-derives-to-p, sem-derives-to-v, sem-derived-from-adj, sem-derived-from, sem-derives-to-adj, sem-derived-from-v*.

In addition to ruling out possible contradictory negative examples, we require that each target word has exactly one positive pair and exactly one negative pair in the fine-tuning dataset. In this way, the classes are kept balanced. The total number of context-gloss pairs in the fine-tuning dataset is 111 272.

The dataset for the multiple choice is constructed in a similar manner. The senses that could lead to contradiction with the correct sense of the target word are not included in the list of potential senses. The resulting fine-tuning dataset in this case has 55 636 entries in total.

In order to perform a step of enriching the gloss similar to *Enhancing Sense Representations*, described above, we selected the related synsets following the relations mentioned above, because they implied synsets that follow logically from the target synset. In addition, we selected not only the immediately connected synsets, but also those that were on the transitive closure of the relations. We restricted the number of synsets from which we select lemmas to be up to 10.

Partitioning of the Datasets. The choice of training, validation, and test sets also matters for fine-tuning. Since most of the synsets have more than one example and we construct an example-gloss pair for each example, the same glosses are repeated across the datasets. This enables the model to overfitting with respect to the glosses. This means that the model becomes very well tuned to the glosses it observed, but it is not able to deal easily with a sense that it did not observe during the fine-tuning. In addition this overfitting adjusts the model to the features that were explicated within the examples. Therefore, such potential overfitting restricts the application of the model to the annotated examples that are related to the senses in the training set. If we consider using the model for disambiguation with the synsets of the same wordnet, the overfitting to the glosses does not cause big problems, and thus it can be ignored. In this case, a simple data split over the examples in the dataset is good enough. We denote this split as *split by examples*.

However, it is clear that the disambiguation model should work well not only over the senses within the training set, but also over new, unseen glosses. For such cases the overfitting should be stopped beforehand. Thus, we use a validation set that does not contain the same glosses for early stopping. We achieve this by splitting the dataset in such a way that examples from the same synset cannot be part of different partitions of the fine-tuning dataset. We call such a division of the fine-tuning dataset a *split by synsets*. The choice for a negative synset is also restricted to the synsets of the same set, which makes the context-gloss pairs totally independent. The new restrictions slightly lower the size of the fine-tuning dataset — it totals to 92 958 for the pairs, and to 46 479 for the multiple choice set.

Our intuition behind such a split is that fine-tuning with it forces the model to generalize over the context-gloss pairs in a better way. Thus, it makes the inference during the exploitation of the model more independent from the actual glosses the model observed during the fine-tuning step. Making the validation set independent leads to an earlier increase in the validation loss during training. This way the training process can be stopped earlier avoiding potential overfitting over the glosses and the resulting model should be better suited for disambiguation over new glosses.

4 Experiments and Results

In this section, we report our findings from several experiments that follow the approaches described in the previous section.

We fine-tuned the models on 8 Nvidia A100s. The training was the same for both – the binary classification of the example-gloss pairs, and the multiple choice disambiguation. We trained for 3 epochs with a learning rate of 2e-05 with linear decay and batch size of 32x8 and 8x8 for the two tasks, respectively. The validation loss was calculated on every 100 steps (out of roughly 1000 for the 3 epochs) and the best model was chosen. The best model performance was usually achieved midway through training, with slightly increasing validation loss thereafter.

We performed Binary classification experiments with both pre-trained models - **BERT-L** and **BERT-XL**. The results are given in Table 3. We organized the experiments in two tasks: **Task 1** includes experiments for a binary classification of context-gloss pairs with weak supervision. In this task, we do not add the target word in front of the gloss; **Task 2** is similar, but the target word is added to the gloss as a prefix. For both tasks, the experiments use the two splits of the pairs — "split by examples" and "split by synsets". Furthermore,

Task 03:	**Multiple choice disambiguation**			
	BERT Model	**ESR**	**Size Training**	**Accuracy**
Split: by examples				
17.	BERT-XL	ESR -	44556	83.82
18.	BERT-XL	ESR +	44556	**89.78**
Split: by synsets				
19.	BERT-XL	ESR -	42627	72.09
20.	BERT-XL	ESR +	42627	**86.82**
Task 04:	**Multiple choice disambiguation** **Gloss with target word prefix:** ***target word:gloss***			
Split: by examples				
21.	BERT-XL	ESR -	44336	85.03
22.	BERT-XL	ESR +	44336	**89.96**
Split: by synsets				
23.	BERT-XL	ESR -	42420	74.14
24.	BERT-XL	ESR +	42420	**89.24**
Task 05:	**Multiple choice disambiguation using binary classification model** **Gloss with target word prefix:** ***target word:gloss***			
Split: by examples				
21.	BERT-XL	ESR -	44336	75.47
22.	BERT-XL	ESR +	44336	81.52
Split: by synsets				
23.	BERT-XL	ESR -	42420	83.07
24.	BERT-XL	ESR +	42420	87.71

Table 4: The results from the experiments with multiple choice disambiguation.

the experiments were performed over pairs with added lemmas from related synsets (ESR column). The results show that the exploration of a larger model produces better results. This is evident from the comparison of experiments 01, 02, 05, 06, 09, 10, 13, 14 with experiments 03, 04, 07, 08, 11, 12, 15, 16 respectively. This observation is similar to the results in other NLP tasks. Thus, for the other setups we use only the larger pre-trained model (BERT-XL).

The addition of lemmas from related synsets also improves the results. Here, the observation is that the added value from the application of ESR is much higher in the cases of "split by synsets" than "split by examples". In our view, this is due to the overfitting in the case of "split by examples". In the other case, the impact of the model generalization over the glosses and the related information is much bigger. The last improvement of the results arises from the addition of the target word to the glosses in the context-gloss pairs. This is visible by comparing the experiments in Task 01 and Task 02.

The results from our second setup — multiple choice disambiguation — are given in Table 4. The results here are parallel to the ones reported above for the binary classification ones. Thus, all the methods for expanding the semantic content in the senses work in the same way. The main difference is the result produced by the multiple choice disambiguation — distribution over the glosses that are candidates for the sense of the target word. In our view, this approach could be useful for many more applications than the binary classification approach. One such application is, for example, finding new senses for a given lemma.

The results from both setups are not comparable. The binary classification accuracy shows how confident the model is that the word in the example is used with a specific sense. If we would like to use the binary classification for disambiguation, the model should be inferred with all possible senses of the target word separately, and then the senses can be ranked according to the confidence of the model predicting label Yes. The multiple choice disambiguation model does that process in a single step. Thus, a more suitable comparison of the binary classification model to the multiple choice model should be the result from the above-mentioned setup, which is given in Table 4 as Task 05. The data show that the multiple choice model

performs better for disambiguation overall.

5 Conclusion and Future Work

In this paper, we describe several LLM-based models for the WSD task in Bulgarian. We provide two setups for solving the problem. The two approaches complement each other. The binary classification setup provides a better result but requires several applications in order for the best solution to be found. It is not easy to recognize the cases where there is a gap in the lexical resource with respect to senses. The multiple choice approach solves the problem at once. It is also relatively easy to add an option for the missing senses. Last but not least, among our models there are such that are a SOTA in WSD for Bulgarian.

We also demonstrate that the proper selection of training, development, and test sets from a dataset of all context-gloss pairs is important for the quality and behavior of the model. Then the application of the models will depend on the task that we would like to solve. We recognize these differences because we started with examples for the senses within BGWN. We think that such problems arise also when it comes to whole texts annotated with senses.

One direction of future work is to improve the knowledge resource that we use – BGWN. It can be improved through the incorporation of more and diverse semantic information within the synsets. This might be done, for example, by improving the informativeness of the definitions by requiring a more complex structure for them, where various characteristics of the lemma meanings are made explicit. Also, through the addition of more examples and relations among the synsets. To sum up, more information has to be added in the direction of a more dense and versatile hierarchy, as well as in the direction of definitions and related examples.

Acknowledgments

The reported work has been supported by CLaDA-BG, *the Bulgarian National Interdisciplinary Research e-Infrastructure for Resources and Technologies in favor of the Bulgarian Language and Cultural Heritage, part of the EU infrastructures CLARIN and DARIAH*. We also acknowledge the provided access to *the e-infrastructure of the Centre for Advanced Computing and Data Processing* (the Grant No BG05M2OP001-1.001-0003).

References

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. Recent Trends in Word Sense Disambiguation: A Survey. In *International Joint Conference on Artificial Intelligence*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. GlossBERT: BERT for Word Sense Disambiguation with Gloss Knowledge. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3509–3514. ACL.

Raihan Kibria, Sheikh Intiser Uddin Dipta, and Muhammad Abdullah Adnan. 2024. On Functional Competence of LLMs for Linguistic Disambiguation. In *Proceedings of the 28th Conference on Computational Natural Language Learning*, pages 143–160, Miami, FL, USA. Association for Computational Linguistics.

Yurii Laba, Volodymyr Mudryi, Dmytro Chaplynskyi, Mariana Romanyshyn, and Oles Dobosevych. 2023. Contextual Embeddings for Ukrainian: A Large Language Model Approach to Word Sense Disambiguation. In *Proceedings of the Second Ukrainian Natural Language Processing Workshop (UNLP)*, pages 11–19, Dubrovnik, Croatia. Association for Computational Linguistics.

Kiril Simov and Petya Osenova. 2023. Recent Developments in BTB-WordNet. In *Proceedings of the 12th Global Wordnet Conference*, pages 220–227, University of the Basque Country, Donostia - San Sebastian, Basque Country. Global Wordnet Association.

Yang Song, Xin Cai Ong, Hwee Tou Ng, and Qian Lin. 2021. Improved Word Sense Disambiguation with Enhanced Sense Representations. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4311–4320. ACL.

Deshan Sumanathilaka, Nicholas Micallef, and Julian Hough. 2024. Assessing GPT's Potential for Word Sense Disambiguation: A Quantitative Evaluation on Prompt Engineering Techniques. *2024 IEEE 15th Control and System Graduate Research Colloquium (ICSGRC)*, pages 204–209.

Jan Philip Wahle, Terry Ruas, Norman Meuschke, and Bela Gipp. 2021. Incorporating Word Sense Disambiguation in Neural Language Models. *ArXiv*, abs/2106.07967.