DeepMistake at LSCDiscovery: Can a Multilingual Word-in-Context Model Replace Human Annotators?

Daniil Homskiy[▽] **Nikolay Arefyev**[◇,▽,△]

[▽]Lomonosov Moscow State University / Moscow, Russia
[△]National Research University Higher School of Economics / Moscow, Russia
[◇]Samsung Research Center Russia / Moscow, Russia
homdanil123@gmail.com, nick.arefyev@gmail.com

Abstract

In this paper we describe our solution of the LSCDiscovery shared task on Lexical Semantic Change Discovery (LSCD) in Spanish (D. Zamora-Reina et al., 2022). Our solution employs a Word-in-Context (WiC) model, which is trained to determine if a particular word has the same meaning in two given contexts. We basically try to replicate the annotation of the dataset for the shared task, but replacing human annotators with a neural network. In the graded change discovery subtask, our solution has achieved the 2nd best result. In the main binary change detection subtask, our F1-score is 0.655 compared to 0.716 of the best submission, corresponding to the 5th place. However, in the optional sense gain detection subtask we have outperformed all other participants.[1]

During the post-evaluation experiments we compared different ways to prepare WiC data in Spanish and fine-tune our model. We have found that it helps leaving only examples annotated as 1 (unrelated senses) and 4 (identical senses) rather than using 2x more examples including intermediate annotations. Generating additional examples from a WSD dataset also significantly improves the results.

1 Introduction

Given a list of words, a Lexical Semantic Change Detection (LSCD) system applied to diachronic corpora shall determine how these words change their meaning over time. The LSCDiscovery (D. Zamora-Reina et al., 2022) shared task on LSCD in Spanish consists of two main subtasks and a few optional ones. In the graded change discovery subtask, the participants were asked to rank 4385 words according to the degree of their change. In the binary change detection subtask, it was necessary to develop a binary classifier that finds among 60 given words those that have either lost some old senses, or obtained some new ones. Two optional binary subtasks required separately finding words with lost senses and words with new senses.

In order to annotate the test set for the shared task, for each word from the test set some examples were sampled from the old and the new corpus. Then human annotators were asked to annotate pairs of examples with scores from 1 to 4 according to the similarity of two occurrences of the same word by meaning. This kind of annotation is very similar to the Word-in-Context (WiC) task, which asks a model to determine if two occurrences of the same word have the same or different meaning.

2 Background

2.1 The Word-in-Context model

In order to solve the LSCD task, we address the Words-in-Context (WiC) task first. The WiC task is a simplified version of the Word Sense Disambiguation (WSD) task that can be reduced to binary classification. Each example in WiC consists of two occurrences of the same usually polysemous target word **w** (probably, in different grammatical forms) in two different contexts. The task is to determine if the target word has the same or different senses in two contexts. In our work we employ the Multilingual and Cross-Lingual Word-in-Context (MCL-WiC) dataset from SemEval-2021 Task2 (Martelli et al., 2021). Table 1 shows some statistics for this dataset.

We employ the WiC model proposed in (Davletov et al., 2021). In this model, the encoder from XLM-R (Conneau et al., 2020) is used to vectorize input examples. XLM-R is a Transformer-based neural network pre-trained as a masked language model (MLM) on about 2TB of texts in 100 languages. This not only makes our WiC model multilingual, but also enables zero-shot cross-lingual

[1]The code is available: https://github.com/Daniil153/DM-in-Spanish-LSCDiscovery

Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change, pages 173 - 179

trasferability, i.e. after training on the MCL-WiC dataset it can be applied even to those languages that are not present in this dataset (for instance, Spanish).

The architecture of the WiC model is the following. Two input sentences are concatenated and fed into XLM-R in the following format:

```
<s>sentence1</s>sentence2</s>
```

For each sentence, the outputs of XLM-R on all subwords of the target word are averaged (mean pooling). This results in two embeddings for two occurrences of the target word. Then these two embeddings are combined and fed into the binary classification head (see details below).

2.2 The RuShiftEval-2021 shared task

Our solution for the graded change discovery subtask was initially developed during the RuShiftEval-2021 shared task on LSCD for the Russian language (Kutuzov and Pivovarova, 2021), where it was the second best system during the competition and outperformed the best system in the post-competition experiments (Arefyev et al., 2021). However, in this shared task Spearman's correlation with the gold COMPARE scores (Schlechtweg et al., 2018) was the only metric for evaluation unlike the LSCDiscovery shared task, which offers more diverse metrics and several subtasks.

The best results in RuShiftEval-2021 were achieved with the following hyperparameters and design choices. To combine the embeddings of two occurrences of the target word, the L1-distance between the normalized embeddings and the dot product between the normalized embeddings are concatenated ($(\|\bar{x} - \bar{y}\|_1, \langle \bar{x}, \bar{y} \rangle)$). After batch normalization, this representation is fed into a linear classification head. All the weights of the network are fine-tuned with the cross-entropy loss. Two-step fine-tuning procedure consists of fine-tuning on examples in 6 languages from the training and the development sets of the MCL-WiC dataset, and then fine-tuning on examples in Russian from the RuSemShift (Rodina and Kutuzov, 2020) dataset, which served as the training and the development set in RuShiftEval-2021.

3 WiC-based LSCD

3.1 WiC training

To solve the Spanish LSCD task we used the WiC model with the architecture and hyperparameters described in 2.2 that have previously shown the best results. Additionally, we fine-tuned the model on the following data in Spanish (see table 1 for statistics).

Subset/language	size	#words	Avg. len.
MCL-WiC			
en-en	8008	3728	48
ru-ru	708	352	41
fr-fr	708	352	46
ar-ar	708	354	45
zh-zh	708	342	-
en-nen*	32	16	51
RuSemShift			
ru-ru	3898	70	51
DWUG_es			
es-es$^{bin1}_{COMP}$	4831	15	167
es-es$^{bin2}_{COMP}$	2638	15	165
es-es$^{bin1}_{ALL}$	9465	15	168
es-es$^{bin2}_{ALL}$	5443	15	167
es-es$^{bin1}_{COMP}$ (valid)	1376	5	155
Spanish XL-WSD			
es-es	8260	310	98

Table 1: Training and development data for our WiC model. L1-L2 means that the first sentence in each pair is in language L1, while the second sentence is in L2. en-nen* are en-ru, en-ar, en-fr, en-zh cross-lingual examples.

DWUG_es is the development set from the shared task. In the previous experiments binarizing human annotations and training the WiC model as a binary classifier has shown better results than training it as a regression model. Thus, we try two binarization methods. In the first method (**bin1**), the examples with annotations of 3 or 4 are treated as positive examples, and those with annotations of 1 or 2 as negative. In the second method (**bin2**), the examples with annotations of 2 or 3 were filtered out first, and the rest were treated as before.

Also, we have created the **COMP** version of the training set containing only COMPARE pairs (with the first sentence from the old corpus and the second from the new corpus), and the **ALL** version containing all pairs of sentences. We have separated all COMPARE pairs for 5 out of 20 words and used them as a validation set for early stopping during fine-tuning of the WiC model.

XL-WSD (Pasini et al., 2021) is a WSD dataset in 18 languages. We used only the development and the test subsets in Spanish to create additional training data for the WiC model. After generating all pairs of word occurrences with the same word lemma, the pairs of word occurrences having the same sense label were labeled as positive pairs,

while the pairs of occurrences with different sense labels were labeled as negative ones.

The WiC model was initialized with the standard XLM-R weights from MLM pre-training. Then we fine-tuned the model for the WiC task in one, two or three steps.

MCL→RSS. This is the best performing model from (Arefyev et al., 2021), which outperformed the winning solution of the RuShiftEval-2021 shared task in the post-evaluation period. This model was fine-tuned on multilingual MCL-WiC data, and then on RuSemShift data in Russian.

MCL→RSS→DWUG_es. The previous model was additionally fine-tuned on Spanish DWUG to improve the quality for Spanish.

MCL→DWUG_es. We hypothesised that fine-tuning on examples in Russian may hurt the performance for Spanish, thus, excluded this intermediate fine-tuning step from the previous fine-tuning scheme.

MCL→DWUG_es+XL-WSD. Finally, we decided to add the examples from XL-WSD in Spanish to the examples from DWUG_es to fine-tune on as many examples in Spanish as possible.

MCL→RSS→DWUG_es+XL-WSD. Our best model from RuShiftEval-2021 fine-tuned on all examples in Spanish we had.

MCL+RSS+DWUG_es+XL-WSD. We hypothesised that fine-tuning the model in many steps may result in forgetting information from the earlier steps. Thus, we try fine-tuning on all WiC data together in a single step.

3.2 Average Pairwise Distance (APD)

3.2.1 Graded change subtasks

For each target word, we retrieved 100 examples (or all examples, if there were fewer than 100) from the old and the modern corpora provided by the organizers. To find the positions of the target words, we used the lemmatizer from Spacy version 3.1.1 with the Spanish model es_core_news_md[2]. Next we created 100 (or fewer) COMPARE pairs of sentences. In Appendix A we study how the results depend on this number of pairs.

The pairs of sentences are scored by the WiC model. For each pair, the predicted probability of the negative class, i.e. the probability of two occurrences having different senses, is taken from the model. To estimate the graded change, for each target word we average these probabilities for the pairs of sentences containing this target word. The predicted probabilities may violate some metric axioms, hence, they are not distances in the mathematical sense. Nevertheless, we will use the traditional term Average Pairwise Distance (**APD**) (Giulianelli et al., 2020) to denote our final word scores. For the optional COMPARE subtask we used the same scores.

3.2.2 Binary subtasks

To solve the binary subtasks, we use only the examples provided by the organizers for 60 words from the test set. There are 20 old and 20 new examples for each word, let us call them the gold examples. Some pairs consisting of these examples were annotated by humans, and based on these annotations the gold labels were calculated while creating the test set. Thus, using these examples instead of the randomly sampled ones shall improve the chances to correctly predict the gold labels. However, it is likely that some rare new or lost senses are not among those 40 examples provided by the organizers. In real applications sampling more examples will likely be beneficial.

We generate all possible COMPARE pairs of the gold examples and calculate APDs for them. To produce binary predictions, we apply APD thresholding (**APD-t**). The threshold was selected to maximize the F1-score on the development set. The same predictions are used for the binary change, sense loss and sense gain detection subtasks.

3.3 Correlation Clustering (CC)

Since the gold COMPARE score for each word is calculated by averaging human judgements about the similarity of word occurrences taken from different time periods, our APD scores shall correlate well with the negated gold COMPARE scores if our WiC model approximates human judgements reasonably well. However, it is not obvious if they also correlate well with the Jensen-Shannon Distance (JSD) between the inferred sense distributions, which is the main metric in the graded change discovery subtask. Also if a word obtains or loses a rare sense while preserving the most frequent sense, the average distance between old and new examples shall be small and the APD-t method will fail do detect the change.

To address these issues, we try to cluster word uses the same way they were clustered by the organizers while creating the test set, but employing

[2]https://github.com/explosion/spacy-models/releases/tag/es_core_news_md-3.2.0

Method/Team	JSD, SPR	COMP, SPR
Baselines		
baseline1	**0.543** (4)	**0.561**
baseline2	0.092 (8)	0.088
Best results of other teams		
myrachins	**<u>0.735</u>** (1)	**<u>0.842</u>**
aishein	0.553 (3)	0.558
Our submissions: team *DeepMistake*, APD		
MCL→RSS	0.701 (2*)	**0.829**
MCL→RSS→ DWUG_es$^{bin1}_{ALL}$	**0.702** (2)	**0.829**
$^{\#}$MCL→DWUG_es$^{bin1}_{ALL}$	0.650 (2*)	0.787

Table 2: The results of the graded change discovery models. The best result within each block is in **bold**, the best result overall is also **<u>underlined</u>**. * indicates the potential ranks of the corresponding results in the leaderboard if they would have been submitted instead of our best submission. $^{\#}$ indicates buggy submissions (incorrect indices of the target words).

Method/Team	JSD, SPR	COMP, SPR
DWUG_es convertion comparison, APD		
MCL→DWUG_es$^{bin1}_{ALL}$	0.660 (2*)	0.800
MCL→DWUG_es$^{bin2}_{ALL}$	**0.672** (2*)	**0.820**
MCL→DWUG_es$^{bin1}_{COMP}$	0.650 (2*)	0.800
MCL→DWUG_es$^{bin2}_{COMP}$	0.669 (2*)	0.815
WiC fine-tuning schemes, APD		
MCL	0.648 (2*)	0.791
MCL→ DWUG_es$^{bin2}_{ALL}$+XL-WSD	0.712 (2*)	0.854
MCL→RSS→ DWUG_es$^{bin2}_{ALL}$+XL-WSD	0.711 (2*)	**<u>0.855</u>**
MCL+RSS+ DWUG_es$^{bin2}_{ALL}$+XL-WSD	**<u>0.719</u>** (2*)	0.838
CC		
MCL→ DWUG_es$^{bin2}_{ALL}$+XL-WSD	0.650 (2*)	0.748
Gold scores		
COMPARE scores	0.920	1.0
JSD scores	1.0	0.920

Table 3: Post-evaluation experiments with the graded change detection models on the gold examples for 60 test words. * indicates the potential ranks of the corresponding results.

annotations from our WiC model instead of human annotations. We generate all possible pairs of the gold examples and score them with the WiC model. Unlike the APD method which relies on the distances between examples from different corpora only, clustering-based methods can benefit from the distances between examples from the same corpus as well.

We use the implementation of Correlation Clustering (**CC**) by Schlechtweg et al. (2021), which presumably was also used to create the test set.[3] This time we employ the binary predictions of the WiC model instead of the predicted probabilities, and treat positive predictions (same sense) as positive edges and negative predictions as negative edges.[4] After clustering, the aforementioned code calculates both the JSD and the COMPARE scores, and also all predictions for the binary subtasks.

3.4 Computational complexity

In order to solve the graded change discovery subtask, it was necessary to calculate scores for 4385 words. The WiC model processed about 388K pairs of sentences in total, or 89 pairs per word on average. This took about 3 hours on one V100 GPU. Additionally, about 7 hours of CPU time was spent to lemmatize both corpora. The calculation of APDs took insignificantly small time.

For the graded change subtasks, we experimented with correlation clustering only after the competition and processed only 60 words from the test set. This took about 18 hours of CPU time.

4 Results

4.1 Graded subtask

Table 2 shows the results for the graded change discovery subtask. Our best submission has shown 2nd best result according to both metrics. The model from RuShiftEval-2021 further fine-tuned on the Spanish development set has shown the best result among our submissions. However, further fine-tuning has brought very small benefits. This is likely due to suboptimal binarization of the Spanish data.

During the post-evaluation experiments, we have studied how the results depend on the training data. The results in table 3 clearly indicate that leaving only annotations of 1 and 4 (bin2) consistently improve performance despite almost 2x reduction in the number of training examples in Spanish. Using ALL pairs gives 2x increase in the number of examples, but only marginal improvement in the performance. This is probably because we use the model to score COMPARE pairs only. Adding examples generated from the Spanish part of XL-WSD gives significant boost. This may be due to training on

[3] https://github.com/Garrafao/WUGs

[4] The negative and positive predictions were converted to the annotations of 1 and 2 respectively. We changed only the arguments specifying the annotation range (min=1, max=2) and the binarization threshold=1.5. The default values for other hyperparameters were used: lowerrangemin=1, lowerrangemax=3, upperrangemin=3, upperrangemax=5, lowerprob=0.01, upperprob=0.1

	Binary change			Sense gain			Sense loss		
Method/Team	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**
Baselines									
baseline1	**0.537** (9)	**0.846**	**0.393**	-	-	-	-	-	-
baseline2	0.222 (10)	0.500	0.143	0.211 (7)	0.400	0.143	0.0 (6)	0.0	0.0
Best results of other teams									
myrachins	**0.716** (1)	**0.615**	0.857	**0.491** (3)	0.333	0.927	**0.688** (1)	**0.564**	**0.880**
dteodore	0.709 (2)	0.549	**1.0**	0.0 (8)	0.0	0.0	0.0 (6)	0.0	0.0
rombek	0.687 (3)	0.590	0.821	0.490 (4)	0.343	0.857	0.593 (3)	0.552	0.640
kudisov	0.658 (4)	0.510	0.929	0.520 (2)	**0.361**	**0.929**	0.600 (2)	0.514	0.720
Our submissions: team *DeepMistake*									
$^{\#}$MCL→ DWUG_es$^{bin1}_{ALL}$ + XL-WSD (CC)	0.420 (10*)	**0.800**	0.290	0.417 (6*)	**0.500**	0.360	0.280 (6*)	**1.0**	0.160
MCL→ DWUG_es$^{bin1}_{ALL}$ + XL-WSD (APD-t)	**0.655** (5)	0.633	**0.679**	**0.591** (1)	0.433	**0.929**	**0.582** (4)	0.533	**0.640**
Post-evaluation results for APD-t									
MCL→DWUG_es$^{bin1}_{ALL}$	0.706 (3*)	0.600	0.860	0.520 (1*)	0.350	**1.0**	0.650 (2*)	**0.530**	0.840
MCL→DWUG_es$^{bin2}_{ALL}$	0.680 (4*)	0.560	0.860	0.490 (3*)	0.330	**1.0**	0.620 (2*)	0.490	0.840
MCL→DWUG_es$^{bin1}_{COMP}$	0.640 (6*)	**0.610**	0.680	**0.580** (1*)	**0.420**	0.930	0.570 (4*)	0.520	0.640
MCL→DWUG_es$^{bin2}_{COMP}$	0.695 (3*)	0.590	0.860	0.510 (2*)	0.340	**1.0**	0.640 (2*)	0.510	0.840
MCL→ DWUG_es$^{bin2}_{ALL}$ + XL-WSD	**0.712** (2*)	0.580	**0.930**	0.480 (4*)	0.310	**1.0**	**0.660** (2*)	0.510	**0.920**
Post-evaluation results for CC									
MCL→ DWUG_es$^{bin2}_{ALL}$ + XL-WSD	0.693 (3*)	0.553	0.929	0.462 (4*)	0.316	0.857	0.528 (4*)	0.500	0.560

Table 4: The results for the binary subtasks. * indicates the potential ranks of the corresponding results in the leaderboard if they would have been submitted instead of our best submission. $^{\#}$ indicates buggy submissions (CC incorrectly executed).

2.5x more examples, but also 22x more different target words. Fine-tuning on all datasets in one step improves Spearman's correlation with the JSD scores a bit, but not with the COMPARE scores. Comparing multi-step and single-step fine-tuning is an interesting direction for the future work.

The CC method works worse than APD, a thorough analysis is required to understand the reasons. Also we notice that the gold COMPARE scores have Spearman's correlation with the gold JSD scores of 0.92. This means that the limits of the APD method are not achieved yet, and further improvement of the WiC model for better reproduction of human annotations is a reasonable way to improve the results.

4.2 Binary subtask

Table 4 shows the results for the binary subtasks. Our model has outperformed all other participants in the optional sense gain detection subtask. However, the F1-score for the main binary change detection subtask is 6% below the best result. During the post-evaluation experiments we have changed the binarization to bin2, and also set the natural threshold of 0.5, which improved the results for binary change and sense loss detection to the level comparable with 2nd best result in the leaderboard. The APD-t method works better than CC, even though it reuses the same predictions for all binary subtasks.

5 Conclusion

This paper makes the first step towards answering the question in its title: can a multilingual word-in-context model replace human annotators for solving the LSCD task? For now, it seems that our word-in-context model is not good enough to do that. However, we have shown that experimenting with the training data is a promising direction to achieve this goal.

Acknowledgements

We are grateful to our anonymous reviewers. This research was partially supported by the Basic Research Program at the HSE University and through computational resources of HPC facilities at HSE University (Kostenetskiy et al., 2021).

References

N. Arefyev, M. Fedoseev, V. Protasov, D. Homskiy, A. Davletov, and A. Panchenko. 2021. Deepmistake: Which senses are hard to distinguish for a word-in-context model. In *Computational linguistics and intellectual technologies*, 20, page 16 – 30, Russian Federation.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Frank D. Zamora-Reina, Felipe Bravo-Marquez, and Dominik Schlechtweg. 2022. Lscdiscovery: A shared task on semantic change discovery and detection in spanish. In *Proceedings of the 3rd International Workshop on Computational Approaches to Historical Language Change*, Dublin, Ireland. Association for Computational Linguistics.

Adis Davletov, Denis Gordeev, Nikolay Arefyev, and Emil Davletov. 2021. LIORI at SemEval-2021 task 8: Ask transformer for measurements. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 1249–1254, Online. Association for Computational Linguistics.

Mario Giulianelli, Marco Del Tredici, and Raquel Fernández. 2020. Analysing lexical semantic change with contextualised word representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3960–3973, Online. Association for Computational Linguistics.

P. S. Kostenetskiy, R. A. Chulkevich, and V. I. Kozyrev. 2021. HPC resources of the higher school of economics. *Journal of Physics: Conference Series*, 1740:012050.

Andrey Kutuzov and Lidia Pivovarova. 2021. Rushifteval: A shared task on semantic shift detection for russian. In *Computational linguistics and intellectual technologies*, 20, Russian Federation.

Federico Martelli, Najla Kalach, Gabriele Tola, and Roberto Navigli. 2021. Semeval-2021 task 2: Multilingual and cross-lingual word-in-context disambiguation (mcl-wic). In *SEMEVAL*.

Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. XL-WSD: An extra-large and cross-lingual evaluation framework for word sense disambiguation. In *Proc. of AAAI*.

Julia Rodina and Andrey Kutuzov. 2020. RuSemShift: a dataset of historical lexical semantic change in Russian. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1037–1047, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Dominik Schlechtweg, Sabine Schulte im Walde, and S. Eckmann. 2018. Diachronic usage relatedness (durel): A framework for the annotation of lexical semantic change. *ArXiv*, abs/1804.06517.

Dominik Schlechtweg, Nina Tahmasebi, Simon Hengchen, Haim Dubossarsky, and Barbara McGillivray. 2021. DWUG: A large Resource of Diachronic Word Usage Graphs in Four Languages. *CoRR*, abs/2104.08540.

A Graded change detection results depending on the number of pairs sampled

In the post-evaluation phase, we measured the performance of the model in the graded change detection subtask depending on how many pairs of sentences are sampled. For this experiment, we sampled 1000 sentences with replacement from each corpora, built 1000 COMPARE pairs and annotated them with the WiC model. Then for each number of pairs we sampled this number of pairs 100 times, and calculated the APD scores and the target metrics. Finally, we calculated the mean and the standard deviation of the target metrics for each number of pairs.

We compare these results to the results on the gold COMPARE pairs, i.e. annotating with our WiC model the same pairs that were annotated by humans. There are 278 unique pairs per word on average. Also we compare to using all COMPARE pairs consisting of gold examples only. There are 400 such pairs per word consisting of 20 old and 20 new examples.

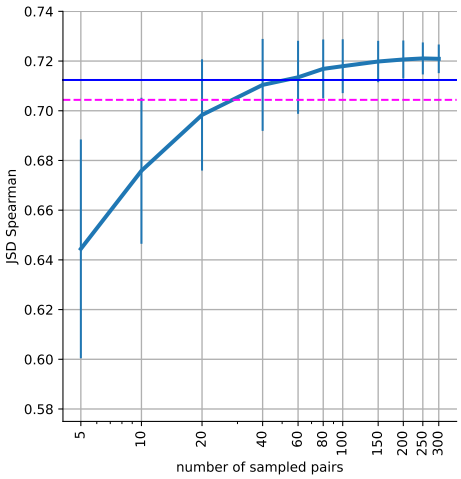

Figure 1: Spearman's correlation of our APD scores with the gold JSD scores depending on the number of COMPARE pairs sampled per word. Model: MCL→DWUG_es$^{bin2}_{ALL}$+XL-WSD. The solid blue horizontal line corresponds to all COMPARE pairs of the gold examples. The dashed purple horizontal line corresponds to the gold COMPARE pairs. Error bars show one standard deviation.

From figures 1, 2 we can conclude that after 100-150 pairs of sentences sampled per word the average quality stops increasing, only the standard deviation decreases slowly.

Interestingly, when the number of pairs is large

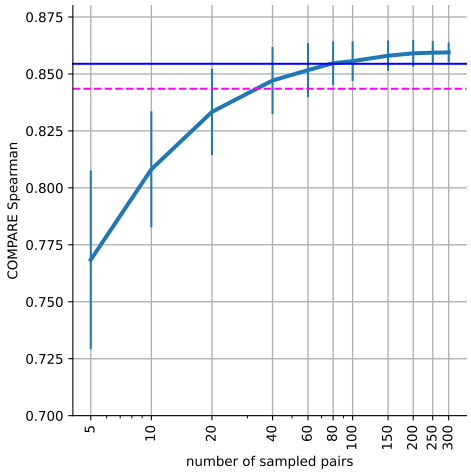

Figure 2: Spearman's correlation of our APD scores with the gold COMPARE scores depending on the number of COMPARE pairs sampled per word. Model: MCL→DWUG_es$^{bin2}_{ALL}$+XL-WSD. The solid blue horizontal line corresponds to all COMPARE pairs of the gold examples. The dashed purple horizontal line corresponds to the gold COMPARE pairs. Error bars show one standard deviation.

enough the results on the retrieved examples are a little bit higher on average than on the gold examples and significantly higher than on the gold COMPARE pairs. This is despite the fact that the gold scores were calculated based on human annotations of the gold pairs, and may be related to the imperfect approximation of human annotations by our WiC model.