GlossReader at LSCDiscovery: Train to Select a Proper Gloss in English – Discover Lexical Semantic Change in Spanish

Maxim Rachinskiy$^{\triangle}$ **Nikolay Arefyev**$^{\diamond,\triangledown,\triangle}$

$^{\triangle}$National Research University Higher School of Economics / Moscow, Russia
$^{\diamond}$Samsung Research Center Russia / Moscow, Russia
$^{\triangledown}$Lomonosov Moscow State University / Moscow, Russia
myurachinskiy@edu.hse.ru, nick.arefyev@gmail.com

Abstract

The contextualized embeddings obtained from neural networks pre-trained as Language Models (LM) or Masked Language Models (MLM) are not well suitable for solving the Lexical Semantic Change Detection (LSCD) task because they are more sensitive to changes in word forms rather than word meaning, a property previously known as the word form bias or orthographic bias (Laicher et al., 2021). Unlike many other NLP tasks, it is also not obvious how to fine-tune such models for LSCD. In order to conclude if there are any differences between senses of a particular word in two corpora, a human annotator or a system shall analyze many examples containing this word from both corpora. This makes annotation of LSCD datasets very labour-consuming. The existing LSCD datasets contain up to 100 words that are labeled according to their semantic change, which is hardly enough for fine-tuning.

To solve these problems we fine-tune the XLM-R MLM (Conneau et al., 2020) as part of a gloss-based WSD system on a large WSD dataset in English. Then we employ zero-shot cross-lingual transferability of XLM-R to build the contextualized embeddings for examples in Spanish. In order to obtain the graded change score for each word, we calculate the average distance between our improved contextualized embeddings of its old and new occurrences. For the binary change detection subtask, we apply thresholding to the same scores.

Our solution has shown the best results among all other participants in all subtasks except for the optional sense gain detection subtask.

1 Introduction

LSCDiscovery (D. Zamora-Reina et al., 2022) is a shared task on Lexical Semantic Change Detection (LSCD) in Spanish. In general, LSCD is the task of automatically analyzing differences between word senses in two corpora. In the shared task, these two corpora represent two time periods (1810-1906 and 1994-2020), and the participants are asked to analyze changes in the meaning of words over time, or diachronic change.

There are two main subtasks in the shared task: graded change and binary change detection. In the first subtask, the participants are asked to rank a list of words according to the magnitude of change in the relative frequencies of their senses (measured by the Jensen–Shannon distance between the probability distributions over senses automatically inferred by the organizers from the pairwise human annotations). In the second subtask, for each given word the systems should detect if the sets of its senses appearing in the old and the new corpus are different, i.e. if any new senses have appeared or any old senses are not in use anymore.

Despite the success of recurrent and Transformer-based neural networks pre-trained as language models (LM) or masked language models (MLM) on large corpora in a wide variety of NLP tasks, they cannot be applied to the LSCD task in a standard way. Most datasets used to fine-tune such models for different NLP tasks contain tens or hundreds of thousands examples, each of these examples is a text fragment not longer than several hundred words that contain all information required to make a correct prediction. In LSCD one example is a word, however, inspecting many occurrences of this word in both old and new corpora is required to draw correct conclusions about changes of its meaning. This requires a model that can extract information from many word occurrences and somehow aggregate it to produce the final prediction. Also, this makes creating labeled datasets for the task extremely labour-consuming, resulting in typical datasets containing less than 100 labeled words per language (Schlechtweg et al., 2021; Kutuzov and Pivovarova, 2021), which is hardly enough for fine-tuning.

Alternatively, in Laicher et al. (2021) the con-

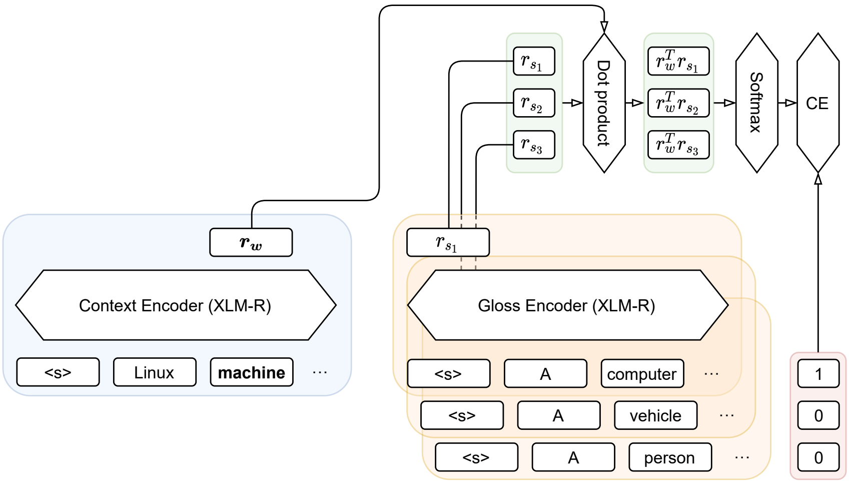

Figure 1: The multilingual gloss-based WSD model based on the BEM architecture.

textualized embeddings calculated by a pre-trained MLM without any fine-tuning were applied to solve the LSCD task. They found that the largest signal in these embeddings corresponds to the grammatical form, not to the meaning of words. This is known as the grammatical or orthographic bias of the contextualized embeddings and prohibits their direct application to the LSCD task.

The main idea behind our solution is that fine-tuning on some task that requires understanding word senses and at the same time ignoring word forms shall help to get rid of grammatical bias in the contextualized embeddings. A suitable task shall also have a large dataset for fine-tuning. In our solution of the LSCD task, we fine-tune a pre-trained MLM as part of a gloss-based WSD system, i.e. a system that can select the most appropriate gloss for a given word in a given context. Our WSD system is based on the architecture proposed in Blevins and Zettlemoyer (2020), however, we replace English BERT with multilingual XLM-R to make our system multilingual. We train the system on English WSD data only, then apply it to the texts in Spanish exploiting zero-shot cross-lingual transferability of XLM-R to obtain the contextualized embeddings for Spanish words.

Despite not using any labeled data in Spanish, the described method of fine-tuning XLM-R results in such contextualized embeddings that are directly applicable for lexical semantic change detection in Spanish. Our solution based on these contextualized embeddings has achieved the best results among all other participants in both main subtasks, and also in all optional subtasks except for the sense gain detection.[1]

2 Background

Our solution is inspired by the BEM (Bi-Encoder Model) system developed by Blevins and Zettlemoyer (2020) to solve the Word Sense Disambiguation (WSD) task in English. While WSD is essentially a classification task requiring to annotate each occurrence of polysemous words with one of their senses described in WordNet (Miller, 1995) or other sense inventory, a huge number of senses in WordNet (more than 100K) and zero or very few examples for most senses and words in the labeled training sets make standard classification approaches not applicable. Instead of treating senses as atomic classes, in BEM they are represented with their glosses from WordNet. Two encoders are introduced: the gloss encoder to build embeddings for glosses, and the context encoder to build contextualized embeddings for word occurrences. These encoders are trained jointly such that for each word occurrence among all glosses of this word a gloss describing its meaning in the given context can be selected by the similarity between the corresponding contextualized embedding and the gloss embeddings.

The original BEM system employs English BERT (Devlin et al., 2019) as both gloss and context encoders. The system is trained on the English WSD dataset SemCor (Miller et al., 1994). We replace English BERT with multilingual XLM-R (Conneau et al., 2020). XLM-RoBERTa (XLM-R for short) is basically the multilingual version

[1] Reproduction code: https://github.com/myrachins/LSCDiscovery

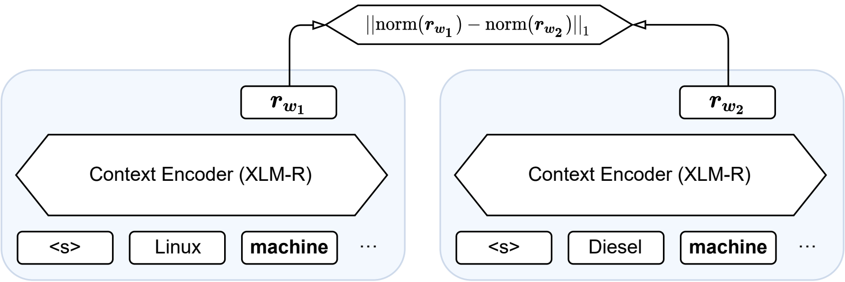

Figure 2: Employing the context encoder for distance estimation.

of RoBERTa (Liu et al., 2019), and RoBERTa is BERT (Devlin et al., 2019) with several improvements in the training procedure. All of these models essentially train the encoder of the Transformer-based machine translation system (Vaswani et al., 2017) with Masked Language Modeling (MLM) objective, i.e. to restore some words in a text fragment from nearby words (see Devlin et al. (2019) for technical details). In contrast to BERT and RoBERTa pre-trained on English texts only, XLM-R is pre-trained on 2.5TB of texts in 100 languages. Surprisingly, this allows not only processing texts in all of these languages but also demonstrates zero-shot cross-lingual transferability, meaning that after fine-tuning XLM-R to solve some classification task on English texts only, it often can solve the same task for texts in other languages reasonably well (Conneau et al., 2020).

Our approach to the LSCD task was initially developed during our participation in the RuShiftEval-2021 shared task on LSCD for the Russian language (Kutuzov and Pivovarova, 2021) and described in Rachinskiy and Arefyev (2021b). However, in RuShiftEval-2021 only a graded change detection task was proposed and the only metric was Spearman's correlation with the gold COMPARE score, which is the average similarity between word occurrences in two corpora (Schlechtweg et al., 2018). The LSCDiscovery shared task in Spanish offers a more thorough comparison of competing approaches by introducing both the graded change and the binary change detection subtasks. It also replaces the gold COMPARE scores with the gold Jensen-Shannon distance between the sense distributions inferred by the organizers, though calculating the gold COMPARE scores as an additional metric as well. Also in RuShiftEval-2021 our best solution was a linear regression model that used different distances between the contextualized embeddings as features and was trained on additional labeled data in Russian. This resulted in consistent but not very large improvement compared to simply using the raw distance between the contextualized embeddings. Thus, for the LSCDiscovery task, we decided to use the simpler solution that also does not require any labeled data in Spanish.

3 System overview

The architecture of our gloss-based WSD system is shown in figure 1. The architecture and the training procedure are borrowed from Blevins and Zettlemoyer (2020), except for the English BERT replaced with multilingual XLM-R in both context and gloss encoders. As usual for XLM-R, the input texts are surrounded by the special tokens `<s>` and `</s>`. To obtain the contextualized embedding for a word in context, the outputs at the positions of the target word are taken from the last layer of the context encoder. If the target word was split into subwords by the XLM-R tokenizer, then mean pooling is applied to the corresponding outputs. For each sense of the target word described in WordNet, the corresponding gloss is encoded by taking the output from the last layer of the gloss encoder at the position of the special `<s>` token.[2] The dot product between the contextualized embedding of the target word and the gloss embeddings for each of its senses is calculated, then the softmax function is applied to obtain the probability distribution over word senses.

The whole system is trained by minimizing the cross-entropy loss between the predicted distribution over senses and the correct sense. Following Blevins and Zettlemoyer (2020), we trained the

[2] This is the standard way of obtaining an embedding for the whole input sequence from MLM models, which is also used in the original BEM model. Some reasonable alternatives are averaging the outputs at all positions, or prepending the target word to each gloss and averaging the outputs at the positions of subwords of the target word. In any case, we believe that fine-tuning is important for obtaining good gloss embeddings.

system on English SemCor (Miller et al., 1994), which is a large dataset consisting of more than 200K sense-annotated word occurrences. The glosses were taken from WordNet 3.0 (Miller, 1995). The SemEval-2007 (Pradhan et al., 2007) WSD dataset served as the development set to choose the final checkpoint. The large version of XLM-R was employed for both encoders. We trained the system for 10 epochs, which took 3 days on two V100 GPUs. The XLM-R model fine-tuned as the context encoder of this WSD system is called the Gloss Language Model (**GLM**) below to distinguish it from the standard XLM-R pre-trained with the MLM objective only.

After the WSD system is trained, in order to estimate the similarity in meaning between two occurrences of the same word, we normalize their contextualized embeddings (divide them by their L1-norm) and calculate the Manhattan distance as shown in figure 2.

3.1 Graded subtasks

For all graded change subtasks, given each target word the score is calculated by the following algorithm.

1. Retrieve all occurrences of the target word in any of its forms from both corpora provided. We employed the same Spanish lemmatizer that was used by the task organizers. Then sample up to 100 pairs of sentences with the first sentence from the old corpus and the second from the new one.[3]

2. For each pair of sentences, calculate the L1-distance (the Manhattan distance) between the normalized embeddings of two occurrences of the target word. In order to normalize the embeddings, we divide them by their L1-norm. This choice is motivated by the previous experiments (Rachinskiy and Arefyev, 2021a,b).

3. Calculate the average of the distances from the previous step. This is known as the Average Pairwise Distance (APD) (Giulianelli et al., 2020).

The APD scores calculated by the last step of this algorithm seem to be a reasonable approximation of the gold COMPARE scores because they both represent the average similarity between word occurrences taken from two different corpora. But they are likely sub-optimal as an approximation of the gold JSD scores. In the future work, it is worth developing some alternatives to specifically approximate JSD.

The most computationally expensive part of this algorithm is calculating embeddings for about 778K word occurrences (4385 target words, 88.68 pairs of occurrences per word on average) This took about 6 GPU-hours on a V100 GPU. Computing distances and final scores takes an insignificant proportion of the whole time.

3.2 Binary subtasks

To obtain binary change predictions, we apply thresholding to our graded change predictions. During the competition, we experimented with two thresholding strategies. First, based on the observation that 9 out of 20 words (45%) in the development set belong to the negative class, we set the threshold equal to the 45-th percentile of APDs for the 60 hidden words revealed after the first subtask (**Thres. revealed**). This results in the same proportions of predicted classes in the test set as the proportions of true classes in the development set.

Alternatively, we calculated the 55-th[4] percentile of APDs for all 4385 target words in the test set from the first subtask **(Thres. all)**. The same binary predictions were submitted for all binary subtasks, which is likely suboptimal and is the subject for improvement in the future.

4 Results

Tables 1, 2 show our results compared to the baselines and to the best results of other participants. In the graded subtasks our solution achieves the best results among all participants. In the post-evaluation experiments, we compared the fine-tuned XLM-R model (GLM) with the original one (MLM). Evidently, fine-tuning XLM-R on the WSD task gives a huge boost in performance. Our APD scores have a much higher Spearman's correlation with the gold COMPARE scores than with the gold JSD scores, which supports our hypothesis that simple averaging of the distances between the contextualized embeddings is more suitable as an approximation of the COMPARE metric.

[3] In (Arefyev et al., 2021) it was observed that taking more than 100 pairs does not significantly improve the results, though this was observed for a different model.

[4] This should have been the 45-th percentile, but we made a mistake and calculated the 55-th percentile instead. In the post-evaluation period, we fixed this error (**Thres. all, fixed** method in Table 2).

Model	JSD	COMPARE
our submissions		
GLM norm L1	**.735** (1)	**.842** (1)
top3 other teams for each metric		
UsrD7	.702 (2)	.829 (2)
aishein	.553 (3)	.558 (4)
akutuzov	.508 (5)	.459 (5)
lscdiscovery baselines		
baseline1	.543 (4)	.561 (3)
baseline2	.092 (8)	.088 (6)
our post-evaluation experiments		
MLM norm L1	.505 (5*)	.511 (4*)

Table 1: Results for the graded subtasks, Spearman's correlation with the gold JSD and COMPARE scores. * denotes the ranks that we would have had if we had submitted only this result.

Model	bin. change	sense gain	sense loss
our submissions			
Thres. all	**.716** (1)	.491 (3)	**.688** (1)
Thres. revealed	.656 (4*)	.510 (3*)	.621 (1*)
top3 other teams for each metric			
dteodore	.709 (2)	.000 (8)	.000 (6)
rombek	.687 (3)	.490 (4)	.593 (3)
kudisov	.658 (4)	.520 (2)	.600 (2)
UsrD7	.655 (5)	**.591** (1)	.582 (4)
lscdiscovery baselines			
baseline1	.537 (9)	-	-
baseline2	.222 (10)	.211 (7)	.000 (6)
our post-evaluation experiments			
Thres. all, fixed	**.722** (1*)	.483 (4*)	.667 (1*)

Table 2: Results for binary subtasks, F1-scores. * denotes the ranks that we would have had if we had submitted only this result.

For the binary change detection and the sense loss detection subtasks our solution also outperforms all other participants. However, for the sense gain detection subtask our solution shows F1-scores of 0.483-0.510, which is about 10 points of F1-score worse than the best result in the competition. Notice that we did not specifically address the optional sense loss and sense gain detection subtasks, instead, we reused the predictions from the main binary change detection subtask.

5 Conclusion

In this paper, we presented a solution for both Graded and Binary Change Detection. Our solution achieves the best results among all participants in both graded change detection subtasks, as well as two out of three binary change detection subtasks. The key component of our solution which is shown to be very important is fine-tuning of a masked language model as part of a gloss-based WSD system.

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.

Terra Blevins and Luke Zettlemoyer. 2020. Moving down the long tail of word sense disambiguation with gloss informed bi-encoders. In *Proceedings of the 58th Association for Computational Linguistics*.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. 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: Papers from the annual conference Dialogue*.

Severin Laicher, Sinan Kurtyigit, Dominik Schlechtweg, Jonas Kuhn, and Sabine Schulte im Walde. 2021. Explaining and improving BERT performance on lexical semantic change detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 192–202, Online. Association for Computational Linguistics.

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.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

George A. Miller, Martin Chodorow, Shari Landes, Claudia Leacock, and Robert G. Thomas. 1994. Using a semantic concordance for sense identification. In *Proceedings of the Workshop on Human Language Technology*, HLT '94, page 240–243, USA. Association for Computational Linguistics.

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, Prague, Czech Republic. Association for Computational Linguistics.

Maxim Rachinskiy and Nikolay Arefyev. 2021a. GlossReader at SemEval-2021 task 2: Reading definitions improves contextualized word embeddings. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 756–762, Online. Association for Computational Linguistics.

Maxim Rachinskiy and Nikolay Arefyev. 2021b. Zeroshot crosslingual transfer of a gloss language model for semantic change detection. In *Computational linguistics and intellectual technologies: Papers from the annual conference Dialogue*, 20, page 578 – 586.

Dominik Schlechtweg, Sabine Schulte im Walde, and Stefanie Eckmann. 2018. Diachronic usage relatedness (DURel): A framework for the annotation of lexical semantic change. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 169–174, New Orleans, Louisiana. Association for Computational Linguistics.

Dominik Schlechtweg, Nina Tahmasebi, Simon Hengchen, Haim Dubossarsky, and Barbara McGillivray. 2021. DWUG: A large resource of diachronic word usage graphs in four languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7079–7091, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.