Using Two Losses and Two Datasets Simultaneously to Improve TempoWiC Accuracy

Mohammad Javad Pirhadi, Motahhare Mirzaei and **Sauleh Eetemadi**

Iran University of Science and Technology at Tehran, Iran

{mohammad_pirhadi, m_mirzaei96}@comp.iust.ac.ir, sauleh@iust.ac.ir

Abstract

WSD (Word Sense Disambiguation) is the task of identifying which sense of a word is meant in a sentence or other segment of text. Researchers have worked on this task (e.g. Pustejovsky, 2002) for years but it's still a challenging one even for SOTA (state-of-the-art) LMs (language models). The new dataset, TempoWiC introduced by Loureiro et al. (2022b) focuses on the fact that words change over time. Their best baseline achieves 70.33% macro-F1. In this work, we use two different losses simultaneously to train RoBERTa-based classification models. We also improve our model by using another similar dataset to generalize better. Our best configuration beats their best baseline by 4.23% and reaches 74.56% macro-F1.

1 Introduction

In 2019, Pilehvar and Camacho-Collados (2018) introduced WiC dataset. It is framed as a binary classification task between pairs of sentences including one identical target word with different meanings. In 2020, XL-WiC was introduced by Raganato et al. (2020) and made WiC richer by providing more examples and adding more languages. We benefit from the English part of XL-WiC as a helping dataset to improve the generalization of our model.

Loureiro et al. (2022b) baselines include: RoBERTa (Liu et al., 2019) base and large, TimeLMs (Loureiro et al., 2022a) 2019-90M and 2021-124M and BERTweet (Nguyen et al., 2020) base and large. They examine two different methods of using these models: Fine-tuning and SP-WSD layer pooling weights as explained in Loureiro et al. (2022c). The best result is for TimeLMs-2019-90M with SP-WSD with 70.33% macro-F1. We examine RoBERTa-base and TimeLMs-Jun2022-153M.

For classification, many previous works (e.g. Peters et al., 2019) use standard practice and concatenate both sentences with a `[SEP]` token and fine-tune the `[CLS]` embedding. In this work, we use two different losses simultaneously, cross entropy loss on RoBERTa classification head output as standard practice and add cosine embedding loss on average of target word output embeddings.

2 Methodology

2.1 Model

We use LMs as base model. We add classification head and also cosine similarity + sigmoid on top of them. The classification head consists of two FC (fully connected) layers and a dropout layer between them (like standard RoBERTa classification head). RoBERTa uses a byte-level BPE (Byte-Pair Encoding) encoding scheme so it's possible that we have multiple embeddings for a single word. For second output path, we average embeddings (it can be more than one as explained) related to target word in first sentence and second sentence and compare them using cosine similarity, finally we apply sigmoid activation to get a binary classification. Our experiments shows that the second output path is more accurate by a large margin. Figure 1 shows an overview of described architecture.

2.2 Loss Function

For the loss function, we have the sum of two losses, one on standard RoBERTa classification head and another on similarity (in case of the same meaning) or dissimilarity (in case of the different meaning) of the embeddings of the last layer related to the target word. For the former, we use cross entropy loss and for latter we use cosine embedding loss. The second loss, help our model to make similar contextual embeddings for target word closer and push dissimilar ones away from each other.
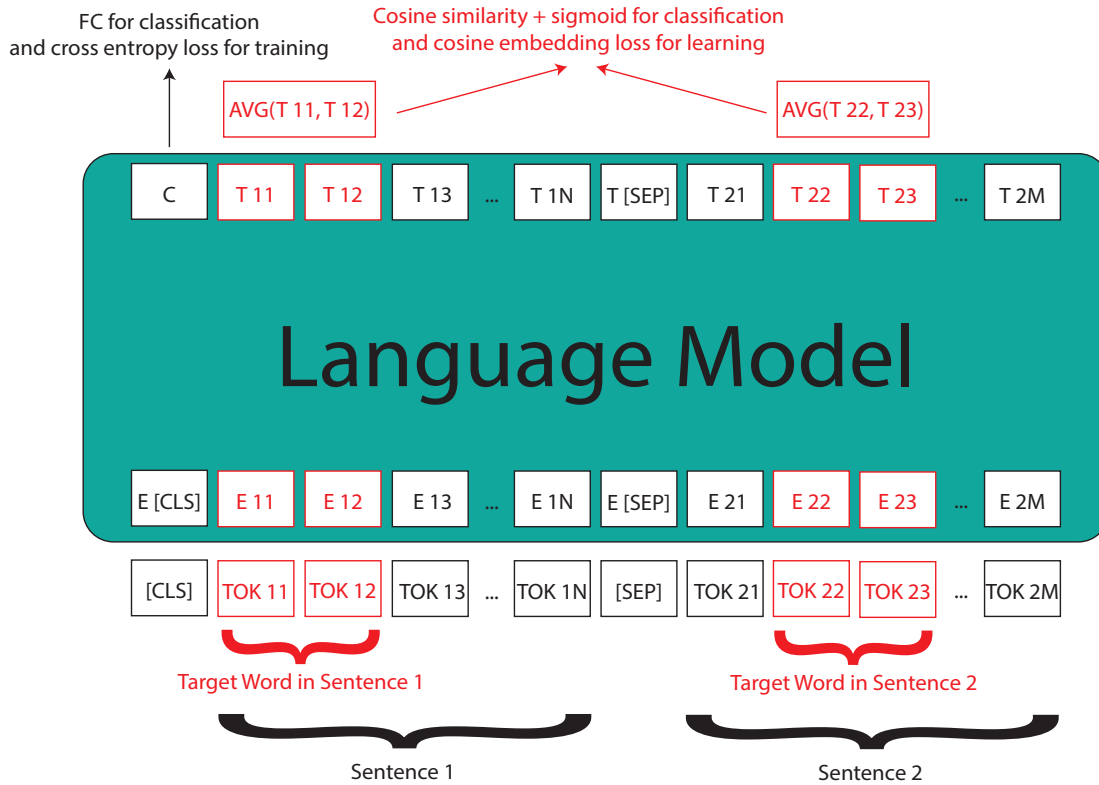
Figure 1: An overview of architecture. We use two losses simultaneously. First one is cross entropy loss on standard classifier head (black path) and the second one is cosine embedding loss on average of target word output embeddings (red path).

2.3 Dataset

The main dataset is TempoWiC, but we also use the XL-WiC dataset to make our model more robust. It's important to know that XL-WiC samples are not tweets so it is out-of-domain data and the added data may cause model accuracy to degrade if the combined dataset is not representative. We explored using the main dataset without adding any sample from the XL-WiC dataset, by adding a random subset of XL-WiC, and also by adding the whole XL-WiC.

2.4 Framework & Tools

We use PyTorch (Paszke et al., 2019) + Hugging-Face transformers (Wolf et al., 2020) to implement our models and for reporting results we use the Codalab online platform[1].

2.5 Hyper-parameters

We use Ray Tune (Liaw et al., 2018) to tune our hyper-parameters including learning rate, train epochs, random seed, batch size and weight decay.

Increasing weight decay helps us avoid over-fitting which was the main problem in our initial model.

3 Experiments

We have multiple configurations to test:

1. Model
   - RoBERTa-base
   - TimeLMs-Jun2022-153M

2. Output
   - Standard Classifier Head (FC)
   - Cosine Similarity + Sigmoid (CS+S)

3. How we use XL-WiC
   - Do not use (No)
   - A subset as described (Sub)
   - Whole XL-WiC (All)

3.1 Results

The biggest problem we were facing was over-fitting. This is expected since we use transformer-based LMs.

13

Output	Model	XL-WiC Use	Macro-F1
Classifier	RoBERTa-base	No	62.35%
		Sub	65.98%
		All	63.56%
	TimeLMs-Jun2022-153M	No	64.54%
		Sub	73.16%
		All	72.77%
Similarity	RoBERTa-base	No	67.26%
		Sub	68.29%
		All	67.29%
	TimeLMs-Jun2022-153M	No	66.69%
		Sub	**74.56%**
		All	72.32%
Official Baselines	TimeLMs-2019-90M-SIM	No	70.33%
	RoBERTa-L-SIM	No	67.09%
	RoBERTa-L-FT	No	59.10%
	TimeLMs-2019-90M-FT	No	57.70%
	Random	No	50.00%
	All True	No	26.79%

Table 1: All results are obtained from the Codalab online platform on Tempo-WiC test set.

The most accurate configuration is TimeLMs-Jun2022-153M with cosine similarity + sigmoid output trained on TempoWiC and a subset of XL-WiC. In the following paragraphs, we are going to analysis the results.

First, the results show that TimeLMs-Jun2022-153M beats RoBERTa in all possible configurations, the reason is simple: TempoWiC consists of tweets and TimeLMs-Jun2022-153M is trained on tweets too, but RoBERTa is not trained on tweets.

Second, using XL-WiC improves results in all cases. Using all XL-WiC example reduces the accuracy because the distribution is different (the samples are not tweets) and it has almost $4\times$ data in comparison to TempoWiC. If we use all of its data, we can not expect better accuracy because the training set distribution will be different from the test distribution.

Last, the cosine similarity + sigmoid output is better in most cases in comparison to the standard classifier head. We think it's because of more focus on the target word embedding in comparison to more focus on the whole context.

4 Future Work

In the future work, more configurations can be explored:

1. Selecting the subset of XL-WiC more wisely, instead of randomly selecting. For exam-

ple, considering the maximum possible use of unique words.

2. Using more layers to calculate similarity, instead of using only the last layer. For example, the sum of the last 4 layers is another common choice in word sense disambiguation settings.

3. Exploring more similarity functions, instead of cosine similarity. For example, euclidean distance can also be explored.

5 Conclusion

In this work, we beat the best baseline of Loureiro et al. (2022b) by a large margin. To do this we use two losses simultaneously (standard classifier head cross entropy loss and cosine embedding loss on average of target word output embeddings) to train SOTA LMs, and also use XL-WiC as a helping dataset to generalize better. The best LM was TimeLMs-Jun2022-153M which is a pre-trained model on 153M tweets.

6 Acknowledgements

References

Richard Liaw, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E. Gonzalez, and Ion Stoica. 2018. Tune: A research platform for distributed model selection and training.

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. *CoRR*, abs/1907.11692.

Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-collados. 2022a. TimeLMs: Diachronic language models from Twitter. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 251–260, Dublin, Ireland. Association for Computational Linguistics.

Daniel Loureiro, Aminette D'Souza, Areej Nasser Muhajab, Isabella A. White, Gabriel Wong, Luis Espinosa-Anke, Leonardo Neves, Francesco Barbieri, and Jose Camacho-Collados. 2022b. TempoWiC: An evaluation benchmark for detecting meaning shift in social media. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3353–3359, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Daniel Loureiro, Alípio Mário Jorge, and Jose Camacho-Collados. 2022c. Lmms reloaded: Transformer-based sense embeddings for disambiguation and beyond. *Artificial Intelligence*, 305:103661.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14, Online. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Matthew E. Peters, Mark Neumann, Robert L. Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. *CoRR*, abs/1909.04164.

Mohammad Taher Pilehvar and José Camacho-Collados. 2018. Wic: 10, 000 example pairs for evaluating context-sensitive representations. *CoRR*, abs/1808.09121.

James Pustejovsky. 2002. The generative lexicon. *Computational Linguistics*, 17.

Alessandro Raganato, Tommaso Pasini, José Camacho-Collados, and Mohammad Taher Pilehvar. 2020. Xlwic: A multilingual benchmark for evaluating semantic contextualization. *CoRR*, abs/2010.06478.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.