UoM&MMU at TSAR-2022 Shared Task: Prompt Learning for Lexical Simplification

Laura Vásquez-Rodríguez[1], **Nhung T. H. Nguyen**[1],
Matthew Shardlow[2], **Sophia Ananiadou**[1]
[1]National Centre for Text Mining,
The University of Manchester, Manchester, United Kingdom
[2]Department of Computing and Mathematics,
Manchester Metropolitan University, Manchester, United Kingdom
{laura.vasquezrodriguez,nhung.nguyen,sophia.ananiadou}@manchester.ac.uk
m.shardlow@mmu.ac.uk

Abstract

We present PromptLS, a method for fine-tuning large pre-trained Language Models (LM) to perform the task of Lexical Simplification. We use a predefined template to attain appropriate replacements for a term, and fine-tune a LM using this template on language specific datasets. We filter candidate lists in post-processing to improve accuracy. We demonstrate that our model can work in a) a zero shot setting (where we only require a pre-trained LM), b) a fine-tuned setting (where language-specific data is required), and c) a multilingual setting (where the model is pre-trained across multiple languages and fine-tuned in an specific language). Experimental results show that, although the zero-shot setting is competitive, its performance is still far from the fine-tuned setting. Also, the multilingual is unsurprisingly worse than the fine-tuned model. Among all TSAR-2022 Shared Task participants, our team was ranked second in Spanish and third in English.

1 Introduction

We present our system submission for the TSAR-2022 Shared Task (ST) on Lexical Simplification (Saggion et al., 2022). The task required participants to develop a lexical simplification system capable of taking a word in context and returning a list of candidate substitutions. The task provided test data in English (EN), Spanish (ES) and Brazilian Portuguese (PT). We chose to submit for all three tracks a system based on the concept of Prompt Learning. Whereas the previous state of the art for Lexical Simplification, LSBert (Qiang et al., 2020), masked the token in context, our approach, namely PromptLS, injects prompts within the context that forces the model to generate appropriate substitutions as in Table 1. We experimented with multiple prompts, varying the syntax and lexicon of the prompt, selecting the best-performing variants.

Context	Training sentences
No	a simple word for **classified** is [MASK] .
5 words (left and right)	triangles can also be classified *(a simple word for **classified** is [MASK])* according to their internal angles
All context	triangles can also be classified *(a simple word for **classified** is [MASK])* according to their internal angles , measured here in degrees .

Table 1: Data examples generated for fine tuning the LMs for the prompt template: *"a simple word for [MASK] is"*. We show the complex word in **bold**.

To fine-tune a language model using prompts, we firstly collected labelled data from different sources corresponding to the three languages. We then combined them and split the data into training and validation subsets. We also tested our prompts with a zero-shot and multilingual settings. As a result, PromptLS performed the best fine-tuned, compared to the multilingual and zero-shot settings.

We finally selected the best configurations to run on the official testing sets. Hence, we could observe the same pattern in the testing set as in our validation subsets, i.e., the fine-tuned setting still produced the best performance across languages.

2 Related Work

Lexical simplification arose as a form of assistive technology (Devlin, 1999; Carroll et al., 1999) for people with aphasia. Early systems used dictionary based replacement methods (Bott et al., 2012), with disambiguation methods to improve the selection of candidates (Paetzold and Specia, 2015).

Recently, simplification systems have focused on the use of transformer architecture to identify

Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022), pages 218 - 224
December 8, 2022 ©2022 Association for Computational Linguistics

appropriate replacements for a given word (Qiang et al., 2021). This can be applied at a single or multi-word level (Przybyła and Shardlow, 2020).

Prompt learning is a method of leveraging the learnt probabilities in a large pre-trained language model to solve NLP tasks (Brown et al., 2020; Liu et al., 2022). This can be done in a zero shot (Sun et al., 2021; Ni and Kao, 2022), or fine-tuned setting (Jiang et al., 2020). Prompt learning requires the design of a prompt (Ding et al., 2022), which can be engineered (Ding et al., 2021), or generated (Shin et al., 2020).

3 Methodology

In this section, we start by the description of our selected datasets (Section 3.1) and the design of our prompts (Section 3.2). We then describe our proposed method **PromptLS** that consists of three modules: 1) a large language model (LM) that generates candidates based on a given prompt (Section 3.3), 2) a fine-tuning module that guides the LM to select more appropriate substitutes (Section 3.4) and 3) a candidate filtering module which removes incorrect or inappropriate candidates (Section 3.5).

3.1 Data Collection

In this section we describe our collected state-of-the-art Lexical Simplification (LS) datasets for EN, ES and PT. We include a summary in Table 2.

- **(EN) LexMTurk** (Horn et al., 2014): a dataset obtained from the alignment of 137K sentences from English Wikipedia and Simple English Wikipedia. The LexMTurk corpus represents a random sample of 500 candidates, where each sentence was manually annotated by 50 MTurk[1] workers.
- **(EN) NNSEval** (Paetzold and Specia, 2016b): a dataset based on an user study of 400 non-native speakers who judged simplification samples from Wikipedia, LSEval and LexMTurk. The NNSEval datasets is a subset of 239 instances from LSEval (De Belder and Moens, 2012) and LexMTurk, which was improved and refined for LS using complexity annotations.
- **(EN) BenchLS** (Paetzold and Specia, 2016a): is a combined dataset of 929 instances based on LexMTurk and LSeval. All lexical candidates were improved and ranked by native speakers from the United States.

Language	Datasets	Instances
EN	LexMTurk	500
	NNSEval	239
	BenchLS	929
	CEFR	414
ES	EASIER	5,130
PT	SIMPLEX-PB-3.0	1,582

Table 2: All labelled datasets used in this work.

- **(EN) CEFR dataset** (Uchida et al., 2018): a dataset of 414 instances based on the Common European Framework of References for Languages (CEFR).[2] Sentences were extracted from university textbooks and words were filtered with the corresponding level based on words lists. Candidates were selected and ranked with the support online thesaurus and CEFR levels annotations.
- **(ES) EASIER corpus** (Alarcon et al., 2021): a collection 260 documents annotated by a linguist and verified by experts and a target audience. As a result, a LS Spanish dataset of 5130 instances was created, with at least one candidate per target word.
- **(PT) SIMPLEX-PB-3.0** (Hartmann and Aluisio, 2021): a Brazilian Portuguese corpus of 1582 instances which has been iterative improved from SIMPLEX-PB (Hartmann et al., 2018) and SIMPLEX-PB 2.0 (Hartmann et al., 2020) with manual annotations adapted to children needs as a main audience. These annotations include 52 different features including complex words definitions and linguistic information.

3.2 Template Design

For the implementation of prompt-learning in Lexical Simplification we have designed a template using equivalent keywords or substitutes appropriate for each language. For example, in English, we used a template composed by two prompts as follows:

A(n) <Prompt1> <Prompt2> for <target_word>

The templates for Spanish and Portuguese are translations of the English template, which resulted better with performance in comparison with other alternatives evaluated. The selected prompts for each language are listed in Table 3.

[1] https://www.mturk.com/

[2] https://www.coe.int/en/web/common-european-framework-reference-languages

LN	Prompt1	Prompt2
EN	easier, simple	word, synonym
ES	palabra, sinónimo	fácil, simple
PT	palavra, sinônimo	fácil, simples

Table 3: All prompts used in our work. Notice that for ES and PT, the equivalent prompts has to be inverted with respect to English due to grammar rules.

We used the masked token tailored to each model (e.g., [MASK] token) to predict less complex words instead. We also investigate the impact of context around a target word on the model by adding context words into the training sentences. Table 1 illustrates our selected prompts in English for 3 defined scenarios: no context, context within a window size (delimited by a number of characters on each side) and all context, where all the sentence is considered. We selected the best-performing prompts after experimenting with multiple templates.

3.3 Language Models

We selected our models based on their language, size and performance to evaluate a prompt-learning setting. These models were trained for Masked language modeling (MLM) objective.[3]

- (multiL[4]) **mBERT** (Devlin et al., 2019): a BERT-based model trained over a large multilingual corpus using Wikipedia in 102 languages in a unsupervised way.
- (EN) **RoBERTa-large** (Liu et al., 2019): an improved version of BERT (Devlin et al., 2019) model, trained in a large English corpus (160GB of uncompressed data) with no labels (i.e., unsupervised).
- (ES) **BERTIN** (De la Rosa et al., 2022): a RoBERTa-based model trained in a the Spanish portion of mC4 dataset (Raffel et al., 2022), which has 1 TB of uncompressed data. Due to the difficulties of using such a large corpus, a subsection of the dataset was selected using perplexity sampling.
- (PT) **BR_BERTo**[5]: a roBERTa-based model trained on 6.9M of sentences in PT.

3.4 Fine Tuning

To fine tune our LMs, given a sentence from the original dataset and its target word (i.e., complex), we generate a source sentence by masking the target word in our prompt. Then, we generate the target sentence by replacing the masked token ([MASK]) in the source sentence with its top-k simplified candidates. As a result, for each sentence containing a complex word, we have k target sentences. For example, with the training sentence in the first row of Table 1, with $k = 3$ we have the following target sentences:

a simple word for **classified** is *grouped* .
a simple word for **classified** is *organized* .
a simple word for **classified** is *categorized* .

We performed similarly with the other scenarios (n-words context, all-context). Then, we repeated this generation process with all our templates (see Section 3.2) and across the three languages.

3.5 Candidate Filtering

To maximise the accuracy of our model, we implemented a post-processing step to remove unsuitable candidates. To decide best on the filtering strategies, we performed a manual analysis of the results from the trial data provided by the ST.[6] For all three languages, we remove characters that could represent an undefined candidate such as “unknown” or “[UNK]”. Also, we removed the complex candidate and any non-words that could be suggested by the model. For Spanish and Portuguese, we lowercased all candidates and kept only those words of length higher than 2. We also removed duplicated candidates. Finally, for English, we filtered antonyms using Wordnet.[7]

4 Experiments

4.1 Datasets

For English, we concatenated all the datasets and removed duplicates in the combined corpus. For Spanish and Portuguese, we used the EASIER and SIMPLEX corpora, respectively. In all languages, the corpus was split in two portions: 90% for training and 10% for validation, using a random sampling. We used the official release of the gold-standard from the ST as the testing set.

4.2 Training Settings

We test PromptLS in three different settings:

1. **Zero-shot**: we input the source sentences templates with the complex candidate into the MLM and obtain top-k simple candidates.

[3] Please refer to the Appendix A for additional systems that we considered for our benchmarks.

[4] We refer to our multilingual models as multiL.

[5] `https://huggingface.co/rdenadai/BR_BERTo`

[6] `https://github.com/LaSTUS-TALN-UPF/TSAR-2022-Shared-Task/tree/main/datasets`

[7] `https://www.nltk.org/howto/wordnet.html`

LN	Model	Setting	Prompt1	Prompt2	w	k	Acc@1	A@3	M@3	P@3
EN	RoBERTa-L	zero-shot	easier	word	all	0	0.378	0.303	0.251	0.606
			easier	word	10	0	0.356	0.553	0.251	0.612
		fine-tune	simple	word	5	5	0.830	0.899	0.644	0.941
			easier	word	5	5	0.803	0.904	0.644	0.941
	mBERT	multiL	easier	word	10	10	0.681	0.718	0.503	0.824
			easier	synonym	5	7	0.644	0.739	0.510	0.840
ES	BETO	zero-shot	palabra	simple	10	10	0.064	0.115	0.031	0.115
			palabra	fácil	10	2	0.053	0.103	0.030	0.103
	BERTIN	fine-tune	sinónimo	fácil	all	3	0.396	0.589	0.191	0.589
			palabra	simple	all	3	0.402	0.559	0.184	0.559
	XLM-RoBERTa-L	multiL	sinónimo	fácil	5	10	0.304	0.409	0.136	0.409
			sinónimo	simple	10	10	0.302	0.404	0.135	0.406
PT	ALBERT-pt	zero-shot	sinônimo	fácil	5	1	0.013	0.045	0.010	0.045
			sinônimo	simples	10	3	0.013	0.039	0.008	0.039
	BR_BERTo	fine-tune	palavra	simples	all	8	0.497	0.594	0.420	0.600
			sinônimo	fácil	all	10	0.516	0.574	0.433	0.594
	XLM-RoBERTa-L	multiL	sinônimo	sinônimo	10	5	0.271	0.406	0.180	0.439
			sinônimo	simples	5	5	0.277	0.419	0.188	0.452

Table 4: Best-performing configurations on the validation set for each model. **LN** refers to "Language", ***w*** to the number of tokens in the context window, ***k*** is the number of candidates used to augment the training data, **Acc@1** refers to MAP@1/Potential@1/Precision@1, **A@3** to Accuracy@3@top_gold_1, **M@3** to MAP@3, and **P@3** to Potential@3.

2. **Fine-tuned MLM**: we train the model with the augmented source sentences and their corresponding labels to fine-tune the MLM. At inference step, the steps are similar to the zero-shot setting.

3. **Multilingual**: We run both (i) and (ii) scenarios using multilingual MLMs.

We also combine the three settings with different sizes of the window context including 5, 10, and all context. The performance of PromptLS was additionally evaluated with different k numbers of top-k candidates used to generate the training data ($k = 1, 3, 5, 7, 10$).

In all experiments, we used the evaluation script provided by the organiser (Saggion et al., 2022) to calculate the following metrics: **MAP@K** (Mean Average Precision @ K) with K=1,3,5,10; **Potential@K** with K=1,3,5,10; **Accuracy@K@top1** with K=1,2,3.

4.3 Training Details

We performed our training using 2 NVIDIA v100 GPU (16GB RAM) using the HuggingFace (Wolf et al., 2020) framework for the implementation of our models. Our models were trained for 5 epochs, with a learning rate of 5e-5 using AdamW optimizer, a batch size of 8, a linear scheduler with no warm-up steps and a Cross Entropy loss. We did not perform further variations on these hyperparameters due to the increased variability of our prompt-based experiments.[8]

5 Results

For English, we executed 48 runs in a zero-shot setting, 240 for the fine-tuned MLM, and 192 for the multilingual settings. For Spanish, we executed 160 runs for the zero-shot and fine-tuned model and 140 for the multilingual setting. Similarly, for Portuguese, we ran 106 runs for zero-shot, 169 for fine-tuned setting and 144 for our multilingual setting.

Overall, we ran more than 600 experiments for each language with multiple combinations of prompts, context windows, number of candidates for data augmentation, models and settings for the selection of our submitted system.[9] In Table 4, we include the best two configurations of each model

[8] Our code is available on Github: https://github.com/lmvasque/ls-prompt-tsar2022

[9] We publish our settings selection scripts on Github: https://github.com/lmvasque/ls-prompt-tsar2022/tree/main/scripts/benchmark

LN	#	Model	Setup	Prompt1	Prompt2	w	k	Acc@1	A@3	M@3	P@3
EN	1	RoBERTa-L	fine	simple	word	5	5	**0.6353**	**0.5308**	**0.4244**	**0.8739**
	2	mBERT	multi	easier	word	10	10	0.4959	0.4235	0.3273	0.7560
	3	RoBERTa-L	zero	easier	word	5	0	0.2654	0.268	0.1820	0.4906
ES	1	BERTIN	fine	sinónimo	fácil	0	3	0.3451	**0.2907**	**0.2238**	**0.5543**
	2			palabra	simple	0	10	0.3614	**0.2907**	0.2225	0.538
	3			sinónimo	fácil	10	10	**0.3668**	0.269	0.2128	0.5326
PT	1	BR_BERTo	fine	palavra	simples	0	8	**0.1711**	0.1096	0.1011	0.2486
	2			sinônimo	fácil	0	10	0.1363	0.0962	0.0944	0.2379
	3			sinônimo	simples	5	10	0.1577	**0.1283**	**0.1071**	**0.2834**

Table 5: Results on the official testing set. “LN" refers to Language, “#” to the number of submission, “fine" to fine-tune and “zero" to zero-shot.

in the benchmark for each language. Nevertheless, for our ST submission, we selected the three best-performing runs in the development set using a ranking of all the models results. We show our final systems in Table 5.

6 Discussion

In Table 4, we showed that in the case of English, using zero-shot combined with context produced a relatively reasonable performance. Meanwhile, it is not the case for Spanish and Portuguese, which scored significantly lower. Such performance can be attributed to the size of the MLMs. In contrast, the fine-tuning setting led to a higher performance although we used small annotated corpora to fine-tune the MLMs. The performance gap between zero-shot and fine-tuning ones is between 0.3 and 0.4 across metrics and languages. It is unsurprising that the multilingual LMs did not outperform the monolingual ones.

Candidates for the prompts (e.g., “easier”, “word”) affected the performance of PromptLS. In the English language, the best prompts are composed by “easier word” and “simple word”. Meanwhile, it was more suitable to use “palabra simple” and “palabra fácil” for the Spanish model and “palavra simples” and “sinônimo simples” for the Portuguese model. Our selections in Portuguese were done based on the knowledge of a second-language learner without the support of a native Brazilian Portuguese speaker. Therefore, there might be space for improvement on the selected settings of this model.

In addition, we observed that context words around a complex word are important in this task. In all the settings reported in Table 4, we had to use at least a window of 5 words to obtain good performance. Using multiple candidates for a complex word to augment the training data helps improve the performance as well. Finally, we selected the best settings and applied them to the official testing set. The results (Saggion et al., 2022) of our three runs in each language are reported in Table 5.

Concerning the model selection, it is noted that T5 model (Raffel et al., 2022) is also a suitable baseline for a prompt-based setting. However, unlike the experimental MLMs, T5 has a decoder, which requires additional effort to apply it to Lexical Simplification. We therefore leave this implementation for future work.

7 Conclusion

In this paper, we presented the implementation of a prompt-learning system for LS. Our experiments indicate we can obtain reasonable results even in zero-shot settings, especially for full resourced languages such as English. We demonstrate that by fine-tuning our prompt templates, we obtain competitive results in all languages. As future work, we intend to experiment with better datasets, including better filtering and ranking methods for LS.

8 CRediT author statement

Laura Vásquez-Rodríguez: Methodology, Software, Validation, Writing - Original Draft, Writing – Review & Editing. **Nhung T. H. Nguyen**: Methodology, Software, Validation, Writing - Original Draft, Writing – Review & Editing. **Matthew Shardlow**: Conceptualization, Methodology, Software, Writing - Original Draft, Writing – Review & Editing. **Sophia Ananiadou**: Funding acquisition, Supervision.

References

Rodrigo Alarcon, Lourdes Moreno, and Paloma Martinez. 2021. Lexical simplification system to improve web accessibility. *IEEE Access*, PP:1–1.

Stefan Bott, Luz Rello, Biljana Drndarevic, and Horacio Saggion. 2012. Can Spanish be simpler? LexSiS: Lexical simplification for Spanish. In *Proceedings of COLING 2012*, pages 357–374, Mumbai, India. The COLING 2012 Organizing Committee.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

John Carroll, Guido Minnen, Darren Pearce, Yvonne Canning, Siobhan Devlin, and John Tait. 1999. Simplifying text for language-impaired readers. In *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 269–270, Bergen, Norway. Association for Computational Linguistics.

José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*.

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.

Jan De Belder and Marie-Francine Moens. 2012. A dataset for the evaluation of lexical simplification. In *Computational Linguistics and Intelligent Text Processing*, pages 426–437, Berlin, Heidelberg. Springer Berlin Heidelberg.

Javier De la Rosa, Eduardo G Ponferrada, Manu Romero, Paulo Villegas, Pablo González de Prado Salas, and María Grandury. 2022. Bertin: Efficient pre-training of a spanish language model using perplexity sampling. *Procesamiento del Lenguaje Natural*, 68.

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.

Siobhan Lucy Devlin. 1999. *Simplifying natural language for aphasic readers.* Ph.D. thesis, University of Sunderland.

Ning Ding, Yulin Chen, Xu Han, Guangwei Xu, Pengjun Xie, Hai-Tao Zheng, Zhiyuan Liu, Juanzi Li, and Hong-Gee Kim. 2021. Prompt-learning for fine-grained entity typing. *arXiv preprint arXiv:2108.10604*.

Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Haitao Zheng, and Maosong Sun. 2022. OpenPrompt: An open-source framework for prompt-learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 105–113, Dublin, Ireland. Association for Computational Linguistics.

Nathan S. Hartmann, Gustavo H. Paetzold, and Sandra M. Aluísio. 2018. Simplex-pb: A lexical simplification database and benchmark for portuguese. In *Computational Processing of the Portuguese Language*, pages 272–283, Cham. Springer International Publishing.

Nathan S. Hartmann, Gustavo H. Paetzold, and Sandra M. Aluísio. 2020. A dataset for the evaluation of lexical simplification in portuguese for children. In *Computational Processing of the Portuguese Language*, pages 55–64, Cham. Springer International Publishing.

Nathan Siegle Hartmann and Sandra Maria Aluisio. 2021. Automatic lexical adaptation in brazilian portuguese informative texts for elementary education. *Linguamatica*, 12(2).

Colby Horn, Cathryn Manduca, and David Kauchak. 2014. Learning a lexical simplifier using Wikipedia. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 458–463, Baltimore, Maryland. Association for Computational Linguistics.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2022. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*. Just Accepted.

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.

Shiwen Ni and Hung-Yu Kao. 2022. Electra is a zero-shot learner, too.

Gustavo Paetzold and Lucia Specia. 2015. LEXenstein: A framework for lexical simplification. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 85–90, Beijing, China. Association for Computational Linguistics and The Asian Federation of Natural Language Processing.

Gustavo Paetzold and Lucia Specia. 2016a. Benchmarking lexical simplification systems. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3074–3080, Portorož, Slovenia. European Language Resources Association (ELRA).

Gustavo Paetzold and Lucia Specia. 2016b. Unsupervised lexical simplification for non-native speakers. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

Piotr Przybyła and Matthew Shardlow. 2020. Multi-word lexical simplification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1435–1446, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jipeng Qiang, Yun Li, Zhu Yi, Yunhao Yuan, and Xindong Wu. 2020. Lexical simplification with pretrained encoders. *Thirty-Fourth AAAI Conference on Artificial Intelligence*, page 8649–8656.

Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, Yang Shi, and Xindong Wu. 2021. Lsbert: Lexical simplification based on bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3064–3076.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2022. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Horacio Saggion, Sanja Štajner, Daniel Ferrés, Kim Cheng Sheang, Matthew Shardlow, Kai North, and Marcos Zampieri. 2022. Findings of the tsar-2022 shared task on multilingual lexical simplification. In *Proceedings of TSAR workshop held in conjunction with EMNLP 2022*.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Yi Sun, Yu Zheng, Chao Hao, and Hangping Qiu. 2021. Nsp-bert: A prompt-based zero-shot learner through an original pre-training task–next sentence prediction.

Satoru Uchida, Shohei Takada, and Yuki Arase. 2018. CEFR-based lexical simplification dataset. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

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.

A Baselines

In addition to the submitted baselines, we also considered the following systems:

- (multiL) **XLM-RoBERTa-large** (Conneau et al., 2020): a multilingual RoBERTa-based model trained over 2.5T of data from the CommonCrawl in 102 languages in an unsupervised way.
- (EN) **bert-large-uncased** (Devlin et al., 2019): a BERT-based model trained on the BookCorpus, a dataset of 11,038 books and English Wikipedia.
- (EN) **ALBERT-large** (Lan et al., 2020): a BERT-based model optimised to consume less memory with reduced training time. It is also trained for sentence order prediction task with a self-supervised loss to compensate its performance drop from the parameters reduction.
- (ES) **BETO** (Cañete et al., 2020): a BERT-based model trained in large (300M lines) Spanish corpora from different sources[10].
- (PT) **ALBERT-pt-br**[11]: an ALBERT-based model trained in Brazilian Portuguese data.
- (PT) **RoBERTa-pt-br**[12]: a roBERTa-based model trained in Brazilian Portuguese data.

[10] https://github.com/josecannete/spanish-corpora

[11] https://huggingface.co/josu/albert-br

[12] https://huggingface.co/josu/roberta-pt-br