PresiUniv at TSAR-2022 Shared Task: Generation and Ranking of Simplification Substitutes of Complex Words in Multiple Languages

Peniel John Whistely, Sandeep Mathias and **Galiveeeti Poornima**
Information Retrieval Lab, Department of Computer Science and Engineering
Presidency University, Bangalore
{peniel.20212AIE0002, sandeepalbert, galiveetipoornima}@presidencyuniversity.in

Abstract

In this paper, we describe our system, **PresiUniv**, to generate and rank candidate simplifications using publicly available pre-trained language models (BERT, BETO, and BERTimbeau), word embeddings (Eg. FastText, NILC), and part-of-speech taggers (NLTK PoS Tagger, Stanford PoS Tagger and Mac-Morpho), to generate and rank candidate contextual simplifications for a given complex word. In this shared task, our system was placed **first** in the Spanish track, 5th in the Brazilian-Portuguese track, and 10th in the English track. We upload our codes and data for this project to aid in replication of our results. We also analyze some of the errors and describe design decisions which we took while writing the paper.

1 Introduction

Lexical Simplification (LS) is a task of natural language generation that aims to substitute difficult words and phrases in a sentence for simpler ones that convey the same information (Paetzold and Specia, 2017). This is a challenging task because not only must the substitution retain the original meaning while still adhering to the grammatical requirements of the sentence that is being simplified, but different people may have different needs for simplification (Alva-Manchego et al., 2020). Figure 1 shows the pipeline for lexical simplification (Shardlow, 2014).

In light of this, the TSAR 2022 Workshop organized a shared task on lexical simplification, where participating teams have to generate and rank simplifications for a given complex word (Saggion et al., 2022). Each team is allowed to submit three runs for their system. This paper describes the performance of our team, **PresiUniv**[1] at this shared task.

[1]Code:http://www.github.com/lwsam/TSAR-2022/

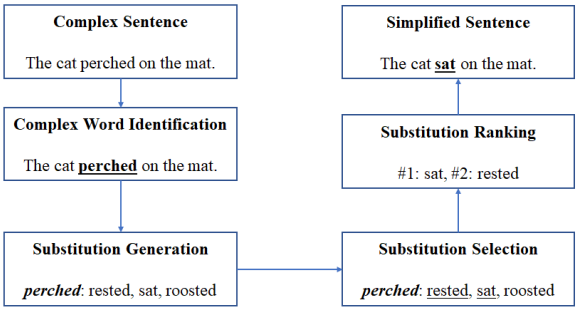

Figure 1: Pipeline of Lexical Simplification

2 Problem Statement

Our problem is defined as follows:

"Given a context and a possible complex word, we need to generate a *ranked list* of candidate simplifications."

Hence, our task is divided into two sub-tasks. The first sub-task involves generating words that would replace a complex word in the target sentence, which would simplify it. The second subtask consists of ranking the top 10 most suitable words.

3 Related Work

Lexical simplification must identify complex words and choose the optimal replacement (Shardlow, 2014; Paetzold and Specia, 2017). Previous shared tasks have already been done as a part of **SemEval 2016** (Paetzold and Specia, 2016) and **BEA 2018** (Yimam et al., 2018). While the first shared task dealt with a single training and test set in English alone, the second shared task dealt with complex word identification in multiple languages (English, German, and Spanish), as well as a multilingual scenario (where the system is tested in a fourth language, French).

Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022), pages 213 - 217

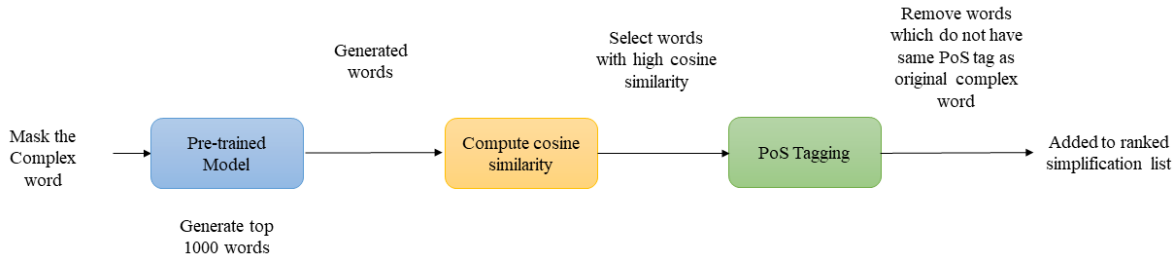

Figure 2: The method that we used for simplification

Language	Pre-trained Language Model	Word Vectors	Part-of-Speech Tagger
English	BERT	FastText	NLTK PoS Tagger
Spanish	BETO	FastText	Stanford PoS Tagger
Brazilian Portuguese	BERTimbeau	NILC Embeddings	Mac-Morpho

Table 1: Resources used for each language

4 Method

Figure 2 shows the different steps that we take to generate and select our candidates. It consists of the following steps:

1. Generation of candidate tokens
2. Candidate word selection
3. Candidate word pruning

Consider the following input sentence: "A Spanish government source, however, later said that banks able to cover by themselves losses on their toxic property assets will not be forced to remove them from their books while it will be **compulsory** for those receiving public help." Let the target word (the one being replaced) be "compulsory"[2].

4.1 Candidate Token Generation

We first generate a list of the top k tokens using a pre-trained language model (Eg. BERT-base-uncased (Devlin et al., 2019)). The pre-trained language model generally selects the most probable word to replace the masked token[3]. Since simpler words are more probable than more complex words (Leroy and Kauchak, 2014), we consider that the words generated are already ranked in order of difficulty from simplest to hardest.

Hence the above example sentence becomes "A Spanish government source, however, later said that banks able to cover by themselves losses on their toxic property assets will not be forced to remove them from their books while it will be **[MASK]** for those receiving public help." The generated tokens (in order of probability) are: "available", "easier", "safe", "beneficial", "provided", "safer", "better", "convenient", "appropriate", "done", "mandatory", ...

4.2 Candidate Word Selection

The next step is to select ***only*** the words which are suitable in meaning to the complex word. For example, the word "done" is not exactly a synonym for the word "compulsory"[4]. On the other hand, the word "mandatory" is a synonym[5]. In order to do that, we select words whose similarity is **above** a threshold value but less than 1 (because a cosine similarity of 1 would imply that the replacement is the same as the original complex word). For a threshold value of 0.50, we select the words "mandatory", "obligatory", "voluntary" and "mandated".

4.3 Candidate Word Pruning

Finally, we prune the selected words selected using a part-of-speech tagger to ensure that the chosen words with the correct inflexion as the complex word are chosen. From the above four words, we see that the word "mandated" is not of the same part of speech as "compulsory" (verb vs adjective)[6], and hence, the final ranked list of words is

[2]This example is taken from the trial dataset of the shared task.

[3]We trim out tokens which are not completely alphabetic, like "##ching"

[4]The cosine similarity using our word embeddings between **done** and **compulsory** is 0.119

[5]The cosine similarity using our word embeddings between **mandatory** and **compulsory** is 0.767

[6]In the given context, "mandated" would behave as an *adjectival*.

Rank	English		Spanish		Brazilian-Portuguese	
	Team	Acc@1	Team	Acc@1	Team	Acc@1
1	UniHD	0.8096	**PresiUniv**	**0.3695**	GMU-WLV	0.4812
2	MANTIS	0.6568	UoM&MMU	0.3668	Cental	0.3689
3	UoM&MMU	0.6353	PolyU-CBS	0.3586	PolyU-CBS	0.3262
4	LSBert	0.5978	GMU-WLV	0.3532	LSBert	0.3262
5	RCML	0.5442	Cental	0.3097	**PresiUniv**	**0.3074**
6	GMU-WLV	0.5174	LSBert	0.2880	TUNER	0.2219
7	CL Lab PICT	0.5067	TUNER	0.1195	UoM&MMU	0.1711
8	teamPN	0.4664	OEG_UPM	0.1032	-	-
9	PolyU-CBS	0.4316	-	-	-	-
10	**PresiUniv**	**0.4021**	-	-	-	-
11	CILS	0.3860	-	-	-	-
12	Cental	0.3619	-	-	-	-
13	TUNER	0.3404	-	-	-	-
14	twinfalls	0.1957	-	-	-	-
15	NU HLT	0.1447	-	-	-	-

Table 2: Comparison of our system with other systems. The ranking of the systems is as per the Accuracy@1 values of the best run submitted by the team. The results also include the performances by a pair of baseline systems - LSBert and TUNER (Štajner et al., 2022).

"mandatory", "obligatory" and "voluntary".

The solution from the gold file (without ties and space separated) is "mandatory required essential forced important necessary obligatory unavoidable".

5 Dataset

There is no training dataset for the TSAR-2022 Shared Task. A sample of 10 or 12 instances with gold standard annotations is provided here as the trial dataset. For the testing data, between 368 to 374 instances were given, with the annotations released upon the completion of the competition.

5.1 Trial dataset

The trial dataset consists of a set of 10 instances (for English and Portuguese) and 12 instances (for Spanish) of a sentence, a target complex word. The trial_none files contain only the sentences and the complex word, while the trial_gold files contain the sentences, the complex word and a set of gold simplifications.

5.2 Test dataset

The test_none files (used for the evaluation benchmark) contain the instances with the sentences and target complex words. The English test_none file had 373 instances, the Spanish test_none file had 368 instances, and the Brazilian Portuguese test_none file had 374 instances. The test_gold files contain the sentences, target complex words, and gold annotations for each of the test_none files.

6 Experimental Setup

6.1 Resources Used

In our experiments, we used the following resources:

- A **pre-trained language model** to generate a list of contextual candidate words to replace the complex word.
- A set of **dense word vectors** to find out which words that were generated earlier are similar in meaning to the complex word.
- A **part-of-speech tagger** to tag the sentence with the replacement and verify that the replacement word is of the same inflexion as the original complex word.

Due to the language requirements, we use a different set of resources for each language. Table 1 shows the different resources used for each language. For English, we used the **BERT** (Devlin et al., 2019) pre-trained language model, 300 dimension FastText (Grave et al., 2018) word vectors, and the default NLTK Part-of-Speech tagger with the Penn Treebank Tagset (Marcus et al., 1994). For Spanish, we used the **BETO** (Cañete et al.,

2020) pre-trained language model, 300 dimension FastText (Grave et al., 2018) word vectors, and the Stanford Part-of-Speech tagger (Toutanova et al., 2003). For Portuguese, we used the **BERTimbeau** (Souza et al., 2020) pre-trained language model, 300 dimension NILC Embeddings (Hartmann et al., 2017) and the Mac-Morpho part-of-speech tagger (Aluísio et al., 2003).

We set the value of k (the number of candidates generated) at **1000**, and we run our experiments for thresholds of similarity as **0.40**, **0.50**, and **0.60**.

6.2 Evaluation Metric

The following evaluation metrics are used for our experiments:

- Mean Accurate Precision - **MAP@K** [K=1,3,5,10]. MAP@K for Lexical Simplification evaluates the following aspects
  - Are the predicted substitutes relevant?
  - Are the top-ranked predicted substitutes at the top positions?
- **Potential@K** [K=1,3,5,10] - The percentage of instances for which at least one of the substitutions predicted is present in the set of gold annotations.
- **Accuracy@K** [K=1,3,5,10] - The ratio of instances where at least one of the K top predicted candidates matches the most frequently suggested synonym/s in the gold list of annotated candidates.

7 Results and Analysis

The results of our experiments on the testing dataset are given in Table 2. These results denote the best performance of a given team based on the MAP@1 for their three runs. While our system performed admirably in the Spanish lexical simplification ranking task (coming **first** overall), we did not do as well overall in the other languages.

7.1 Error Analysis

As we saw in the example in Section 4, ***antonyms*** can also be selected as candidates. For instance, let us consider the words **good** and **bad**, which have a high cosine similarity[7]. Both the words are antonyms, yet they would be selected as a replacement for the other because they have a high cosine similarity and the same part of speech.

[7]The cosine similarity between **good** and **bad** is 0.752.

7.2 Discussion

In this section, we discuss a couple of important design decisions which we made for our experiments. The first decision that we took was the order of the approaches. One of the approaches which we considered was to first select a similar word and then compute the language model score and rank the output words by the most probable sentences. However, this does not work out because the most similar words are usually ***different forms*** of the original word. For example, the top 5 most similar words for "compulsory" are: "Compulsory", "mandatory", "non-compulsory", "compulsary", and "complusory". As we can see, the most common words are either different forms of "compulsory", or they are spelling mistakes (Eg. "compulsary" and "compulusory"), with very few good candidate words (like "mandatory").

The next design decision is the values of the thresholds for cosine similarity, which we selected. Selecting a very low threshold for candidate selection will ensure that almost all the candidates generated will be selected, while a high threshold will eliminate almost all candidates (Eg. if we had a threshold of 0.8, then even candidates like "mandatory" won't be selected for "compulsory"). This is also why we selected threshold values of 0.40, 0.50 and 0.60 for our experiments.

8 Conclusion and Future Work

In this paper, we describe the participation of our team, **PresiUniv**, in the TSAR 2022 Shared Task on the generation and ranking of lexical simplification substitutes. Overall, we achieved the best performance in the Spanish track but finished 5th in the Portuguese track and 10th in the English track.

In the future, we plan to extend our work towards document-level simplification as well as personalized text simplification (Alva-Manchego et al., 2020).

Acknowledgements

We would like to thank the anonymous reviewers of the shared task for their constructive feedback which helped us improve our paper. We would also like to acknowledge The Presidency University Faculty Seed Grant Award (Ref: **ACC/26/08/2021-2**), dated Autust 26, 2021 for funding this research.

References

Sandra Aluísio, Jorge Pelizzoni, Ana Raquel Marchi, Lucélia de Oliveira, Regiana Manenti, and Vanessa Marquiafável. 2003. An account of the challenge of tagging a reference corpus for brazilian portuguese. In *International Workshop on Computational Processing of the Portuguese Language*, pages 110–117. Springer.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2020. Data-driven sentence simplification: Survey and benchmark. *Computational Linguistics*, 46(1):135–187.

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*.

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.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Nathan Hartmann, Erick Fonseca, Christopher Shulby, Marcos Treviso, Jéssica Silva, and Sandra Aluísio. 2017. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In *Proceedings of the 11th Brazilian Symposium in Information and Human Language Technology*, pages 122–131, Uberlândia, Brazil. Sociedade Brasileira de Computação.

Gondy Leroy and David Kauchak. 2014. The effect of word familiarity on actual and perceived text difficulty. *Journal of the American Medical Informatics Association*, 21(e1):e169–e172.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Gustavo Paetzold and Lucia Specia. 2016. SemEval 2016 task 11: Complex word identification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 560–569, San Diego, California. Association for Computational Linguistics.

Gustavo H Paetzold and Lucia Specia. 2017. A survey on lexical simplification. *Journal of Artificial Intelligence Research*, 60:549–593.

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*.

Matthew Shardlow. 2014. A survey of automated text simplification. *International Journal of Advanced Computer Science and Applications*, 4(1):58–70.

Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. 2020. BERTimbau: pretrained BERT models for Brazilian Portuguese. In *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 252–259.

Sanja Štajner, Daniel Ferres, Matthew Shardlow, Kai North, Marcos Zampieri, and Horacio Saggion. 2022. Lexical simplification benchmarks for English, Portuguese, and Spanish. *Frontiers in Artificial Intelligence*, 5.

Seid Muhie Yimam, Chris Biemann, Shervin Malmasi, Gustavo Paetzold, Lucia Specia, Sanja Štajner, Anaïs Tack, and Marcos Zampieri. 2018. A report on the complex word identification shared task 2018. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 66–78, New Orleans, Louisiana. Association for Computational Linguistics.