ALEXSIS+: Improving Substitute Generation and Selection for Lexical Simplification with Information Retrieval

Kai North[1], Alphaeus Dmonte[1], Tharindu Ranasinghe[2]
Matthew Shardlow[3], Marcos Zampieri[1]

[1]George Mason University, USA, [2]Aston University, UK
[3]Manchester Metropolitan University, UK

knorth8@gmu.edu

Abstract

Lexical simplification (LS) automatically replaces words that are deemed difficult to understand for a given target population with simpler alternatives, whilst preserving the meaning of the original sentence. The TSAR-2022 shared task on LS provided participants with a multilingual lexical simplification test set. It contained nearly 1,200 complex words in English, Portuguese, and Spanish and presented multiple candidate substitutions for each complex word. The competition did not make training data available; therefore, teams had to use either off-the-shelf pre-trained large language models (LLMs) or out-domain data to develop their LS systems. As such, participants were unable to fully explore the capabilities of LLMs by re-training and/or fine-tuning them on in-domain data. To address this important limitation, we present ALEXSIS+, a multilingual dataset in the aforementioned three languages, and ALEXSIS++, an English monolingual dataset that together contains more than 50,000 unique sentences retrieved from news corpora and annotated with cosine similarities to the original complex word and sentence. Using these additional contexts, we are able to generate new high-quality candidate substitutions that improve LS performance on the TSAR-2022 test set regardless of the language or model.

1 Introduction

Text simplification (TS) is utilized in educational technologies to automatically reduce the complexity of texts making them more accessible for various target populations, including children, second language learners, individuals with low-literacy, or those suffering from a reading disability, such as dyslexia or aphasia (Paetzold and Specia, 2017b; North et al., 2022c, 2023).

With an increase in online learning, there has emerged a greater need for personalized learning platforms (McCarthy et al., 2022). These educational technology platforms need to be accessible to users. TS systems provide a solution by adapting content specifically for a user's level of literacy in a given target language (Figure 1).

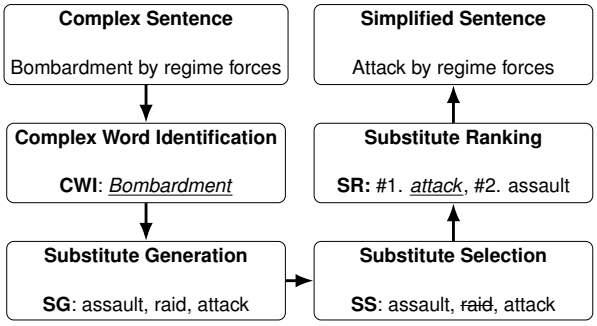

Figure 1: LS Pipeline. We only focus on SG and SS.

Lexical simplification (LS) is a precursor to TS (Paetzold and Specia, 2017b; North et al., 2022c). LS replaces challenging words, known as complex words, with simpler alternatives, hereby referred to as candidate substitutions. The generation of these candidate substitutions is known as substitute generation (SG) (Qiang et al., 2020; North et al., 2022b; Ferres and Saggion, 2022). SG attempts to predict viable candidate substitutions for an identified complex word. These candidate substitutions need to be easier to read and comprehend as well as be semantically similar to the identified complex word in its given context. An LS system would identify a complex word, for instance, "*bombardment*", as being in need of simplification. It would then suggest such words as "*attack*","*assault*" or "*raid*" as being valid candidate substitutions since they are shorter, more familiar to a set of annotators, or are found to be more frequent within a reference corpus. These candidate substitutions would then be passed to a TS system that would, in turn, simplify any unnecessary syntax resulting in an easier to read sentence.

Various methods have been applied to the task

Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), pages 404–413

of SG for LS. The use of pre-trained LLMs trained with a masked language modeling (MLM) objective is the most favored approach to this task and has been shown to outperform other methods (Saggion et al., 2022). However, the performance of MLM for SG is largely dependent on the model and the dataset it has been pre-trained on (North et al., 2022a). This hinders SG for LS, since many LS datasets contain a small number of instances or a low number of gold candidate substitutions (North et al., 2022b). As such, participants in the TSAR-2022 shared-task on LS (Saggion et al., 2022) were forced to conduct zero-shot predictions for SG due to insufficient training data.

This paper presents ALEXSIS+ and ALEXSIS++[1], two new datasets for LS. We propose an information retrieval (IR) approach that utilizes collected data from news sources. These two datasets contain 50,000 additional contexts for the original 1,500 complex words of the ALEXSIS dataset (Štajner et al., 2022), and can be used to generate accurate candidate substitutions for SG in a zero-shot condition identical to that at TSAR-2022 (Saggion et al., 2022). We demonstrate how these new datasets can be applied to any language or model without re-training or fine-tuning to increase LS performance on the TSAR-2022 test set. ALEXSIS+ and ALEXSIS++ were also constructed using only the data available to the participants of the TSAR-2022 shared-task, making our IR approach to SG, and later substitute selection (SS), highly adaptable. Furthermore, unlike ALEXSIS, which only features candidate substitutions, ALEXSIS+ and ALEXSIS++ feature multiple sentences per complex word providing new contexts that serve as useful data for MLM. Finally, as the approach doesn't require manually annotating data as in the original ALEXSIS, it can be used to improve the same unsupervised LS approaches purposed for the TSAR-2022 shared-task.

The main contributions of this paper are:

1. We propose an IR-based language independent approach to SG and SS. To the best of our knowledge, data collection efforts of this kind have not been explored within the context of LS.

2. We release ALEXSIS+ and ALEXSIS++, two new datasets for LS which open new avenues for unsupervised models with performances surpassing those reported at TSAR-2022.

3. We evaluate multiple models on the two datasets, and we discuss the results in detail.

2 Related Work

Pipeline The LS pipeline contains three subtasks (Figure 1). The first of these is SG which produces $k = n$ of candidate substitutions for a complex word with k normally being set to k = [1, 3, 5, or 10] (Paetzold and Specia, 2017b). The top candidate (k@1) is then chosen to replace the complex word. This candidate is selected through two additional sub-tasks: SS, and substitute ranking (SR). SS filters inappropriate candidate substitutions by removing candidates that are equal to or semantically dissimilar to the complex word along with those that are inappropriate in that context. SR orders a list of candidate substitutions based on their appropriateness. Techniques for SS and SR include sorting or filtering on frequency (North et al., 2022a), word length (Paetzold and Specia, 2017b), cosine similarity between word embeddings (Song et al., 2020). More recent approaches have used regression (Maddela and Xu, 2018), referred to as lexical complexity prediction (LCP) designed to replace binary CWI (North et al., 2022c), as well as prompt learning (Aumiller and Gertz, 2022). The TSAR-2022 shared-task (Saggion et al., 2022) challenged participating teams with generating a list of $k = 10$ candidate substitutions for a given complex word in English, Spanish, and Portuguese. One of TSAR's key findings is that SR is less impactful on overall LS performance compared to SG, regardless of language. Systems that relied solely on SG with minimal SR outperformed those that employed various SR methods. LS systems that relied purely on SG were often found to have used a pre-trained LLM trained with an MLM objective to generate their top-k candidate substitutions. We therefore focus on an IR approach that only improves the performance of LS through the generation and selection of additional candidate substitutions for the TSAR test set.

Masked Language Modeling MLM for LS involves feeding two concatenated sentences into an LLM separated by the [SEP] special token. The first sentence is the unaltered original sentence. The second sentence is the same as the original sentence, however, the target complex word is converted into the [MASK] special token. The LLM

[1]ALEXSIS+ and ALEXSIS++ have been made publicly available at: https://github.com/LanguageTechnologyLab/ALEXSIS2.0

	ALEXSIS			ALEXSIS+			ALEXSIS++
Languages	EN	ES	PT	EN	ES	PT	EN
Total unique complex words	386	381	386	386	381	386	386
Total unique contexts	386	381	386	12,831	13,353	13,541	33,149
Total unique candidate subs.	3,676	3,775	3,404	120,645	101,470	99,563	289,379
Avg. # of unique contexts per complex word.	1	1	1	54.60	95.90	60.15	108.18

Table 1: Comparison of the ALEXSIS, ALEXSIS+, and ALEXSIS++ datasets. Total unique candidate subs. refers to the number of unique candidate substitutions returned from generating k=10 candidate substitutions per context.

then examines both the first unaltered sentence and the words left and right of the [MASK] special token in the altered second sentence. It uses this information to predict a candidate substitution for the masked complex word. From this, an LLM is able to predict a candidate substitution that is suitable for both the provided context and for replacing the complex word. Qiang et al. (2020) was the first to apply MLM for Spanish SG. Their LSBert model surpassed all prior state-of-the-art approaches (Paetzold and Specia, 2017b), including the use of lexicon, rule-based, statistical, n-gram, and word embedding models (Paetzold and Specia, 2017b). LSBert was used as the baseline model at the TSAR-2022 shared-task (Saggion et al., 2022). Inspired by the performance of LSBert, other studies have subsequently used MLM for SG (Ferres and Saggion, 2022; North et al., 2022a; Whistely et al., 2022; Wilkens et al., 2022).

Available Resources A number of LS datasets containing complex words in context with gold candidate substitutions are available (North et al., 2022c). For English, there are LexMTurk (Horn et al., 2014) with 500 complex words, BenchLS (Paetzold and Specia, 2016a) with 929 complex words, and NNSeval (Paetzold and Specia, 2016b) with 239 complex words. For other languages, there is EASIER (Alarcón et al., 2021) with 5,310 Spanish complex words, SIMPLEX-PB (Hartmann and Aluísio, 2020) with 730 Portuguese complex words, and HanLS (Qiang et al., 2021) with 534 Chinese complex words. There are also datasets that contain a large number of complex words in context without gold candidate substitutions (Yimam et al., 2018; Maddela and Xu, 2018; Shardlow et al., 2020, 2022). The largest LS dataset that contains both context and gold candidate substitutions is ALEXSIS, referring to the combined English, Spanish (ALEXSIS-ES) (Ferres and Saggion, 2022), and Portuguese (ALEXSIS-PT) (North et al., 2022b) dataset used at the TSAR-2022 shared-task.

3 ALEXSIS+

As detailed in Section 2, MLM requires context in order to predict a suitable candidate substitution for a given complex word. Furthermore, MLM also requires a set of gold candidate substitutions to evaluate the quality of those it produces. With this in mind, we expand the ALEXSIS dataset by including a large number of unique additional contexts (Table 1). We then use these additional contexts to produce alternative candidate substitutions through MLM that differ from those generated solely on the original ALEXSIS dataset with examples of these alternative candidate substitutions being provided in Table 2. As such, we introduce ALEXSIS+ and ALEXSIS++, two large expansions of the original ALEXSIS dataset that allow for an IR approach to SG and SS, and that demonstrate how the collection of additional contexts can be used to improve LS performance under the same conditions of the TSAR-2022 shared-task (Saggion et al., 2022).

ALEXSIS+ and ALEXSIS++ were constructed using only the data made available to the participants of the TSAR-2022 shared-task (Saggion et al., 2022). We retrieve instances from the CommonCrawl News (CC-News) dataset[2] by searching for the 386 English, 381 Spanish, and 386 Portuguese complex words given to the original participants of TSAR-2022. The CC-News dataset contains crawled data from news articles all over the world. We restricted our search to news articles with domain urls that contained either one of the following: *.uk*, *.usa* or *.com* for English, *.es*, *.mx*, *.ve*, *.pes*, *.cl*, or *.ec* for Spanish, and *.pt* or *.br* for Portuguese. In this way, we reduced the likelihood of articles containing multiple languages, and we were able to make sure that each context was in the same language as the searched for complex word. Those contexts which contained a match with the original complex word were then extracted. No additional data pre-processing or cleaning was conducted on

[2]CC-News: https://data.commoncrawl.org/crawl-data/CC-NEWS/index.html

Lang.	Complex Word	Data	Type	Sentences with same Complex Word	Generated Candidate Subs. (Word.Sim)	Sent.Sim.
EN	***replica***	**A**	**Original**	The statue was moved to the Academia, Gallery and later replaced... by a replica.	**duplicate** (0.398), replacement (0.333), restoration (0.286), statue (0.426), ...	0.308
		A+	**Additional**	His project that he chose an exact day and time for the replica he created....	**copy** (0.503), version (0.322) prototype (0.518), clone (0.456), ...	
ES	***municipio***	**A**	**Original**	Cobisa es un municipio español de la.. [Cobisa is a Spanish municipality in the]...	**pueblo** (0.5143), ayuntamiento (0.750), localidad (0.691), barrio (0.561), ..	0.490
		A+	**Additional**	El tortuga reapareció en el municipio... [The turtle reappeared in the municipality]...	**pedanía** (0.542), pedanías (0.542), barriada (0.501), huerta (0.276), ...	
PT	***incremento***	**A**	**Original**	Coronel reconheceu incremento roubos... [Colonel acknowledged increased robberies]...	**crescimento** (0.856), aumento (0.878), incre (0.835), avanço (0.680), ...	0.585
		A+	**Additional**	Projetos inscritos devem... incremento... [Submitted projects must... increase]...	**ativos** (0.505), relevantes (0.541), diversos (0.407), essenciais (0.507), ...	

Table 2: Example instances including original and additional sentences (contexts) and candidate substitutions taken from the ALEXSIS (A) and ALEXSIS+ (A+) datasets. Generated candidate substitutions were produced via MLM per Section 3 with the best candidate substitution being shown in bold. Complex words are underlined and translations shown in [...]. Only snapshots of the sentences are provided. The sentence similarity (Sent.Sim) and word similarity (Word.Sim) between the additional and original sentence embeddings and the embedding of the complex word are also shown.

the extracted contexts.

ALEXSIS+ has a total of 12,831, 13,353, and 13,541 matched complex words in unique contexts for English, Spanish, and Portuguese, respectively. The larger ALEXSIS++ dataset contains matched complex words in 33,149 unique contexts only for English, including those contexts already provided by ALEXSIS+. Both datasets provide embedding similarity scores between their additional sentences and the original context (Sent.Sim), as well as between their additional candidate substitutions and the original complex word (Word.Sim). Sentence embeddings were generated using Sentence-BERT (SBert) (Reimers and Gurevych, 2019). SBert is a state-of-the-art sentence-encoder. It employs siamese and triplet network structures to produce sentence embeddings that can be used to compare the semantic similarity between sentences by calculating the cosine similarity between sentence embeddings. English word embeddings were obtained using the *en-vectors-web-lg* model that provides ∼500k word vectors. Spanish and Portuguese word embeddings were taken from the *pt-core-news-lg*, and *es-core-news-lg* models trained on crawled news articles.

Dataset Format ALEXIS+ and ALEXSIS++ are divided into three sub-corpora corresponding to the three languages, English (EN), Spanish (ES), and Portuguese (PT). Each dataset contains: original CW, context, and candidate substitutions from the TSAR-2022 shared-task, new contexts and new candidate substitutes generated on each new context, cosine similarities between new and old contexts and word similarities between word embeddings of the new candidate substitutions and the target complex word. ALEXSIS+ and ALEXSIS++ have the following nine headers separated by tab (\t):

1. **ID**: instance id that is made up of the original instance id (e.g. 01) and the new additional context id. (e.g. 104): 01-104.
2. **ALEXSIS.CW**: the original complex word taken form ALEXSIS and used at TSAR-2022.
3. **ALEXSIS.Context**: the original context for the given complex word taken form ALEXSIS and used at TSAR-2022.
4. **Candidate.Subs@n**: the candidate substitutions generated using MLM on the instances provided by TSAR-2022.
5. **Additional.Context**: new additional context obtained from the CC-News dataset.
6. **Additional.Subs@n**: new additional candidate substitutions generated using MLM on the additional contexts taken from the CC-News dataset.
7. **Sent.Sim**: the cosine similarities between the SBert sentence embedding of the additional context and the original context provided by TSAR-2022.
8. **Word.Sim**: : the cosine similarities between the word embeddings of the additional candidate substitutions and the original complex word provided by TSAR-2022.
9. **Gold.Labels**: the original gold candidate substitutions provided by TSAR-2022.

4 Approach

4.1 Substitute Generation

We experimented with three pre-trained LLMs trained with a MLM objective. Following the results of Ferres and Saggion (2022) and North et al. (2022a), we chose three monolingual rather than multilingual LLMs given their superior performance for language-specific SG (Saggion et al., 2022). We use ELECTRA (Clark et al., 2020) for English, RoBERTa-large-BNE (Fandiño et al., 2022) for Spanish, and BERTimbau (Souza et al., 2020) for Portuguese. ELECTRA was pre-trained on English Wikipedia data with a vocabulary size of 30,522 tokens. RoBERTa-large-BNE was pre-trained on the National Library of Spain corpus (Fandiño et al., 2022) that consists of 135 billion Spanish tokens scraped from Spanish websites. BERTimbau was pre-trained on the Brazilian Web as Corpus (Wagner Filho et al., 2018) that contains 2.7 billion Portuguese tokens scraped from Brazilian websites.

Figure 2 outlines our approach. We used our MLM models to generate $k = 10$ candidate substitutions for each masked complex word in context taken from the original TSAR-2022 dataset (ALEXSIS) as well as ALEXSIS+ or ALEXSIS++. Those candidate substitutes generated by the additional contexts provided by ALEXSIS+ or ALEXSIS++ were subject to several SS filters or steps. If a candidate substitution managed to pass these SS filters, then that candidate substitution would be used instead of the previous candidate substitution generated on the original ALEXSIS dataset (Figure 2). We explain each SS filter in the following section.

4.2 Substitute Selection

A total of five different SS filters were applied to the candidate substitutions generated by the additional contexts of ALEXSIS+ or ALEXSIS++. These filters were inspired by well-establish methods of SS, including the use of WordNet (Fellbaum, 2010), semantic similarity between word embeddings (EmbeddingSim) and word length (Paetzold and Specia, 2017a), as well as recent advances in deep learning, such as chain-of-thought prompting (Aumiller and Gertz, 2022; Vásquez-Rodríguez et al., 2022). These different SS filters have been used in different experimental pipeline setups, as later described in Section 4.3.

WordNet+EmbeddingSim WordNet was used to calculate the similarity between a candidate substitution and the original complex word. The returned similarity score was used alongside the cosine similarity produced by comparing the word embedding of a candidate substitution and the original complex word. These word embeddings were generated by the language models described in Section 3 and were dependent on the language. Early experiments on the ALEXSIS+ dataset were conducted to identify optimum threshold values for both word similarity metrics. Similarity values between 0.55 and 0.65 were found to produce the highest number of candidate substitutions from the additional contexts that went on to replace the original candidate substitution, regardless of language. Interestingly, WordNet's limited vocabulary was seen to aid this filtering process since out-of-vocabulary words that may have been problematic were automatically removed from the list of potential candidate substitutions.

WordFreq Zipf's Law suggests that words with lower frequency in a text tend to be longer and thus can be seen as more complex than words that appear more often and are shorter (Quijada and Medero, 2016; Desai et al., 2021). We subsequently used word frequency as a second initial SS filter during our early experiments. Those candidate substitutions which had been generated from the additional contexts more than twice passed this filter, whereas those with a generated frequency of less than two were removed.

EmbeddingSim Later SS approaches required that a greater number of candidate substitutions passed the initial filters. As such, the WordNet Lin similarity and WordFreq thresholds from our initial experiments were dropped. However, we maintained a cosine similarity of 0.5 between the word embedding of a candidate substitution and the original complex word. We named this SS filter: EmbeddingSim (EmbSim).

PromptLearning Prompt learning (PromptL) is a new state-of-the-art technique used for LS (Aumiller and Gertz, 2022; Vásquez-Rodríguez et al., 2022). It involves feeding input into a LLM, referred to as a prompt or set of prompts, that both describe the task and are worded in such a way as to elicit a desired output. For instance, we fed three prompts into a GPT-3 (Brown et al., 2020) model that were designed to identify three viable candi-

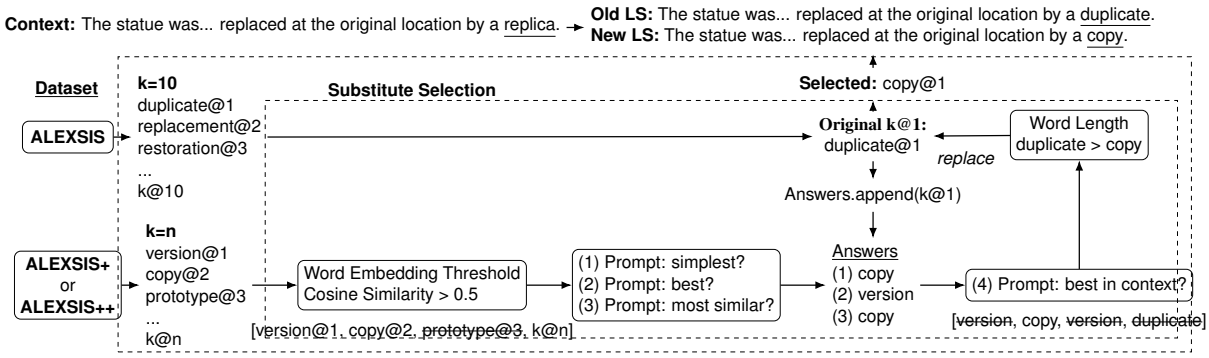

Figure 2: Our second IR approach (**pipeline b**) for LS via MLM using the ALEXSIS+ and ALEXSIS++ datasets. This approach is responsible for the results shown in Table 3. n being the number of additional candidate substitutions produced for a given a complex word from the ALEXSIS dataset using a different sentence from the ALEXSIS+ or ALEXSIS++ datasets.

date substitutions from a list of potential candidates returned from previous filters.

1. **Prompt:** What word is the *simplest* replacement for <Complex.Word> in this list?

2. **Prompt:** What word is the *best* replacement for <Complex.Word> in this list?

3. **Prompt:** What word is the *most similar* word to <Complex.Word> in this list?

The GPT-3 model then selects a maximum of one candidate substitution which best answers each of these prompts. The outputted three candidate substitutions are then appended to a new list, whereby the previous candidate substitution generated from the original ALEXSIS dataset is also appended. The model is then fed one final prompt:

4. **Prompt:** Given the *above context*, what is *the best replacement* for <Complex.Word> in this list?

This fourth prompt is able to determine out of the *simplest*, *best*, and *most similar* candidate substitution to the complex word, which is the best fit in the complex word's provided context. Through such chain-of-thought prompting, we are able to deduce the most appropriate candidate substitution for a given context and complex word from those generated from all of the additional contexts in the ALEXSIS+ and ALEXSIS++ datasets.

WordLength We used word length as an additional SS filter. This SS filter was also inspired by Zipf's Law. It was applied to the candidate substitution returned from our prompt learning SS filter. If the returned candidate substitution generated was greater in length than the original candidate substitution generated from the ALEXSIS dataset, then it is removed and the original candidate substitution is put forward. If, however, said additional candidate substitution is shorter, then it was used to replace the original candidate substitution and sent to our final filter.

BertEmbSim Our final filter used the pre-trained word embeddings from the BERT model to compute the cosine similarity of the complex word with the original candidate substitution (cos_old), and the cosine similarity of the complex word with the new candidate substitution (cos_new) generated from ALEXSIS+ or ALEXSIS++. If cos_old was greater than cos_new, and if the absolute value of the difference between the two was more than 10%, we used the original candidate substitution, else we returned the new candidate substitution.

4.3 Substitute Selection Pipeline

We experimented with three combinations of the above SS filters which resulted in three SS pipelines. These SS pipelines, (a). to (c)., are described below.

Pipeline (a). This SS pipeline was used during early experiments. Candidate substitutions produced by the additional contexts were subject to two WordNet Lin and cosine word embedding similarity thresholds both set to 0.5. Candidate substitutions that passed these threshold were then subject to a word frequency check (>2) and a word length check (<original candidate substitution) before replacing the original candidate substitution.

Pipeline (b). This SS pipeline is depicted in Figure 2. It is responsible for the results shown in

Lang.	Source	Size	ACC	MAP	POT
EN	ALEXSIS++	33,149	**0.495**	**0.495**	**0.495**
	ALEXSIS+	12,831	0.479	0.479	0.479
	ALEXSIS	374	0.484	0.484	0.484
ES	ALEXSIS+	13,353	**0.110**	**0.138**	**0.138**
	ALEXSIS	368	0.108	0.135	0.135
PT	ALEXSIS+	13,541	**0.479**	**0.489**	**0.489**
	ALEXSIS	374	0.476	0.487	0.487

Table 3: Performance of ALEXSIS, ALEXSIS+, and ALEXSIS++ when utilized by the same model for SG and evaluated on k@1 candidate substitution. Performances were evaluated on the original TSAR-2022 test set. Best performances are shown in **bold**.

Table 3. We dropped the lin similarity threshold produced by WordNet to increase the number of candidate substitutions passed to later SS filters. However, the same cosine word embedding similarity threshold of 0.5 was maintained. Additional candidate substitutions were then filtered by applying prompt learning. The first round of prompt learning reduces the list of potential candidate substitutions to three. The original candidate substitution generated from the ALEXSIS dataset is then added to this list. The last round of prompt learning selects only one out of the now four candidate substitutions. The returned candidate substitution is then subjected to a final word length check (<original candidate substitution).

Pipeline (c). After conducting the majority of our experiments, we discovered several occasions whereby the additional candidate substitution selected by our prompt learning SS filter was unsuitable for the given context (Section 5.1). To account for this, we applied an additional cosine similarity threshold between BERT produced word embeddings (BertEmbSim). All other SS filters are the same as SS pipeline (b) shown in Figure 2.

5 Evaluation

This section evaluates the performance of Electra, RoBERTa-large-BNE, and BERTimbau on the TSAR-2022 test set using our IR approach to SG and SS and the ALEXSIS+ and ALEXSIS++ datasets (Section 5.1). We also provide the performance of our various SS pipelines (Section 5.3). For the evaluation, we removed duplicate gold labels within the TSAR-2022 test set. Performances are reported in terms of accuracy (ACC), mean absolute precision, and potential following the TSAR-2022 shared-task (Saggion et al., 2022).

The performances reported at the TSAR-2022 shared-task (Section 2) show that LS is still challenging. Even small improvements in performances can lead to greater gains down-stream for TS. For this reason, LS is often primarily evaluated on the quality of the top candidate substitution produced (k@1). The accuracy of the top k@1 candidate substitution (ACC@1) is the ratio of instances whereby the best candidate generated is also the most appropriate candidate substitution among the gold labels. ACC@1 is often used to determine the overall performance of a LS system, since it is this candidate substitution which replaces the complex word. In addition, LS is also evaluated on its F1-score, potential (POT) and mean average precision (MAP). POT is the ratio of the candidate substitutions that are within all of the gold labels. MAP provides a score of the number of the returned candidate substitutions which match a gold label and its index.

5.1 ALEXSIS+ Performance

Our Spanish (RoBERTa-large-BNE) and Portuguese (BERTimbau) models benefited from the additional candidate substitutions provided by ALEXSIS+ (Table 3). RoBERTa-large-BNE's k@1 candidate substitutions increased in accuracy going from an ACC@1 score of 0.108 to 0.110, BERTimbau's final candidate substitutions saw an almost identical increase in its ACC@1, increasing from 0.476 and 0.479. However, this increase did not apply to our English (ELECTRA) model.

There are two possible causalities for this irregular improvement. The first was recognized when examining BERTimbau's MAP@3 score: 0.292, after having generated three (k@3) rather than one candidate substitution. This score is superior to that achieved by using only the original ALEXSIS dataset which obtained a MAP@3 of 0.290 when likewise generating the same number of candidate substitutions. MAP evaluates the quality of the candidate substitution produced in compari-

			Approach	Top-k=1 (@1)			Top-k=3 (@3)		
Lang.	Source	**SG**	**SS**: Step1→Step2→Step3→Step4	ACC	MAP	POT	ACC	MAP	POT
EN	**ALEXSIS++**	**MLM**	(c). EmbSim→PromptL→WordLen→BertEmbSim	**0.495**	**0.495**	**0.495**	**0.765**	**0.337**	**0.765**
			(b). EmbSim→PromptL→WordLen	**0.495**	**0.495**	**0.495**	0.757	0.329	0.757
			(a). WordNet+EmbedSim→WordFreq→WordLen	0.487	0.487	0.487	0.733	0.335	0.733
EN	**ALEXSIS**	**MLM**	None	0.484	0.484	0.484	0.738	0.336	0.738

Table 4: Shows performances of various SS approaches applied to the additional candidate substitutions generated by ALEXSIS++ and evaluated on the original TSAR-2022 test set. Best performances are shown in **bold**.

son to the gold labels as well as its positional rank (Section 5). From this, we can infer that the use of ALEXSIS+ has resulted in an original candidate substitution at rank 2 or 3 being moved to a rank 1 position. This would explain BERTimbau's improved ACC@1, since it's k@1 candidate substitution is now more aligned with the k@1 candidate substitutions within the TSAR 2022 test set's gold labels. The second feasible causality may be the fourth prompt within our prompt learning SS filter. Previously mentioned in Section 4.3, we discovered several occasions whereby the returned additional candidate substitution was unsuitable for the given context. Take the following complex word in context (a), and the simplifications produced by using the original (old) and additional (new) candidate substitutions as an example.

(a) **Complex:** "There's **conflicting** evidence about whether sick ants actually smell different from healthy ones or not."

(b) **Old LS:** "There's **mixed** evidence about whether sick ants actually smell different from healthy ones or not."

(c) **New LS:** "There's **some** evidence about whether sick ants actually smell different from healthy ones or not."

The additional candidate substitution: "*some*" returned from our prompt learning SS filter, and used in the generated (new) simplification, may be considered to be simpler in comparison to the original candidate substitution: "*mixed*". Nevertheless, in this context "*mixed*" is the more suitable candidate. This is because it is more semantically similar to the complex word "*conflicting*". GPT-3 has, therefore, failed to select the most appropriate candidate substitution after having received our fourth context orientated prompt (Section 4.3). ALEXSIS++ and the additional BERT Embedding Similarity threshold (BertEmbSim) were created to overcome this issue by either supplying more candidate substitutions or by improving the performance of our SS pipeline (b). The following sections provide model performances on ALEXSIS++ (Section 5.2) as well as performances before and after incorporating the BertEmbSim SS filter (Section 5.3).

5.2 ALEXSIS++ Performance

The additional contexts provided by ALEXSIS++ improved the quality of the candidate substitutions selected by our approach (Figure 2). These additional contexts allowed for the generation of more high quality candidate substitutions through MLM. A total of 289,379 additional candidate substitutions were provided surpassing the 120,645 produced by ALEXSIS+. As a result, increases in performances were recorded across all metrics for our English (ELECTRA) model. ACC@1, POT@1, and MAP@1 rose to 0.495 from 0.479, respectively. Despite increasing in performances being small, it is clear that the use of ALEXSIS++ is able to further increase LS performance beyond that achieved by the ALEXSIS and ALEXSIS+ datasets. It is, therefore, highly likely that the degree of improvement caused by our IR approach positively correlates with the number of additional contexts it takes into consideration going from 386 for ALEXSIS, 12,831 for ALEXSIS+, to 33,149 for ALEXSIS++. However, this positive correlation is only realized if an accurate SS pipeline is applied.

5.3 BERT Embeddings

We compared the performance of attaching the BertEmbSim SS filter to pipeline (b) against that achieved by our previous SS pipelines and LS performance without SS (Table 4). It was found that this new pipeline (c) outperformed all of our previous methods of SS for English when set to produce three candidate substitutions (k@3). The use of the BetEmbSim SS filter (c) saw an increase in ACC@3 of 0.765 from 0.757 in comparison to our previous pipeline (b). This coincided with improvements in MAP and POT scores, with a MAP@3 and POT@3 also rising to 0.337 from 0.329 and

0.765 from 0.757, respectively. In addition, having no SS filter achieved an inferior ACC@3 and POT@3 of 0.738 and 0.738, respectively, when compared to pipelines (b) and (c).

The BertEmbSim SS filter (c) was seen to produce candidate substitutions that were more suited for a complex word's context than in comparison to the previous prompt learning filter (b). This was the case for the previous example shown in Section 5.2, as the BertEmbSim SS filter was able to correctly identify "*mixed*" as being a more appropriate candidate substitution for the complex word "*conflicting*" than compared to the additional candidate substitution "*some*". In this instance, the cosine similarity between BERT word embeddings of a candidate substitution and a complex word has, thus, exceeded GPT-3's ability at determining the most appropriate replacement for a given context. This explains the superior performance of our BetEmbSim SS filter (c).

6 Conclusion and Future Work

This paper presents ALEXSIS+ and ALEXSIS++, two new version of the ALEXSIS dataset used at the TSAR-2022 shared-task (Saggion et al., 2022). These datasets contain more than 50,000 unique sentences covering three languages retrieved from news corpora and annotated with cosine similarities to the original complex word and sentence.

We have demonstrated that the use of these datasets, alongside an effective method of SS, can be used to generate and then select a more appropriate candidate substitution which, in turn, improves LS performance without the need for re-training or fine-tuning. In other words, results showed that the use of additional unique contexts can result in increases in LS performance, despite these contexts being dissimilar from the original context of the complex word. This increase in performance may appear small. However, even a small improvement in LS can have wider downstream implications that enhance the performance of a TS system substantially. We hypothesize that through further experimentation with alternative SG methods and SS filters, the performance gained by using ALEXSIS+ and ALEXSIS++ will increase. We provide these two new LS datasets and make them publicly available to the wider research community.

To the best of our knowledge, this is the first IR approach to LS opening exciting new avenues for research in this field. We show that the approach increases overall performance and that it can be applied to any LS model or language. In the future, we would like to incorporate this IR-based approach in a real-world personalized TS system that can be used in educational technology applications and online learning (McCarthy et al., 2022).

Acknowledgment

We would like to thank the creators of the ALEXSIS datasets for making the datasets available for our researcher. We further thank the anonymous BEA reviewers for their insightful feedback.

References

Rodrigo Alarcón, Lourdes Moreno, and Paloma Martínez. 2021. Lexical Simplification System to Improve Web Accessibility. *IEEE Access*.

Dennis Aumiller and Michael Gertz. 2022. UniHD at TSAR-2022 Shared Task: Is Compute All We Need for Lexical Simplification? In *Proceedings of TSAR*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, and Others. 2020. Language Models Are Few-Shot Learners. In *Proceedings of NeurIPS*.

Kevin Clark, Minh-Thang Luong, Quoc Le, and Christopher Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *Proceedings of ICLR*.

Abhinandan Desai, Kai North, Marcos Zampieri, and Christopher Homan. 2021. LCP-RIT at SemEval-2021 Task 1: Exploring Linguistic Features for Lexical Complexity Prediction. In *Proceedings of SemEval*.

Asier Gutiérrez Fandiño, Jordi Armengol Estapé, Marc Pàmies, Joan Llop Palao, Joaquin Silveira Ocampo, and Others. 2022. Maria: Spanish language models. *Procesamiento del Lenguaje Natural*, 68.

Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.

Daniel Ferres and Horacio Saggion. 2022. ALEXSIS: A dataset for lexical simplification in Spanish. In *Proceedings of LREC*.

Nathan Siegle Hartmann and Sandra Maria Aluísio. 2020. Adaptação lexical automática em textos informativos do português brasileiro para o ensino fundamental. *Linguamática*, 12(2):3–27.

Colby Horn, Cathryn Manduca, and David Kauchak. 2014. Learning a lexical simplifier using Wikipedia. In *Proceedings of ACL*.

Mounica Maddela and Wei Xu. 2018. A word-complexity lexicon and a neural readability ranking model for lexical simplification. In *Proceedings of EMNLP*.

Kathryn S McCarthy, Scott A Crossley, Kayla Meyers, Ulrich Boser, Laura K Allen, Vinay K Chaudhri, Kevyn Collins-Thompson, Sidney D'Mello, Munmun De Choudhury, Kumar Garg, et al. 2022. Toward more effective and equitable learning: Identifying barriers and solutions for the future of online education. *Technology, Mind, and Behavior*, 3(1).

Kai North, Alphaeus Dmonte, Tharindu Ranasinghe, and Marcos Zampieri. 2022a. GMU-WLV at TSAR-2022 Shared Task: Evaluating Lexical Simplification Models. In *Proceedings of TSAR*.

Kai North, Tharindu Ranasinghe, Matthew Shardlow, and Marcos Zampieri. 2023. Deep learning approaches to lexical simplification: A survey. *arXiv preprint arXiv:2305.12000*.

Kai North, Marcos Zampieri, and Tharindu Ranasinghe. 2022b. ALEXSIS-PT: A new resource for portuguese lexical simplification. In *Proceedings of COLING*.

Kai North, Marcos Zampieri, and Matthew Shardlow. 2022c. Lexical Complexity Prediction: An Overview. *ACM Computing Surveys*.

Gustavo Paetzold and Lucia Specia. 2017a. Lexical simplification with neural ranking. In *Proceedings of ACL*.

Gustavo H. Paetzold and Lucia Specia. 2017b. A Survey on Lexical Simplification. *Journal of Artificial Intelligence Research*, 60(1):549–593.

Gustavo Henrique Paetzold and Lucia Specia. 2016a. Benchmarking lexical simplification systems. In *Proceedings of LREC*.

Gustavo Henrique Paetzold and Lucia Specia. 2016b. Unsupervised lexical simplification for non-native speakers. In *Proceedings of AAAI*.

Jipeng Qiang, Yun Li, Zhu Yi, Yunhao Yuan, and Xindong Wu. 2020. Lexical simplification with pretrained encoders. In *Proceedings of AAAI*.

Jipeng Qiang, Xinyu Lu, Yun Li, Yunhao Yuan, and Xindong Wu. 2021. Chinese Lexical Simplification. *IEEE Press*, 29:1819–1828.

Maury Quijada and Julie Medero. 2016. HMC at SemEval-2016 Task 11: Identifying Complex Words Using Depth-limited Decision Trees. In *Proceedings of SemEval*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of EMNLP-IJCNLP*.

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*.

Matthew Shardlow, Michael Cooper, and Marcos Zampieri. 2020. CompLex — a new corpus for lexical complexity prediction from Likert Scale data. In *Proceedings of READI*.

Matthew Shardlow, Richard Evans, and Marcos Zampieri. 2022. Predicting lexical complexity in english texts: the complex 2.0 dataset. *Language Resources and Evaluation*, 56:1153–1194.

Jiayin Song, Jingyue Hu, Leung-Pun Wong, Lap-Kei Lee, and Tianyong Hao. 2020. A New Context-Aware Method Based on Hybrid Ranking for Community-Oriented Lexical Simplification. In *Proceedings of DASFAA*.

Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. 2020. BERTimbau: pretrained BERT models for Brazilian Portuguese. In *Proceedings of BRACIS*.

Sanja Štajner, Daniel Ferrés, Matthew Shardlow, Kai North, Marcos Zampieri, and Horacio Saggion. 2022. Lexical Simplification Benchmarks for English, Portuguese, and Spanish. *Frontiers in Artificial Intelligence*.

Laura Vásquez-Rodríguez, Nhung Nguyen, Sophia Ananiadou, and Matthew Shardlow. 2022. UoM&MMU at TSAR-2022 Shared Task: Prompt Learning for Lexical Simplification. In *Proceedings of TSAR*.

Jorge A. Wagner Filho, Rodrigo Wilkens, Marco Idiart, and Aline Villavicencio. 2018. The brWaC corpus: A new open resource for Brazilian Portuguese. In *Proceedings of LREC*.

Peniel John Whistely, Sandeep Mathias, and Galiveeti Poornima. 2022. PresiUniv at TSAR-2022 Shared Task: Generation and Ranking of Simplification Substitutes of Complex Words in Multiple Languages. In *Proceedings of TSAR*.

Rodrigo Wilkens, David Alfter, Rémi Cardon, Isabelle Gribomont, and Others. 2022. CENTAL at TSAR-2022 Shared Task: How Does Context Impact BERT-Generated Substitutions for Lexical Simplification? In *Proceedings of TSAR*.

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 BEA*.