LeSS: A Computationally-Light Lexical Simplifier for Spanish

Sanja Štajner[1], Daniel Ibáñez[2], Horacio Saggion[3]
[1]Karlsruhe, Germany, stajner.sanja@gmail.com
[2]Sevilla, Spain, danielibanezgarcia@gmail.com
[3]LaSTUS Lab / TALN Group, Universitat Pompeu Fabra, Spain, horacio.saggion@upf.edu

Abstract

Due to having knowledge of only basic vocabulary, many people cannot understand up-to-date written information and thus make informed decisions and fully participate in the society. We propose LeSS, a modular lexical simplification architecture that outperforms state-of-the-art lexical simplification systems for Spanish. In addition to its state-of-the-art performance, LeSS is computationally light, using much less disk space, CPU and GPU, and having faster loading and execution time than the transformer-based lexical simplification models which are predominant in the field.

1 Introduction

Even in the highly-developed countries, many people (16.7% on average) only have a knowledge of a basic vocabulary thus encountering difficulties in understanding written information on a daily basis (OECD, 2013). This limits their active participation in the society and can negatively influence their life choices. According to the Adult Literacy Report from 2013 (OECD, 2013), this problem is particularly prominent in Spain, where 28.3% of people are in this situation (Štajner, 2021).

Lexical simplification (LS) is the process of substituting complex words or phrases with their simpler variants. It is an important factor in making texts more accessible for people with aphasia (Carroll et al., 1998; Devlin and Unthank, 2006; Devlin and Tait, 1998), dyslexia (Rello et al., 2013b,a), autism spectrum disorders (Orăsan et al., 2018), cognitive impairments (Feng et al., 2009; Saggion et al., 2015), low literacy levels (Aluísio et al., 2008; Watanabe et al., 2010), deaf and hard-of-hearing people (Inui et al., 2003; Alonzo et al., 2020), children (De Belder et al., 2010), and non-native speakers (Hirsh and Nation, 1992; Heilman et al., 2007).

Due to its evident potential for great social impact, lexical simplification has been attracting a growing attention from the natural language processing (NLP) community (Paetzold and Specia, 2017; Štajner, 2021). SemEval-2021 Task 1 on Lexical Complexity Prediction attracted 198 teams, out of which 91 submitted their systems for one of the two sub-tasks, single-word or multi-word lexical complexity prediction (Shardlow et al., 2021). The recent TSAR-2022 shared task on Multilingual Lexical Simplification received 33 system submissions for English, 17 for Spanish, and 16 for (Brazilian) Portuguese (Saggion et al., 2022).

While it is generally considered that the more frequent words are easier to understand for everyone, which words should be considered as complex or simple can vary from one target population to another (Yimam et al., 2017), and is subjective even within one target group (Yimam et al., 2018). Manual lexical simplification requires an extensive knowledge of the language and the particular simplification needs of the target user, thus being expensive and time-consuming. Automatic text simplification systems, in contrast, could offer a possibility for an easier customisation and on-demand personalized simplification. To enable that, it is important to build modular systems which offer possibility of using customized resources and substitute ranking modules.

We propose LeSS, a new state-of-the-art lexical simplifier for Spanish, that uses less computational power than the previous state of the art and a modular architecture that enables easy customization.[1] As it will be shown in Section 5, LeSS outperforms the transformer-based state-of-the-art lexical simplification systems for Spanish, while being computationally much more efficient.

[1]The full code for the system is available at: https://github.com/danielibanezgarcia/less.

Proceedings of Recent Advances in Natural Language Processing, pages 1132–1142
Varna, Sep 4–6, 2023
https://doi.org/10.26615/978-954-452-092-2_120

Work	Substitutes Generation	Candidate Ranking
(Bott et al., 2012)	word vector model, thesaurus	word frequency, word length
(Baeza-Yates et al., 2015)	Google Books Ngrams, thesaurus	web frequencies
(Ferrés et al., 2017)	word vector model, thesaurus	word frequency
(Alarcón et al., 2021)	word2vec, sense2vec, FastText, BERT	word frequency, BERT prediction, semantic similarity
(Ferrés and Saggion, 2022)	thesaurus, MLM	word frequency, MLM probability
(Štajner et al., 2022)	MLM	MLM probability
(Whistely et al., 2022)	MLM	cosine similarity, POS check
(Vásquez-Rodríguez et al., 2022)	LM with prompt	fined-tuned BERT model as classifier
(Chersoni and Hsu, 2022)	MLM	MLM-, GPT-2- and sentence probability, cosine similarity
(Wilkens et al., 2022)	MLM	word frequency, binary classifier
(North et al., 2022)	MLM	MLM probability, Zipf word frequency

Table 1: Overview of approaches used for substitutes generation and for candidate ranking in lexical simplification systems for Spanish (MLM = Masked Language Model; LM = Language Model; POS = Part-of-Speech).

2 Related Work

Apart from English, Spanish is the language that attracted most attention from the lexical simplification research community.

2.1 Evaluation Datasets

Only three evaluation datasets for Spanish lexical simplification were compiled and made publicly available so far.

EASIER-500[2] (Alarcón et al., 2021) consists of 500 instances with exactly one target complex word in each, and three simpler synonyms for each target word. Being the first publicly released lexical simplification dataset for Spanish, EASIER-500 has several limitations that were addressed in the later compiled datasets: (1) target words were selected based on the assessments of only one (expert) annotator; (2) each instance contains only three simpler synonyms for the target complex word; (3) all simpler synonyms were suggested by only one annotator; (4) it does not provide ranking of the simpler synonyms (and is thus not suitable for evaluation of full lexical simplification pipelines).[3]

EASIER[4] (Alarcon et al., 2023) consists of 5100 complex/target words in context (sentence) for which at least one simpler synonym was proposed (7892 simpler synonyms in total) by a linguist. The strength of this corpus is that the quality of the annotations (selection of complex words and the suggested simpler synonyms) was assessed by elderly people and people with intellectual disabilities. The limitations of this dataset are the following: (1) for most target words, only one simpler synonym is proposed; (2) for any target word, only up to three simpler synonyms were proposed; (3) it does not provide the ranking of the simpler synonyms (in the cases where more than one simpler synonym was proposed).

ALEXSIS (Ferrés and Saggion, 2022) was the first dataset for evaluation of full lexical simpflication pipelines for Spanish. It consists of 381 instances/contexts, each with one target word marked as complex, and a list of simpler (near-)synonyms of the given target word. For each instance, the corresponding simpler synonyms were proposed by 25 crowdsourced workers. The subset of 368 instances of this dataset was used in the TSAR-2022 shared task on multilingual lexical simplification (Saggion et al., 2022), with only a few slight modifications described in the work by Štajner et al. (2022).

2.2 Lexical Simplification Systems

The earliest approaches for Spanish lexical simplification relied on thesauri for generating potential substitutes, while since 2022, all proposed approaches are based on the use of the transformer-based masked language models (see Table 1).

Bott et al. (2012) built LexSiS, a lexical simplification system that uses an online dictionary and Web as a corpus to compute three features (word vector model, word frequency, and word length) for finding the best substitution candidates, and a combination of hand-crafted rules and dictionary look-up for morphological generation of the right inflection for the best substitute.

[2] https://data.mendeley.com/datasets/ywhmbnzvmx/2

[3] A full lexical simplification pipeline usually consists of four modules that perform the following operations: complex word identification, generation of substitution candidates, ranking of the substitution candidates, generation of the correct inflections (Shardlow, 2014; Paetzold and Specia, 2017).

[4] https://github.com/LURMORENO/EASIER_CORPUS

Baeza-Yates et al. (2015) proposed CASSA, an approach that uses Google Books Ngram Corpus, the Spanish OpenThesaurus, and web frequencies for finding the best substitution candidates. CASSA does not offer the full lexical simplification pipeline, as it only finds the best lemma and does not perform morphological generation of the right inflection.

Ferrés et al. (2017) proposed TUNER, a lexical simplifier for Spanish, Portuguese, Catalan, and Galician which simplifies content words (common nouns, verbs, adjectives, and adverbs) in context. It consists of six modules that are sequentially executed: complex word identification, document analysis, word sense disambiguation (WDS), synonyms ranking, morphological generation, and context adaptation.

Alarcón et al. (2021) experimented with several neural LS systems for Spanish, which leverage pre-trained word embedding vectors and BERT models. The systems were evaluated on the EASIER-500 dataset for only three lexical simplification subtasks: complex word identification, substitution generation, and substitution selection. The ranking of substitutes was not evaluated as the EASIER-500 dataset does not provide rankings of the substitutes (Alarcón et al., 2021).

Ferrés and Saggion (2022) compared results of three architectures for LS (thesaurus-based TUNER system, a single transformer-based system, and several combinations of transformer-based systems) on ALEXSIS dataset.

Štajner et al. (2022) built LSBert-ES, the Spanish version of the state-of-the-art LS system for English – LSBert (Qiang et al., 2020). They compared the performances of LSBert-ES and TUNER on ALEXSIS dataset. Both systems were used as strong baselines for the TSAR-2022 shared task.

Whistely et al. (2022) built the winning system of the TSAR-2022 shared task on lexical simplification in Spanish (Saggion et al., 2022). The system generates substitution candidates by using a masked language model BETO (Cañete et al., 2020), ranks the candidates based on the cosine similarity of their word embeddings with the word embeddings of the target word (using FastText (Grave et al., 2018)), and filters out candidates that do not share the same Part-of-Speech (PoS) tag as the target word (using Stanford PoS tagger (Toutanova et al., 2003)).

Vásquez-Rodríguez et al. (2022) experimented with pre-trained language models in three settings: zero-shot, fine-tuned (using language-specific data), and multilingual (pre-trained multilingual LM fine-tuned in an specific language), using two different prompts. Their best system (fine-tuned language model) was ranked second in the TSAR-2022 shared task (Saggion et al., 2022).

Chersoni and Hsu (2022) participated in the TSAR-2022 shared task with three fully unsupervised LS systems in which substitution candidates are retrieved by using masked language model, and then ranked based on the lowest average rank across three transformer-based metrics: sentence probability via autoregressive language modeling; sentence probability via masked language modeling; and contextualized embedding similarity.

Wilkens et al. (2022) participated in the TSAR-2022 shared task with BERT-based approaches. They explored two strategies for using masked language models for candidate generation in Spanish: Copy and Query Expansion. The Copy strategy follows the strategy used in LSBert, while the Query Expansion strategy extracts alternative words for the target words from FastText embeddings and then replaces the original sentence with each alternative word. They also experimented with various approaches for candidate ranking: voting (most frequently proposed candidate by various candidate generation methods), probabilities of character-based n-gram language models, binary classifier trained on English SemEval data for simplicity ranking (Specia et al., 2012).

North et al. (2022) participated in the TSAR-2022 shared task with the system that uses a masked language model for substitute generation and the Zipf frequency for substitute ranking.

2.3 State of the Art

The current state-of-the-art lexical simplification systems for Spanish are the transformer-based systems proposed by Ferrés and Saggion (2022) and by Whistely et al. (2022). The former achieves the best results on the ALEXSIS dataset, while the latter achieves the best results on the TSAR-2022 dataset. As it will be shown in Section 5, our word-embedding-based LS system (LeSS) outperforms those (computationally much more expensive) systems on both datasets.

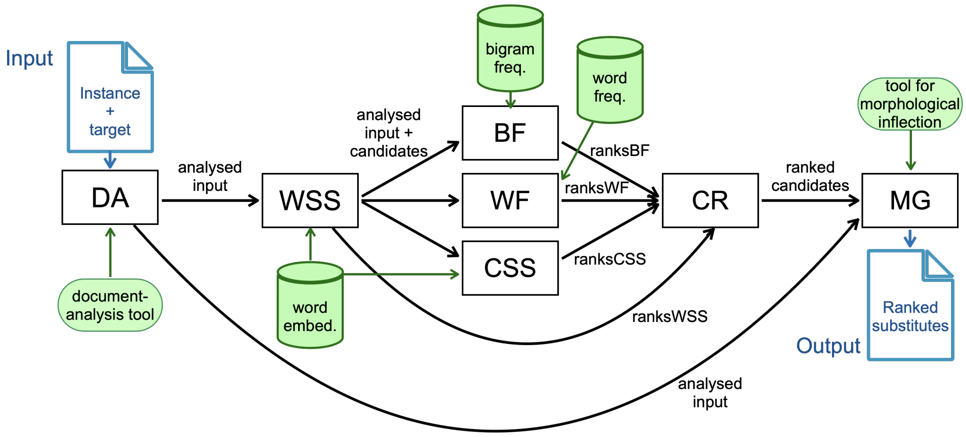

Figure 1: Schema of the system architecture. The seven modules (DA, WSS, BF, WF, CSS, CR, and MG) are shown in rectangular fields, while the language-dependant tools and resources are shown in green/oval shapes.

3 Architecture of LeSS

LeSS is a computationally light lexical simplifier with modular architecture (Figure 1). It comprises seven modules: document analyzer (DA), word-level semantic similarity (WSS), context-level semantic similarity (CSS), word frequency (WF), bigram frequency (BF), candidate ranking (CR), and morphological generator (MG).

Document analyzer (DA) performs sentence splitting, tokenization, part-of-speech (PoS) tagging, lemmatization, and named entity recognition.

Word-level semantic similarity (WSS) module retrieves as substitution candidates those words whose word embedding vectors have the highest cosine similarity with the word embedding vectors of the target word. It initially selects 30 candidates. From those 30, it filters out those that share the lemma with the target word, and those that contain more than 95% of non-alphabetic characters. When multiple candidates share the lemma among themselves (but not with the target word), only the candidate whose word embedding vector has the highest cosine similarity to the word embedding vector of the target word is retained.

Context-level semantic similarity (CSS) module computes cosine similarities between the word embedding vector of the substitution candidate and the context of the target word. This module is envisioned as word sense disambiguation tool. It has been proposed by Glavaš and Štajner (2015) with the idea that the simplification candidates which are synonyms of the correct sense of the target word should be more semantically similar to the context of the target word. The context-level semantic similarity ($csim$) between the target word t and the replacement candidate r is obtained by averaging the cosine similarity between the word embedding vector of the replacement candidate (v_r) and word embedding vectors of each content word (v_w) in the context of the target word (C_t), using the following formula:

$$csim(t,r) = \frac{1}{|C_t|}\sum_{w \in C_t} \cos(v_r, v_w) \quad (1)$$

where the context is the symmetric window of 2 words left and 2 words right from the target word.

Word frequency (WF) module retrieves the frequency of the target word and the candidate replacements in large corpora. Based on the intuition that frequent words are usually simpler to understand, word frequency is often used for the ranking of the substitution candidates in LS systems (Paetzold and Specia, 2017).

Bigram frequency (BF) module returns the average value (arithmetic mean) of the Google Books bigram frequencies for the bigrams $w_{-1}r$ and rw_{+1}, where r is the replacement candidate, w_{-1} is the word preceding the target word, and w_{+1} is the word after the target word in the given sentence. If the target word is the first word in the sentence, the module returns the frequency of the bigram rw_{+1}. If the target word is the last word in the sentence, the module returns the frequency of the bigram $w_{-1}r$. The idea behind this module is that the frequency of the word itself is not always a straightforward measure of its simplicity. The bigram frequency is envisioned to capture the influence of the surrounding words on the word simplicity, which is particularly important in the case of phrasal verbs or multi word expressions. How well the replacement candidate fits in a larger context should be captured by word-level and context-level semantic similarity modules.

Candidate ranking (CR) module computes the final ranking of all replacement candidates, including the target word itself. For each word, it first sums the ranks obtained in separate modules (WSS, BF, WF, and CSS). Then, it ranks the candidates based on those sums. When the value for a candidate replacement cannot be calculated in a certain module, e.g. the word does not appear in pre-trained word embedding model or frequency library, that candidate will receive the rank '10000' in that module, as a penalty for being infrequent, and as such, probably complex.

Morphological generator (MG) module returns the inflected form of the replacement candidate (with the same PoS tag, gender, and number as the target word) given the lemma of the replacement candidate and the PoS tag of the target word.

Here is important to note that LeSS does not explicitly perform complex word identification. It, instead, follows the idea proposed by Glavaš and Štajner (2015) to treat all content words as potentially complex. The complex word identification is, in that case, performed implicitly, by the target word itself being considered as a substitution candidate (together with the 'real' substitution candidates) in the candidate ranking module.

4 Tools and Resources Used in LeSS

For document analysis (DA module), we use FreeLing (Padró and Stanilovsky, 2012) v4.0.[5].

For all operations with word embedding vectors (modules WSS and CSS), we use FastText 2M 300-dimensional cased word embeddings.[6]

Word frequencies (module WF) are obtained using the freely available python *wordfreq* library,[7] which contains word frequencies calculated on the Exquisite Corpus.[8] The Spanish part of this corpus comprises encyclopedic texts (Wikipedia), subtitles (OPUS Open Subtitles and SUBTLEX), news (NewsCrawl 2014 and GlobalVoices), books (Google Books Ngrams 2012), web texts (OSCAR), short-form social media texts (Twitter), and longer-form internet comments (Reddit).

Freeling lexicon		J48 training data	
#lemmas	#forms	corpus	#tokens
70,150	669,216	CoNLL09	427,442

Table 2: Data statistics for the Morphological Generator.

Algorithm	Noun	Verb	Adj	Adv
FreeLing	72.60	95.03	76.21	72.89
J48	99.80	94.32	99.24	98.51
FreeLing+J48	**99.84**	**95.77**	**99.44**	**98.57**

Table 3: Accuracy (%) of different configurations of Morphological Generator. For the configuration that uses only the Freeling lexicon, the results present coverage as the lexicon cannot predict the results for the unseen *(lemma,PoS)* pairs.

For calculating bigram frequences (needed for BF module), we use Google Books Ngrams for Spanish.[9] We store pre-calculated bigram frequences in a look-up table and use it for retrieving particular bigram frequencies at the execution time. The range of years used to create the table was [1990, 2019], where all bigrams containing numbers were removed.

As MG module, we use the morphological generator proposed by Ferrés et al. (2017) which combines lexicon-based generation with predictions from decision trees. The lexical categories supported are: verbs, nouns, adjectives, adverbs, pronouns, determiners, and numerals. The lexicon used is the FreeLing v4.0 morphological dictionary for Spanish. When the lexicon has no inflection for a pair *(lemma, PoS tag)*, the module uses the J48 model (WEKA[10] implementation of C4.5 decision tree) to predict the sequence of edit operations that can transform an unseen pair *(lemma, PoS tag)* to the correct inflected form. Table 2 shows data statistics of this module. The J48 training algorithm uses morphological and lemma-based features, including the Levenshtein edit distance between lemmas and word forms, to create a model for each lexical category (Ferrés et al., 2017). The model was trained on the Spanish training dataset from the CoNLL-2009 shared task,[11] and evaluated using the CoNLL-2009 shared task evaluation dataset for Spanish which consists of 50,635 to-

[5] https://nlp.lsi.upc.edu/freeling/

[6] https://dl.fbaipublicfiles.com/fasttext/vectors-crawl/cc.es.300.vec.gz

[7] https://pypi.org/project/wordfreq/

[8] https://github.com/LuminosoInsight/exquisite-corpus

[9] http://storage.googleapis.com/books/ngrams/books/20200217/spa/spa-2-ngrams_exports.html

[10] http://www.cs.waikato.ac.nz/~ml/weka/

[11] http://ufal.mff.cuni.cz/conll2009-st/

System	MAP				Potential			Accuracy		
	@1	@3	@5	@10	@3	@5	@10	@1@top1	@2@top1	@3@top1
LeSS	**41.6**	**25.7**	**18.3**	**10.6**	**62.5**	**71.2**	**75.8**	19.3	**27.7**	**34.8**
TSAR-2022 best: PresiUniv run 1	36.9	21.4	15.0	8.3	58.4	64.7	72.5	**20.4**	**27.7**	32.9
TSAR-2022: UoM&MMU run 3	36.7	21.3	15.1	9.0	53.3	60.0	69.3	16.0	22.8	26.9
TSAR-2022: PresiUniv run 3	36.1	19.4	13.2	7.1	51.6	55.4	58.1	20.4	25.8	29.6
TSAR-2022: UoM&MMU run 2	36.1	22.2	16.6	9.6	53.8	61.7	70.1	16.0	24.4	29.1
TSAR-2022: PolyU-CBS run 3	35.9	20.1	14.6	8.5	52.4	59.8	67.9	16.3	20.1	23.6
LSBert-baseline	28.8	18.7	13.5	7.9	49.4	61.1	74.7	9.5	14.4	18.2
TUNER-baseline	11.9	5.7	3.6	1.8	14.4	14.5	15.0	6.2	7.8	8.4

Table 4: Evaluation results on TSAR-2022 shared task test set for Spanish for our system (LeSS), the five best performing systems at the shared task (proposed by the teams PresiUniv (Whistely et al., 2022), UoM&MMU (Vásquez-Rodríguez et al., 2022), and PolyU-CBS (Chersoni and Hsu, 2022)), and the official baselines of the shared task (LSBert-baseline and TUNER-baseline) according to the official results (Saggion et al., 2022). The highest value for each metric is shown in bold.

kens. The configuration that uses both, FreeLing and J48, achieves an accuracy over, or close to, 99% in almost all cases, with the exception of the verbs (Table 3). Further details regarding morphological generator can be found in (Ferrés et al., 2017).

5 Evaluation

To compare the performance of our system with the state of the art, we evaluate it on the Spanish portion of the TSAR-2022 shared task dataset, and the ALEXSIS dataset.

5.1 Evaluation on TSAR-2022 Shared Task

To be able to compare the results with the systems submitted to the TSAR-2022 shared task for Spanish, we evaluate our system on the official test set (containing a subset of 368 instances from ALEXSIS dataset) using the official evaluation metrics (MAP@k, Potential@k, and Accuracy@n@top1, where $k \in \{1, 3, 5, 10\}$ and $n \in \{1, 2, 3\}$).[12] **MAP@k** uses a ranked list of generated substitutes, where each substitute can be matched or not matched against the set of the gold-standard substitutes. Unlike the commonly used Precision metric that only measures how many of the generated substitutes are correct (i.e. found in gold data), MAP@k additionally takes into account the ranks of the correct substitutes, i.e. it rewards systems where correct substitutes are ranked higher than the incorrect ones. **Potential@k** calculates the percentage of instances for which at least one of the k best-ranked generated substitutions is present in gold standard.[13] **Accuracy@n@top1** calculates the percentage of instances where at least one of the n top-ranked generated substitutes matches the most frequently suggested synonym in the gold standard for that instance. For all three metrics, higher scores indicate better LS systems.

As can be seen in Table 4, our LeSS system noticeably outperforms the winner of the shared task on all but one metric (accuracy@1@top1). Moreover, LeSS achieves higher results than any participating system on all but two metrics: accuracy@1@top1 and potential@10 (see the full table of official results in (Saggion et al., 2022)). All systems that participated in the TSAR-2022 shared task for Spanish used approaches based on LSBert, except for the system proposed by Vásquez-Rodríguez et al. (2022) which uses GPT-2. In addition to its better performances on the shared task dataset, LeSS requires much less computational power than LSBert (see Section 5.3, Table 6), and thus also less computational power than all the systems that participated in the shared task.

5.2 Comparison with ALEXSIS Systems

To compare our systems with the state-of-the-art LS systems proposed by Ferrés and Saggion (2022), which are not publicly available yet, we evaluate our system also on the full ALEXSIS dataset (381 instances) using the metrics used for the evaluation of those systems: Precision, Accuracy, and Change. We use the definitions provided by Ferrés and Saggion (2022) to compute those metrics for LeSS:

[12]The test set (with and without gold annotations) and evaluation script are freely available at the shared task GitHub account: https://github.com/LaSTUS-TALN-UPF/TSAR-2022-Shared-Task

[13]MAP@1 and Potential@1 are equal by definition.

System	Precision	Accuracy	Change
LeSS	**0.606**	**0.491**	0.701
Thesaurus	0.889	0.089	0.199
LSBert-ES (BETO)	0.278	0.278	1.000
SpanBERTa ∩ RbaseBNE	0.475	0.461	0.986
SpanBERTa ∪ RbaseBNE	0.469	0.469	1.000

Table 5: Performances on ALEXSIS dataset. The results for the last four systems are taken from Table 9 in (Ferrés and Saggion, 2022).

Precision is the ratio of instances where the top ranked candidate is either the target word itself or a word present in the gold standard; **Accuracy** is the ratio of instances where the top ranked candidate is in the gold standard;[14] and **Change** is the ratio of instances where the system suggested any word different from the target word (regardless of whether it is found in the gold standard list or not).

Table 5 shows the performances on the full ALEXSIS dataset of our LeSS system, and the four systems proposed by Ferrés and Saggion (2022): *Thesaurus* (where the substitution candidates are generated based on a thesaurus), *LSBert-ES (BETO)* (a transformer-based LS system) and the two best-performing systems (combinations of transformer-based LS models) which were considered the state of the art on the ALEXSIS dataset. Two things should be noted when interpreting those results. First, by definition, Change is not a measure of how well the system performs lexical simplification, but rather a measure of how conservative it is, i.e. how often it leaves the target word unchanged. Second, Precision is a valuable measure for evaluation of fully automatic lexical simplification systems, for which it is important that they do not perform incorrect substitutions, i.e. leaving the target word unchanged is better than replacing it with an incorrect substitute. As can be seen in Table 5, LeSS outperforms the state-of-the-art systems on both Precision and Accuracy metrics. The thesaurus-based system has a higher Precision than LeSS, but at the cost of a very low Accuracy.

5.3 Computational Power

The comparison of the disk, CPU, and GPU usage by LeSS and LSBert, as well as the loading and execution time, are presented in Table 6. As can be seen, LeSS has a significantly lower load and processing times, and it requires much less disk, CPU, and GPU usage than LSBert. As LSBert is the basis of all recently proposed LS systems for Spanish mentioned in Section 2, which participated in TSAR-2022 shared task, one can infer that LeSS is computationally lighter than all those systems.

[14] By definition, Accuracy is the same as MAP@1 metric used in TSAR-2022 shared task.

		LeSS	LSBert
Size	Disk	4960.19MB	17212.81MB
	CPU	7540.00MB	12505.00MB
	GPU	0.00MB	1530.00MB
Time	Load	0:01:54sec	0:03:38sec
	Processing	0:00:49sec	0:02:24sec

Table 6: Statistics of computing power necessary for running the systems on ten instances using the machine with the following specifications: Processor: Intel Core i9-9900KF CPU @ 3.60GHz x 16; RAM: 32GB, GPU: NVIDIA GeForce RTX 2080 Ti/PCIe/SSE2; Hard Drive: ADATA SX6000PNP (1TB).

6 Error Analysis

We performed error analysis on the full ALEXSIS dataset (381 instances). In 29 instances (7.6%), none of the substitutes generated by LeSS is present in the gold data. In four of those 29 cases, LeSS did not generate any substitutes, as the word embeddings used did not contain those four target words: *pitorreo* (*eng. messing around*) – used only in colloquial jargon in Spain; *expiaciones* (*eng. atonement*) – used in biblical sense; *pedanía* (*eng. district*) – used only for special types of districts in Spain; and *larvas* (*eng. larva*) – used in quotes in a metaphorical sense. In 18 of those 29 cases, LeSS suggested the target word itself as the best-ranked replacement candidate. In a real-world scenario, those 22 cases would limit the simplification power of the system but would not be dangerous as they would leave the target word unchanged. In two of the 29 cases where the candidates suggested by LeSS were not found in the gold data, target word itself was suggested by LeSS as the second-best. In another two, LeSS suggested a correct word but with missing reflexive pronoun: *preparando* instead of *preparandose* (*eng. getting ready*), and *forjar* instead of *forjarse* (*eng. to forge oneself* (figuratively)). In one case, LeSS suggested the words *peligroso* (*eng. dangerous*) and *destructivo* (*eng. destructive*), which were not found in the gold standard but fit the context perfectly, as a simpler substitute for *mortífero* (*eng. lethal*). In an-

(1)	Sentence	A lo largo de sus más de veinte años de experiencia en el medio, ha presentado todo tipo de programas, no sólo informativos, sino también divulgativos, de entrevistas, tertulias e incluso concursos.
	LeSS	**reuniones**, **charlas**, **conversaciones**, tertulias, conferencias
	Gold	**charlas(7)**, **reuniones(6)**, debates(4), **conversaciones(2)**, reunión(2), conversación, reunion, fiestas, coloquios
(2)	Sentence	A pesar de las pocas bajas (menos de 500 en total) y de los inconclusos resultados tácticos, Valmy fue considerada como una de las quince batallas decisivas del mundo, porque una derrota francesa hubiera propiciado la decadencia de la Revolución francesa.
	LeSS	**provocado**, llevado, producido, propiciado, **favorecido**
	Gold	**provocado(5)**, **favorecido(3)**, desencadenado(3), causado(2), favorido, terminado en, ayudado a, facilitado, hecho posible, ayudado, ocasionado, incitado, permitido, fomentando, predispuesto
(3)	Sentence	A comienzos de la década de 1980, se trasladó a Los Ángeles, en California, donde comenzó a labrarse una reputación con sus actuaciones, tanto eléctricas como acústicas.
	LeSS	forjar, consolidar, establecer, formar, buscar
	Gold	formarse(6), construirse(4), hacerse(4), forjarse(2), trabajarse, ganarse, hacerce, crearse, trabajar, ganarse, cultivar, prepararse
(4)	Sentence	Al igual que otros municipios cercanos a Toledo, la población se originó a partir de los caseríos que utilizaban los vecinos de la capital en las épocas de labor.
	LeSS	*pueblos*, poblados, pobladores, *barrios*, parajes
	Gold	casas(5), aldeas(3), las casas(2), hogares(2), casales, trabajos, pueblitos, burgos, domésticos, caseríos, casar, viviendas, casonas, domicilios, vecindario, métodos, lugarejos
(5)	Sentence	Cuanto a los artistas, los únicos que resisten a la compresión y preservan sus personalidades, conocen una tragedia propia: el ideal estético es mortífero, como lo prueba el suicidio del pintor Lucien, inspirado en Vincent Van Gogh, que Mirbeau acaba de descubrir.
	LeSS	*peligroso*, violento, poderoso, *destructivo*, sangriento
	Gold	mortal(12), letal(10), de muerte, fatal, lúgubre

Table 7: Examples of LeSS output (five top-ranked substitution candidates) for ALEXSIS instances (target words are underlined). The number in parenthesis after the word in 'Gold" (standard) represents the number of workers that suggested that word, if the word was suggested by more than one annotator. The substitutes shared between LeSS and human annotators are shown in bold. The correct substitutes generated by LeSS which are not found in gold standard are shown in italics.

other case, LeSS suggested the word *pueblos* (*eng. villages*) which was found in gold standard only in its diminutive form *pueblitos*.

Table 7 presents several instances from ALEXSIS dataset, together with the output of LeSS system and the gold standard annotations. The first two examples show that LeSS is able to find correct simpler synonyms, which are also suggested by several crowdsourced annotators. The last three examples illustrate the errors mentioned in the previous paragraph: where LeSS suggests a correct verb but without reflexive pronoun (ex. 3); where it suggests correct nouns which are not found among the gold standard annotations (ex. 4); and where it (over)simplifies an adjective (ex. 5).[15]

7 Final Discussion and Conclusions

We proposed LeSS, a modular and computationally-light lexical simplifier for Spanish that outperforms the previous state of the art.

Our detailed manual error analysis indicated that LeSS often suggests several simpler synonyms. This indicates that LeSS could be used in real-world applications as a writing aid to human editors for faster simplification and customization to different users. In real-world applications, the use of lexical simplification module as a writing aid is preferred over fully automatic lexical simplification, as it allows for customization (Orăsan et al., 2018; Alonzo et al., 2020) and prevents unintended harms to vulnerable populations (Štajner, 2021).

In future, we would like to investigate if this architecture can yield state-of-the-art results in other languages, especially those with limited resources. The currently predominant approaches in the field, based on LSBert and masked language models, have noticeably better performances in English than other languages, even in the case of comparable evaluation datasets (see the results of TSAR-2022 shared task (Saggion et al., 2022)).

[15]The output of LeSS for the ALEXSIS and TSAR-2022 datasets is provided at: `https://github.com/danielibanezgarcia/less`.

Broader Impact

Lexical simplification can have a significant social impact by making texts understandable to people with various reading and cognitive impairments (see Section 1) and thus enabling them to actively participate in the society. We showed that carefully designed modular architectures can achieve state-of-the-art results and outperform popular architectures that are computationally much more expensive. Computationally-light architectures, such as the one we propose, are especially important for bringing lexical simplification closer to the real-world usage, as they can be easily used on mobile devices. Furthermore, the proposed modular architecture offers possibilities for building personalized lexical simplification systems by adjusting the ranking functions to the specific needs of each user.

Ethical Considerations

The final users of lexical simplification systems cannot fully understand original texts. That makes them vulnerable to the system's mistakes. Therefore, it is important to have thorough checks for system failures on different domains and types of texts, and have a manual post-editing function, should lexical simplification systems be used in real-world scenarios. Furthermore, the state-of-the-art lexical simplification systems rely on the use of word embeddings and transformers, which are known to propagate certain racial and gender biases. Before their application in real-world scenarios, it is thus important to thoroughly check for any type of ethical biases that may have been induced due to the underlying resources used in the system.

Acknowledgements

We would like to thank Daniel Ferrés for his help with the integration of the morphological generator. Horacio Saggion is thankful for the support from the individual project Context-aware Multilingual Text Simplification (ConMuTeS) PID2019-109066GB-I00/AEI/10.13039/501100011033 awarded by Ministerio de Ciencia, Innovación y Universidades (MCIU) and by Agencia Estatal de Investigación (AEI) of Spain. He also acknowledges support from the project MCIN/AEI/10.13039/501100011033 under the Maria de Maeztu Units of Excellence Programme (CEX2021-001195-M) and partial support from Departament de Recerca i Universitats de la Generalitat de Catalunya.

References

Rodrigo Alarcón, Lourdes Moreno, and Paloma Martínez. 2021. Exploration of Spanish Word Embeddings for Lexical Simplification. In *Proceedings of the First Workshop on Current Trends in Text Simplification CTTS 2021)*, volume 2944 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Rodrigo Alarcon, Lourdes Moreno, and Paloma Martínez. 2023. Easier corpus: A lexical simplification resource for people with cognitive impairments. *PLOS ONE*, 18(4):1–23.

Oliver Alonzo, Matthew Seita, Abraham Glasser, and Matt Huenerfauth. 2020. Automatic text simplification tools for deaf and hard of hearing adults: Benefits of lexical simplification and providing users with autonomy. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–13.

Sandra M. Aluísio, Lucia Specia, Thiago A. S. Pardo, Erick G. Maziero, and Renata P. M. Fortes. 2008. Towards Brazilian Portuguese automatic text simplification systems. In *ACM Symposium on Document Engineering*, pages 240–248.

Ricardo A. Baeza-Yates, Luz Rello, and Julia Dembowski. 2015. CASSA: A Context-Aware Synonym Simplification Algorithm. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1380–1385, Denver, Colorado, USA.

Stefan Bott, Luz Rello, Biljana Drndarević, and Horacio Saggion. 2012. Can Spanish Be Simpler? LexSiS: Lexical Simplification for Spanish. In *Proceedings of the 24th International Conference on Computational Linguistics*, COLING, pages 357–374.

John Carroll, Guido Minnen, Yvonne Canning, Siobhan Devlin, and John Tait. 1998. Practical Simplification of English Newspaper Text to Assist Aphasic Readers. In *Proceedings of the AAAI'98 Workshop on Integrating AI and Assistive Technology*, pages 7–10.

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*.

Emmanuele Chersoni and Yu-Yin Hsu. 2022. PolyU-CBS at TSAR-2022 shared task: A simple, rank-based method for complex word substitution in two steps. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 225–230, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Jan De Belder, Koen Deschacht, and Marie-Francine Moens. 2010. Lexical simplification. In *Proceedings of the 1st International Conference on Interdisciplinary Research on Technology, Education and Communication*, Kortrijk, Belgium.

Siobhan Devlin and John Tait. 1998. The Use of a Psycholinguistic Database in the Simplification of Text for Aphasic Readers. *Linguistic Databases*, pages 161–173.

Siobhan Devlin and Gary Unthank. 2006. Helping Aphasic People Process Online Information. In *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 225–226, New York, NY, USA. ACM.

Lijun Feng, Noémie Elhadad, and Matt Huenerfauth. 2009. Cognitively Motivated Features for Readability Assessment. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL, pages 229–237, Stroudsburg, PA, USA. Association for Computational Linguistics.

Daniel Ferrés, Ahmed AbuRa'ed, and Horacio Saggion. 2017. Spanish Morphological Generation with Wide-Coverage Lexicons and Decision Trees. *Procesamiento del Lenguaje Natural*, 58:109–116.

Daniel Ferrés, Horacio Saggion, and Xavier Gómez Guinovart. 2017. An adaptable lexical simplification architecture for major Ibero-Romance languages. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 40–47, Copenhagen, Denmark. Association for Computational Linguistics.

Daniel Ferrés and Horacio Saggion. 2022. ALEXSIS: A Dataset for Lexical Simplification in Spanish. In *Proceedings of the Language Resources and Evaluation Conference*, pages 3582–3594, Marseille, France. European Language Resources Association.

Goran Glavaš and Sanja Štajner. 2015. Simplifying Lexical Simplification: Do We Need Simplified Corpora? In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, ACL, pages 63–68.

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

Michael Heilman, Kevyn Collins-Thompson, Jamie Callan, and Maxine Eskenazi. 2007. Combining lexical and grammatical features to improve readability measures for first and second language texts. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 460–467, Rochester, New York. Association for Computational Linguistics.

David Hirsh and Paul Nation. 1992. What vocabulary size is needed to read unsimplified textsfor pleasure? *Reading in a Foreign Language*, 8(2):689–696.

Kentaro Inui, Atsushi Fujita, Tetsuro Takahashi, Ryu Iida, and Tomoya Iwakura. 2003. Text Simplification for Reading Assistance: A Project Note. In *Proceedings of the 2nd International Workshop on Paraphrasing: Paraphrase Acquisition and Applications*, pages 9–16.

Kai North, Alphaeus Dmonte, Tharindu Ranasinghe, and Marcos Zampieri. 2022. GMU-WLV at TSAR-2022 shared task: Evaluating lexical simplification models. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 264–270, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

OECD. 2013. OECD Skills Outlook 2013: First Results from the Survey of Adult Skills. Technical report, OECD Publishing.

Constantin Orăsan, Richard Evans, and Ruslan Mitkov. 2018. *Intelligent Text Processing to Help Readers with Autism*, pages 713–740. Springer International Publishing, Cham.

Lluís Padró and Evgeny Stanilovsky. 2012. FreeLing 3.0: Towards Wider Multilinguality. In *Proceedings of LREC 2012*. ELRA.

Gustavo H. Paetzold and Lucia Specia. 2017. A survey on lexical simplification. *Journal of Artificial Intelligence Research*, 60(1):549–593.

Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. 2020. Lsbert: A simple framework for lexical simplification.

Luz Rello, Ricardo A. Baeza-Yates, Stefan Bott, and Horacio Saggion. 2013a. Simplify or help?: text simplification strategies for people with dyslexia. In *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility*, W4A, Rio de Janeiro, Brazil.

Luz Rello, Ricardo A. Baeza-Yates, Laura Dempere-Marco, and Horacio Saggion. 2013b. Frequent Words Improve Readability and Short Words Improve Understandability for People with Dyslexia. In *Proceedings of the International Conference on Human-Computer Interaction (Part IV)*, INTERACT, pages 203–219, Cape Town, South Africa.

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 the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 271–283, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Horacio Saggion, Sanja Štajner, Stefan Bott, Simon Mille, Luz Rello, and Biljana Drndarević. 2015. Making It Simplext: Implementation and Evaluation of a Text Simplification System for Spanish. *ACM*

Transactions on Accessible Computing (TACCESS), 6(4):14.

Matthew Shardlow. 2014. Out in the Open: Finding and Categorising Errors in the Lexical Simplification Pipeline. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, LREC, Reykjavik, Iceland. European Language Resources Association (ELRA).

Matthew Shardlow, Richard Evans, Gustavo Henrique Paetzold, and Marcos Zampieri. 2021. SemEval-2021 task 1: Lexical complexity prediction. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 1–16, Online. Association for Computational Linguistics.

Lucia Specia, Sujay Kumar Jauhar, and Rada Mihalcea. 2012. SemEval-2012 task 1: English Lexical Simplification. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*, SemEval, pages 347–355, Stroudsburg, PA, USA. Association for Computational Linguistics.

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.

Laura Vásquez-Rodríguez, Nhung Nguyen, Matthew Shardlow, and Sophia Ananiadou. 2022. UoM&MMU at TSAR-2022 shared task: Prompt learning for lexical simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 218–224, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Sanja Štajner. 2021. Automatic text simplification for social good: Progress and challenges. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2637–2652, Online. Association for Computational Linguistics.

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*, 5.

Willian M. Watanabe, Arnaldo Candido Jr., Marcelo A. Amâncio, Matheus De Oliveira, Thiago A. S. Pardo, Renata P. M. Fortes, and Sandra M. Aluísio. 2010. Adapting web content for low-literacy readers by using lexical elaboration and named entities labeling. *New Review of Hypermedia and Multimedia*, 16(3):303–327.

Peniel 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 the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 213–217, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Rodrigo Wilkens, David Alfter, Rémi Cardon, Isabelle Gribomont, Adrien Bibal, Watrin Patrick, Marie-Catherine De marneffe, and Thomas François. 2022. CENTAL at TSAR-2022 shared task: How does context impact BERT-generated substitutions for lexical simplification? In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 231–238, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

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. Association for Computational Linguistics.

Seid Muhie Yimam, Sanja Štajner, Martin Riedl, and Chris Biemann. 2017. Cwig3g2 - complex word identification task across three text genres and two user groups. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 401–407. Asian Federation of Natural Language Processing.