Journal of Intelligent Information Systems
https://doi.org/10.1007/s10844-024-00882-9

RESEARCH

Deep learning approaches to lexical simplification: A survey

Kai North[1] · **Tharindu Ranasinghe**[2] · **Matthew Shardlow**[3] · **Marcos Zampieri**[1]

Received: 1 May 2024 / Revised: 17 August 2024 / Accepted: 19 August 2024

Abstract
Lexical Simplification (LS) is the task of substituting complex words within a sentence for simpler alternatives while maintaining the sentence's original meaning. LS is the lexical component of Text Simplification (TS) systems with the aim of improving accessibility to various target populations such as individuals with low literacy or reading disabilities. Prior surveys have been published several years before the introduction of transformers, transformer-based large language models (LLMs), and prompt learning that have drastically changed the field of NLP. The high performance of these models has sparked renewed interest in LS. To reflect these recent advances, we present a comprehensive survey of papers published since 2017 on LS and its sub-tasks focusing on deep learning. Finally, we describe available benchmark datasets for the future development of LS systems.

Keywords Deep learning · Accessibility · Lexical simplification

1 Introduction

LS improves text readability by replacing complex words with simpler alternatives. Complex words are words which a target population find difficult to read or understand. Various user groups benefit from LS. Previous LS systems have been designed for children (Kajiwara et al., 2013), second language learners (Lee & Yeung, 2018b), individuals with reading disabilities (Rello et al., 2013; Devlin & Tait, 1998; Carroll et al., 1998) or low-literacy (Watanabe et al., 2009; Gasperin et al., 2009), and sign language speakers (Alonzo et al., 2022a, b). LS provides a degree of personalization that is unattainable through approaches that focus on sentence rather than word-level simplification (Yeung & Lee, 2018; North & Zampieri, 2023).

The introduction of deep learning, and more recently, LLMs and prompt engineering, has significantly changed the way we approach many NLP tasks, including LS. Previous LS systems have relied upon statistical, n-gram, lexical, rule-based, and word embedding models to identify complex words and then replace them for simpler alternatives (Paetzold &

✉ Kai North
knorth8@gmu.edu

[1] George Mason University, Fairfax, VA, USA

[2] Lancaster University, Lancaster, UK

[3] Manchester Metropolitan University, Manchester, UK

Published online: 02 September 2024

Specia 2017b). These approaches would identify a complex word, for example, "*rogue*" as being in need of simplification, and would suggest "*thief*" as a suitable alternative (Fig. 1), hereby referred to as a candidate substitution. Transformer-based models, such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), and the latest generation of LLMs with billions of parameters such as as GPT-3 and GPT-4 (Brown et al., 2020), Llama 2 and LLama 3 (Touvron et al., 2023), Mistral (Jiang et al., 2023), and others, automatically generate, select, and rank candidate substitutions. Furthermore, recent shared task results (Shardlow et al., 2024) have confirmed that that LLMs deliver performances superior to traditional approaches resulting in an important paradigm shift within LS which motivates the present survey. This corroborates recent findings of multiple NLP studies that show that the most recent generation of LLMs deliver state-of-the-art performance in various tasks (Minaee et al., 2024).

With the introduction of the aforementioned new deep learning models and LLMs, the field of NLP has seen the arrival of new interested parties: companies, academics, and individuals that may be unfamiliar with prior LS research (Shardlow et al., 2021; Saggion et al., 2022). For this reason, we believe that this is the perfect moment to provide a survey of recent deep learning approaches to LS with the goal of bridging the gap between established researchers and those new to the field. To the best of our knowledge, this is the first survey on deep learning approaches for LS. The paper by Paetzold & Specia (2017b) is the most recent survey on LS, but it has been published before studies that demonstrate the headway made by state-of-the-art deep learning models. A broader survey on TS (Al-Thanyyan & Azmi, 2021), published a few years later, also does not cover recent advances in the field, nor does it focus specifically on LS. Our paper, therefore, fills an important gap in the LS literature by providing the community with the first survey on deep learning approaches to LS and its sub-tasks of substitute generation (**SG**), substitute selection (**SS**), and substitute ranking (**SR**).

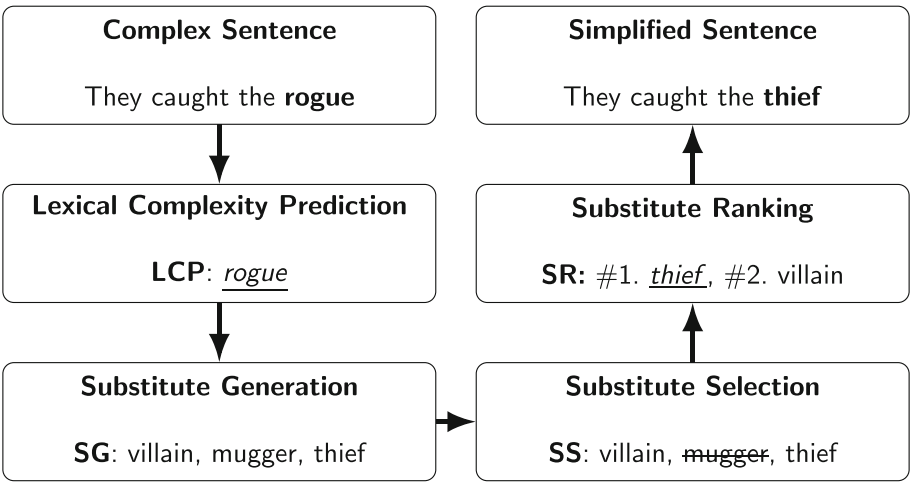

Fig. 1 LS Pipeline. SG, SS, and SR are the main sub-tasks of LS discussed throughout this survey. Figure adapted from (Paetzold & Specia, 2015)

2 Pipeline

We organize this survey around the main components of the LS pipeline with SS and SR being described simultaneously due to the likeliness in their deep learning approaches (Section 3). We discuss recent works collected from prominent LS workshops, shared-tasks, conference proceedings, and journals published in three major repositories, namely ACL Anthology, ACM Digital Library, and IEEE Xplore. In addition, we provide an overview of recent datasets (Section 4), and detail open challenges and unanswered research questions in LS (Section 5.1). Normally, a LS pipeline starts with lexical complexity prediction (**LCP**), also known as complex word identification (CWI). However, since LCP is often considered as a standalone task, we recommend (North et al., 2022b), for a detailed survey on LCP methods.

Substitute Generation (SG) The goal of SG is to produce a number: k, of candidate substitutions that are viable replacements for a complex word. Typically, a LS system will output candidate substitutions in the range of $k = [1, 3, 5, \text{or } 10]$ with *top-k* referring to the most suitable candidates. These candidate substitutions need to be easier to understand or read than the complex word. Candidate substitutions likewise need to preserve the complex word's meaning, especially in its provided context (Table 1). For example, given the sentence: "*They caught the rogue that stole the gold.*", and the target word: "*rogue*", substitute generation would produce k candidate substitutions, such as "*villain*", "*mugger*", "*thief*", and so on.

Multiple approaches have been used to generate viable candidate substitutions for a given complex word, ranging from the use pre-existing lexicons, masked language modeling, to the use of recent LLMs and prompt learning. These approaches are described in more detail within Section 3.1. However, SG is not without its limitations. SG systems have been found to produce candidate substitutions that are unsuitable for a target word's context, and SG systems reliant on multilingual models trained on datasets consisting of multiple languages often produce candidate substitutions not in the desired target language of the complex word. Moreover, in some instances, candidate substitutions are produced that are more difficult to understand than the original complex word. In these instances, additional filtering is conducted via substitute selection or ranking as described in the following sections. For example, an LCP regressor may be trained to predict the lexical complexity of generated candidates on a scale between 0 (easy), 0.5 (neutral), and 1 (difficult); we refer the reader again to North et al. (2022b) for a detailed survey on LCP methods. The approaches described in Section 3.1 attempt to overcome these issues in numerous ways.

Substitute Selection (SS) The aim of SS is to remove generated *top-k* candidate substitutions which are not suitable. At this stage, candidate substitutions that are not synonymous or that are more complex than the original complex word are removed (Table 2). For instance, SS would remove those generated candidates: "*mugger*", "*burglar*", and "*poacher*", since they

Table 1 Example of candidate substitution generated via SG

Target Word	Rank	Candidate Replacement
rogue	#1	villain
	#2	mugger
	#3	thief
	#4	burglar
	#5	poacher
	#n	...

Table 2 Example of candidate substitutes removed during SS

Target Word	Rank	Candidate Replacement
rogue	#1	villain
	#2	
	#3	thief
	#4	
	#5	
	#n	...

are either more complex, semantically dissimilar, or do not fit into the provided context: *They caught the rogue that stole the gold.*

Common approaches to SS include comparing cosine similarities between word embeddings, training independent models for candidate selection, or through the use of prompt learning. These approaches are described within Section 3.2. The main challenge of SS is to not remove correct simplifications. A valid simplification, at times, may have a low similarity with the original complex word, i.e. it is less synonymous with the original complex word in comparison to other alternatives. However, the same simplification may better fit within the original context and therefore be a superior simplification. The approaches outlined in Section 3.2 have used various methods to minimise the likelihood of correct simplifications being removed from the pool of candidate substitutions.

Substitute Ranking (SR)

The purpose of SR is to rank the left over *top-k* candidate substitutions from the most to the least suitable simplification. The original complex word is then substituted with the most viable candidate substitution. The example shown in Table 3 ranks "*thief*" as being a more appropriate simplification than "*criminal*", "*vilian*", or "*bandit*" for the target word "*rogue*". This may, in part, be due to "*thief*" having a higher frequency within a reference corpus or being more frequent within a training set. Alternatively, "*thief*" may have a lower age of acquisition, higher familiarity score, or even concreteness (abstractness) rating.

Approaches used for SS are also frequently employed for SR. Candidate substitutions have been ranked per cosine similarity between embeddings, BERT score, or through the use of prompt learning. These approaches are discussed in Section 3.2 in conjunction with SS approaches. However, the ranking of candidate substitutions is not an easy task. Perceptions on word complexity differ from individual-to-individual and therefore the target demographic

Table 3 Example of candidate substitutions ranked via SR

Target Word	Rank	Candidate Replacement
rogue	**#1**	**thief**
	#2	criminal
	#3	villain
	#4	bandit
	#5	stealer
	#n	...

needs to be taken into consideration when deciding which is the best candidate substitution to replace the original complex word. Unique approaches have been created in this endeavour (outlined in Section 3.2).

3 Deep Learning Approaches

We start our survey of the LS pipeline at the SG phase (Section 3.1). We then move on to SS and SR (Section 3.2). Within these sections, we provide an overview of deep learning approaches for LS and make reference to common evaluation metrics used to assess all LS sub-tasks: precision, recall, F1-score, accuracy (ACC) alongside potential, and mean average precision (MAP) at top-k as defined below.

Potential@k The ratio of predicted candidate substitutions for which at least one of the top-k candidate substitutions was found within the gold labels as shown in equation (1).

$$Potential = \frac{m}{n} \tag{1}$$

where m is the total of predicted top-k candidate substitutions that are found within the gold labels, and n is the sum of gold labels taken into consideration.

MAP@k The ratio of returned top-k candidate substitutions that are equal to the gold labels and have the same positional rank. It is calculated using the following equation (2):

$$MAP = \frac{1}{n} \sum_{i=1}^{n} AP_i \tag{2}$$

where AP is the average precision of each class, i is the class, and n is the number of classes taken into consideration.

3.1 Substitute Generation

Word embedding models were frequently used for SG. Word embedding models, such as Word2Vec (Mikolov et al., 2013), were combined with more traditional approaches, such as querying a lexicon, or generating candidate substitutions based on certain rules (Paetzold & Specia, 2017b). Word embedding models conducted SG by converting potential candidate substitutions into vectors, thus generating word embeddings. They determined the highest cosine similarity or lowest cosine distance between these vectors and the vector representing the target complex word. These vectors were then reverted to their word forms, constituting the top-k candidate substitutions.

Word Embeddings + Transformers Word embedding models continued to be used for SG after 2017. However, they were now used alongside word embeddings produced by transformers or by a model's prediction scores. Alarcón et al. (2021a) utilized BERT and various word embeddings models for generating Spanish candidate substitutions, such as Sense2Vec, Word2Vec, (Trask et al., 2015), and FastText (Bojanowski et al., 2017). It was found that a more traditional approach that generated candidate substitutions by querying a pre-existing lexicon outperformed these word embedding models. Their traditional approach achieved a potential of 0.898, a recall of 0.597, and a precision of 0.043 on the EASIER dataset

(Alarcón et al., 2021b). In contrast, the highest-performing embedding model, Sense2Vec, scored lower with a potential, recall, and precision of 0.506, 0.282, and 0.056, respectively. Interestingly, this went against the assumption that word embedding models would have achieved a higher performance given their state-of-the-art reputation (Paetzold & Specia, 2017a). During error analysis, it was found that word embedding models often proposed antonyms of the complex word as potential replacements, thereby hindering LS performance (Alarcón et al., 2021a).

Seneviratne et al. (2022) used a word embedding model and a pre-trained transformer: XLNet (Yang et al., 2019), to provide an embedding similarity score and a prediction score for SG. They took inspiration from a similar approach conducted by Arefyev et al. (2020). Arefyev et al. (2020) utilized context2vec (Melamud et al., 2016) and ELMo (Peters et al., 2018) to encode the context of the target complex word to gain a probability distribution of each word fitting into that particular context. They utilized this probability distribution to gauge the likelihood or appropriateness of a potential candidate substitution being an effective replacement for the target complex word. This score was combined with a prediction score from either BERT, RoBERTa, or XLNet to generate a final list of top-k candidate substitutions. The combined approach of employing a word embedding model alongside a prediction score was found to under-perform compared to utilizing a single pre-trained transformer (Seneviratne et al., 2022; Arefyev et al., 2020). For instance, Seneviratne et al. (2022) reported inferior performance compared to North et al. (2022a) on the TSAR-2022 dataset.

Masked Language Modeling

The arrival of pre-trained transformers, also saw the introduction of Masked Language Modeling (MLM) for SG. MLM is where words in a sentence are masked or hidden, and the model is tasked with predicting what these masked tokens should be based on the context provided by the rest of the sentence. MLM is therefore well suited for SG. Przybyła & Shardlow (2020) used BERT-based models trained on a MLM objective for multi-word LS, whereas Qiang et al. (2020) were the first to utilize MLM for Spanish SG. MLM has emerged as a prevalent approach in SG, with 7 out of the 11 systems submitted to TSAR-2022 incorporating an MLM objective (Saggion et al., 2022).

Qiang et al. (2020) created LSBert, being a pre-trained BERT-based model for LS. Extracts, in the form of sentences, were taken from the LS datasets: LexMTurk (Horn et al., 2014), BenchLS (Paetzold & Specia, 2016b), and NNSeval (Paetzold & Specia, 2016c). Two versions of each sentence were then concatenated, being separated by the [SEP] special token. They were inputted into their model. The initial sentence mirrored one extracted from the datasets, while the subsequent sentence had its complex word substituted with the [MASK] special token. LSBert then predicts the word replaced by the [MASK] special token by analyzing its context, including both the preceding and succeeding text of the target word, alongside the original sentence. In this way, LSBert outputted candidate substitutions with the highest probability (highest prediction score) of fitting into the surrounding context and that are also similar to the target complex word in the original sentence. For the top-k=1 candidate substitution, LSBert achieved F1-scores for SG of 0.259, 0.272, and 0.218 on the three datasets LexMTurk (Horn et al., 2014), BenchLS (Paetzold & Specia, 2016b), and NNSeval (Paetzold & Specia, 2016c), respectively. Performances surpassed that of all prior

approaches (Paetzold & Specia, 2017b). Their MLM approach's ability to take into consideration context before and after the target word and their use of an overall larger BERT-based model is responsible for this increase in performance. The previous highest F1-score was achieved by a word-embedding model that lacked the same contextual understanding having produced F1-scores of 0.195, 0.236, and 0.218 for the same datasets, respectively (Paetzold & Specia, 2017a).

Prior to the release of the TSAR-2022 shared-task (Saggion et al., 2022), Ferres & Saggion (2022) created a new dataset: ALEXSIS (TSAR-2022 ES), that would later become (together with an additional English and Portuguese dataset) the TSAR-2022 dataset (Saggion et al., 2022). Using their new dataset, they experimented with a number of monolingual transformers and several multilingual transformers. Ferres & Saggion (2022) adopted the MLM approach used by LSBert. They used the Spanish pre-trained models: BETO (Cañete et al., 2020), BERTIN (De la Rosa & Fernández, 2022), RoBERTa-base-BNE, and RoBERTA-large-BNE (Fandiño et al., 2022) for SG. They found that their largest pre-trained Spanish model: RoBERTA-large-BNE, attained the greatest SG performance after having also omitted candidate substitutions that were the same as the complex word, regardless of capitalization or accentuation and being less than two characters long.

North et al. (2022a) was motivated by the success of the monolingual models shown by Ferres & Saggion (2022). They likewise tested a range of pre-trained transformers for SG with a MLM objective, including multilingual models: mBERT, and XLM-R (Conneau et al., 2020), and several monolingual models, including Electra for English (Clark et al., 2020), RoBERTA-large-BNE for Spanish, and BERTimbau (Souza et al., 2020) for Portuguese. Their monolingual models scored an ACC@1 score of 0.517, 0.353, and 0.481 on the English, Spanish, and Portuguese TSAR-2022 datasets, respectively. Whistely et al. (2022) likewise used similar monolingual models for SG. They experimented with BERT for English, BETO for Spanish, and BERTimbau for Portuguese. Surprisingly, their models' performances were lower compared to that of North et al. (2022a), despite their Portuguese LS system consisting of the same pre-trained model. Whistely et al. (2022) produced ACC@1 scores of 0.378, 0.250, and 0.3074 for English, Spanish, and Portuguese, respectively. This is likely because the additional selection and ranking steps implemented by Whistely et al. (2022) and the lack thereof shown within the LS system provided by North et al. (2022a) (Section 3.2).

Wilkens et al. (2022) likewise experimented with a range of monolingual transformers for SG. They employed an ensemble of BERT-like models with three distinct masking strategies: 1) copy, 2) query expansion, and 3) paraphrase. In the copy strategy, similiar to LSBert's approach (Qiang et al., 2020), two sentences were fed into a pre-trained model, concatenated with the [SEP] special token. The first sentence remained unchanged, while the second had its complex word replaced with the [MASK] token. For the query expansion strategy, FastText was utilized to generate five related words with the highest cosine similarity to the target complex word. In iteration 2a) of this strategy, the first sentence remained unaltered, the second substituted the complex word with one of the recommended similar words from FastText, and the third sentence was the masked version. Iteration 2b) replicated 2a), but the second sentence now comprised all five suggested words. Lastly, the paraphrase strategy generated 10 new contexts for each complex word, consisting of paraphrases of the original sentence, each with a maximum of 512 tokens. These ensembles encompassed BERT and RoBERTa for English, several BETO-based models for Spanish, and several BERTimbau-based models for Portuguese. The paraphrase strategy showed the worst performance with a joint MAP/Potential@1 score of 0.217, whereas the query expansion strategy obtained a MAP/Potential@1 score of 0.528, 0.477, and 0.476 for English, Spanish, and Portuguese,

respectively. This outperformed the paraphrase strategy and the original copy strategy used by LSBert, regardless of model.

Prompt Learning Prompt learning is currently the best performing approach for SG as a result of the utilization of larger and more recent LLMs (Table 4). Prompt learning involves inputting into a LLM a string that is presented in such a way as to provide a description of the task as well as to return a desired output. Prompt learning, otherwise referred to as prompt engineering, also entails the optimization of said prompts to achive the best SG performance. This is achieved by either trial and error, or the employment of prompt learning strategies, such as chain-of-thought prompting. LLMs may also be fine-tuned for SG by being provided a dataset that includes example prompts, instructions and their corresponding outputs. However, little research has been conducted on LLM fine-tuning for SG within the LS research community having preferred zero-shot experimentation. Zero-shot refers to a LLM not being exposed to any training material or example instances with corresponding 1-shot, 2-shot, and so on, referring to the number of example instances shown to the LLM.

PromptLS (Vásquez-Rodríguez et al., 2022) is one of the only examples of prompt learning and LLM fine-tuning applied to SG. PromptLS contains a variety of pre-trained models fine-tuned on several LS datasets. These fined-tuned models were fed four types of prompts: a). "a *easier word* for rogue is", b). "a *simple word* for rogue is", c). "a *easier synonym* for rogue is", and lastly, d). "a *simple synonym* for rogue is". These prompts were fed into a RoBERTa model on all of the English data extracted from the NNSeval (Paetzold & Specia, 2016c), LexMTurk (Horn et al., 2014), CEFR-LS (Uchida et al., 2018) and BenchLS (Paetzold & Specia, 2016b) datasets. They were also translated and inputted into BERTIN fine-tuned on the Spanish data obtained from EASIER, along with BR-BERTo fine-tuned on all of the Portuguese data taken from SIMPLEX-PB (Hartmann & Aluísio, 2020). Vásquez-Rodríguez et al. (2022) likewise experimented with these prompts on a zero-shot condition. It was discovered that the fine-tuned models outperformed the zero-shot models on all conditions by an average increase in performance between 0.3 to 0.4 across all metrics: ACC@1, ACC@3, MAP@3, and Precision@3. The prompt combinations that produced the best candidate substitutions were "easier word" for English, "palabra simple" and "palabra fácil" for Spanish, and "palavra simples" and "sinônimo simples" for Portuguese.

Prompt learning has also been applied to more recent LLMs for SG, namely GPT-3 models. Aumiller & Gertz (2022) experimented with a variety of prompts, which they inputted into GPT-3. These prompts included: 1). zero-shot with context, 2). single-shot with context, 3). two-shot with context, 4). zero-shot without context, and 5). single-shot without context. In this instance, the size of each shot: *n*, refers to how many times a prompt is inputted into GPT-3. For instance, those shots with context would input a given sentence and then ask the question, "Given the above context, list ten alternative words for <complex word> that are easier to understand.", *n* number of times. Those without context, however, would input *n* times the following: "Give me ten simplified synonyms for the following word: <complex word>". Aumiller & Gertz (2022) also combined all types of prompts in an ensemble, generating candidate substitutions from each prompt type and then deciding upon final candidate substations through plurality voting and additional selection and ranking steps (Section 3.2). Their ensemble approach outperformed all other prompt types and SG models submitted to TSAR-2022 (Saggion et al., 2022). Their performance is a result GPT-3 being substantially larger than all other models submitted to the shared-task.

Table 4 Top approaches for substitute generation (SG), selection and ranking (SS & SR) on TSAR-2022 datasets. **Pot.** stands for potential

Lang.	SG	SS & SR	ACC	ACC@1	ACC@3	MAP@3	Pot@3	Paper
EN	GPT-3+Prompts	GPT-3	**0.8096**	**0.4289**	**0.6863**	**0.5834**	**0.9624**	(Aumiller & Gertz, 2022)
	MLM	Embeddings+Freq	0.6568	0.3190	0.5388	0.4730	0.8766	(Li et al., 2022)
	BERT+Prompts	MLM Prediction Score	0.6353	0.2895	0.5308	0.4244	0.8739	(Vásquez-Rodríguez et al., 2022)
ES	GPT-3+Prompts	GPT-3	**0.6521**	**0.3505**	**0.5788**	**0.4281**	**0.8206**	Aumiller & Gertz (2022)
	MLM	Embeddings+POS	0.3695	0.2038	0.3288	0.2145	0.5842	Whistely et al. (2022)
	BERT+Prompts	MLM Prediction Score	0.3668	0.160	0.2690	0.2128	0.5326	Vásquez-Rodríguez et al. (2022)
PT	GPT-3+Prompts	GPT-3	**0.7700**	**0.4358**	**0.6299**	**0.5014**	**0.9171**	(Aumiller & Gertz, 2022)
	MLM	MLM Prediction Score	0.4812	0.2540	0.3957	0.2816	0.6871	(North et al., 2022a)
	MLM	Freq+BinaryClassifier	0.3689	0.1737	0.2673	0.1983	0.5240	(Wilkens et al., 2022)

3.2 Substitute Selection and Ranking

Traditional SS approaches are still applied after SG. Methods such as POS-tag and antonym filtering, as well as semantic thresholds have been used to omit inappropriate candidate substitutions generated from the above deep learning approaches (Saggion et al., 2022). However, most modern deep learning approaches have minimal SS, with SS often being conducted at the same time as generation or ranking. For example, the metric used to generate the top-k candidate substitutions, such as similarity between word embeddings or a model's prediction score, tends not to suggest candidate substitutions that are deemed as being inappropriate by other SS methods. Furthermore, SR techniques that order candidate substitutions per their appropriateness will in turn move unsuitable simplifications further down the list of top-k candidate substitutions to the point that they are no longer considered. For this reason, we have combined SS and SR into one section and described new deep learning approaches below.

Word Embeddings Word embedding models continue to play a role in SS. For instance, Song et al. (2020) developed a novel LS system that filtered candidate substitutions based on a semantic similarity threshold. They selected only those candidate substitutions sharing the same POS tag as the target complex word, assessed contextual relevance (a measure of the reasonableness and fluency of a sentence after replacing the complex word), and ranked candidate substitutions by applying cosine similarity between word embeddings. They produced word embeddings by Word2Vec and rated their model's performance on the LS-2007 dataset (McCarthy & Navigli, 2007). It was discovered that the use of Word2Vec enhanced their model's performance, having achieved an ACC@1 of 0.269 compared to a previous score of 0.218.

Neural Regression

Maddela & Xu (2018) introduced the Neural Readability Ranker (NNR) to arrange candidate substitutions based on their complexity. NNR employs regression, trained on the Word Complexity Lexicon (WCL), alongside various features and character n-grams transformed into Gaussian vectors. It assigns a value between 0 and 1 representing the complexity of any given word. Through pairwise aggregation, the model predicts values indicating the relative complexity between pairs of candidate substitutions. A positive value suggests that the first candidate substitution is more complex than the second, while a negative value indicates the opposite. This process is repeated for all combinations of candidate substitutions for a complex word. Subsequently, each candidate substitution is ranked based on its comparative complexity with others. Applying their NNR model to the LS-2012 dataset, Maddela & Xu (2018) surpassed previous word embedding techniques for SR, achieving a Prec@1 of 0.673 compared to 0.656. Their approach therefore benefited from regression fine-tuning on a new human annotated dataset.

Word Embeddings + Transformers A popular approach to SS and SR entails the use of word embeddings and transformers. Seneviratne et al. (2022) filtered and ranked candidate substitutions per the same combined score that they used for SG. Their filter consisted of their MLM model's prediction score of the generated candidate together with the inner product of the target word's embedding and the embedding of the potential candidate substitution. The returned candidate substitutions were then subject to one of three additional ranking metrics. The first ranking metric arranged candidate substitutions based on the cosine similarity between the original sentence and a modified version where the candidate substitution replaced the complex word. The second and third ranking metrics utilized dictionary defini-

tions of the target complex word and its candidate substitutions. They computed the cosine similarity between each embedding of the definitions and the embedding of the sentence containing the target complex word. Those with the highest cosine similarities between either a) the definition of the target complex word and the definition of the candidate substitution, or b) the definition of the target complex word and the word embedding of the original sentence with the candidate substitution replacing its complex word, determined the rank of each candidate substitution. Their analysis revealed similar performances across all three metrics on the TSAR-2022 dataset, with a) achieving an ACC@1 score of 0.375, b) achieving 0.380, and c) achieving 0.386.

Li et al. (2022) created what they refer to as equivalence score for selection and ranking. Equivalence score determines the semantic similarity between candidate substitution and complex word to an extent that was more expressive than cosine similarity between word embeddings. To calculate equivalence score, they used a RoBERTa-based model trained for natural language inference (NLI) which outputs the likelihood of one sentence entailing another. The product of the generated likelihood of the original sentence with the candidate substitution preceding the original sentence and vice-versa equated to the equivalence score. (Li et al., 2022), employing the SG method akin to LSBert but with a transition to RoBERTa, attributed their system's improved performance primarily to its distinctive SR. They achieved an ACC@1 of 0.659, surpassing LSBert's ACC@1 of 0.598 on the TSAR-2022 dataset.

Aleksandrova & Brochu Dufour (2022) employed three metrics to rank candidate substitutions: a) grammaticality, b) meaning preservation, and c) simplicity. Grammaticality was assessed by checking whether the candidate substitution had the same POS tag concerning person, number, mood, tense, etc. If the candidate substitution matched in all POS-tag categories, it was assigned a value of 1; otherwise, it received a value of 0. Meaning preservation was determined by utilizing BERTScore to compute cosine similarities between the embeddings of the original sentence and those of an altered sentence where the target complex word was replaced with the candidate substitution. Lastly, simplicity was gauged using a CEFR vocabulary classifier trained on data from the English Vocabulary Profile (EVP). This classifier was trained by first masking the data and inputting it into a pre-trained BERT-based model. The resulting encodings were then used to train an SVM model, yielding the CEFR classifier. Despite their efforts, their model failed to outperform the baseline LSBert model at TSAR-2022.

LS systems have also solely relied on MLM prediction scores for SS and SR. North et al. (2022a) and Vásquez-Rodríguez et al. (2022) used this approach. They have no extra SR steps and sort their candidate substitutions using their generated MLM prediction scores. That being said, the do apply some basic filtering with both studies omitting duplicates and candidate substitutions that were the same as the complex word. Interestingly, minimal SR outperforms other more technical approaches (Table 4). North et al. (2022a) attained state-of-the-art performance on the TSAR-2022 Portuguese dataset, whereas Vásquez-Rodríguez et al. (2022) consistently produced high performances across the TSAR-2022 datasets.

Prompt Learning Only North et al. (2023) have experimented with prompt learning for SS and SR. They created a unique selection and ranking pipeline (shown in Fig. 2) that removed candidate substitutions with a low cosine similarity between their own BERT embedding and that belonging to the complex word. Remaining candidates were then filtered by GPT 3.5. after being fed a prompt with one of the following adjectives, a). *simplest*, b). *best* or c). *most similar*: "What word is the [*adjective*] replacement for complex word in this list?". GPT 3.5 was then presented with a final prompt which selected the best simplification by assessing each candidates suitability in the original context: "Given the above context, what is *the best*

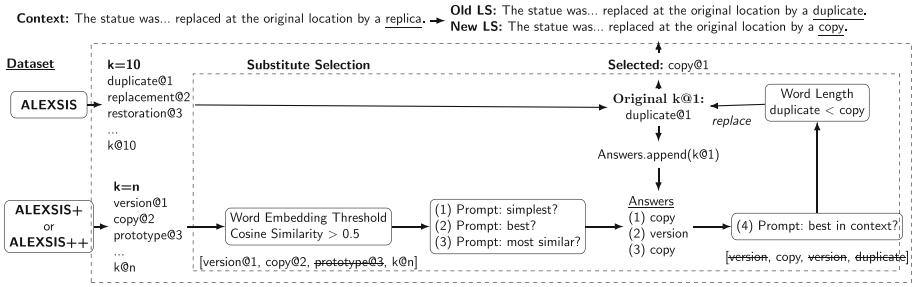

Fig. 2 The selection and ranking pipeline introduced by North et al. (2023)

replacement for complex word in this list?". This unique filter increased overall performance from an ACC@1 of 0.484 to 0.495 on the Portuguese TSAR-2022 dataset demonstrating the advantages of incorporating prompt learning within a SS and SR pipeline.

4 Resources

LS datasets (post-2017) exist for every LS sub-tasks or for a specific purpose (Appendix, Table 5). In addition, shared-tasks ($^{SharedT.}$) often provide their own LS datasets. Resources for LS are available for several languages as shown in the following sections.

4.1 English

Personalized-LS Lee & Yeung (2018b) introduced a dataset containing 12,000 English words ranked on a five-point Likert scale for personalized LS. 15 native Japanese speakers were asked to rate the complexity of each word. Complexity ratings were applied to BenchLS personalizing the dataset for Japanese speakers.

WCL Maddela & Xu (2018) created the Word Complexity Lexicon (WCL). The WCL has 15,000 English words annotated by 11 non-native English speakers using a six-point Likert scale.

LCP-2021$^{SharedT.}$ The dataset provided at the LCP-2021 shared-task (CompLex) (Shardlow et al., 2020), was crowd sourced in the UK, Australia and the US. 10,800 complex words in context were extracted from three corpora: the Bible, biomedical articles, and Europarliamentary proceedings. Annotation of complexity was done via a 5-point Likert scale.

SimpleText-2021$^{SharedT.}$ The SimpleText-2021 shared-task (Ermakova et al., 2021) designed three pilot tasks: 1). to identify passages to be simplified, 2). to select complex concepts within these passages, and 3). to simplify the complex concepts to produce an easier to understand passage. They gave their participants with multiple sources of data: the DBLP+Citation, Citation Network Dataset, ACM Citation network, and The Guardian newspaper with manually annotated keywords.

Table 5 Datasets that can be used for LS arranged in chronological order

	Dataset	LS Pipeline	Languages	#CWs	Avg.Subs	Domain	Annotators	Paper
Pre-2017	**LS–2007**$^{SharedT.}$	SG, SS	EN	201	1	Mix	5 UK-based.	(McCarthy & Navigli, 2007)
	PorSimples	SG, SS	PT	3066	1	News	1 Linguist.	(Aluísio & Gasperin, 2010)
	LS–2012$^{SharedT.}$	SG, SS, SR	EN	201	5	Mix	L1 English Speakers.	(Specia et al., 2012)
	CW Corpus	SS	EN	731	0	Wikipedia	Wikipedia Edits.	(Shardlow, 2013)
	LexMTurk	SG, SS, SR	EN	500	50	Wikipedia	50 US-based.	(Horn et al., 2014)
	JLS	SG, SS, SR	JP	243	5	Mix	5 L1 JP Speakers.	(Kajiwara & Yamamoto, 2015)
	JLS Balanced	SG, SS, SR	JP	2,010	5	Mix	L1 JP Speakers	(Kodaira et al., 2016)
	CWI–2016$^{SharedT.}$	SS	EN	90,458	0	News	400 L2 EN Speakers.	(Paetzold & Specia, 2016a)
	BenchLS	SG, SS, SR	EN	929	7	Mix	US-Based.	(Paetzold & Specia, 2016b)
	NNSeval	SG, SS, SR	EN	239	7	Mix	400 L2 EN Speakers.	(Paetzold & Specia, 2016c)
Post-2017	**CERF-LS**	SG, SS, SR	EN	406	12	Academic	1 L1 EN Speaker.	(Uchida et al., 2018)
	Personalized-ZH	SG, SS, SR	ZH	600	7	Mix	8 L1 ZH Speakers	(Lee & Yeung, 2018a)
	WCL	SS, SR	EN	15,000	0	Mix	11 L2 EN Speakers.	(Maddela & Xu, 2018)
	ReSyf	SG, SS	FR	57,589	3	Mix	L1 FR Speakers.	(Billami et al., 2018)
	Personalized-LS	SG, SS, SR	EN	929	7	Mix	15 L2 EN Speakers.	(Lee & Yeung, 2018b)
	CWI–2018$^{SharedT.}$	SS, SR	EN, FR, GR, ES	62,550	0	News	L1&L2 EN Speakers.	(Yimam et al., 2018)
	PorSimplesSent	SG, SS	PT	6109	1	News	3 Linguists.	(Leal et al., 2018)
	SIMPLEX-PB	SG, SS, SR	PT	730	5	Academic	pt-BR Speakers.	(Hartmann & Aluísio, 2020)
	JWCL-JSSL	SG	JP	18,000	0	Mix	5 L1 JP Speakers.	(Nishihara & Kajiwara, 2020)
	ALexS-2020$^{SharedT.}$	SG	ES	723	0	Academic	430 ES Speakers.	(Zambrano & Ráez, 2020)
	LCP-2021$^{SharedT.}$	SS, SR	EN	10,800	0	Mix	7 US/UK/AUS-based.	(Shardlow et al., 2020)
	SimpleText-2021	SG, SS, SR	EN	1000	10	Academic	Participating Teams.	(Ermakova et al., 2021)
	ES-CWI	SG	ES	3,887	0	Academic	40 L1 ES speakers.	(Merejildo, 2021)
	EASIER	SG, SS	ES	5,310	3	News	L1 ES speakers.	(Alarcón et al., 2021b)

Table 5 continued

Dataset	LS Pipeline	Languages	#CWs	Avg.Subs	Domain	Annotators	Paper
FrenLys	SG, SS, SR	FR	57,589	3	Mix	20 L1 FR Speakers.	(Rolin et al., 2021)
HanLS	SG, SS, SR	ZH	534	8	Mix	5 L1 ZH Speakers.	(Qiang et al., 2021)
TSAR-2022	SG, SS, SR	EN, ES, PT	1153	20	News	21 UK/ES/BR-based.	(Saggion et al., 2022)
WCL-DHH	SS, SR	EN	15,000	0	Mix	11 DHH Annotators.	(Alonzo et al., 2022a)
RUS-LCP	SS, SR	RUS	931	0	Bible	10 RUS-based.	(Abramov & Ivanov, 2022)

Marked datasets ($^{SharedT.}$) were used in benchmark competitions (shared-tasks)
L1 and L2 refers to first and second language speakers
A split between pre-2017 and post-207 is shown to illustrate the advances made within LS since the last 2017 LS survey (Paetzold & Specia, 2017a)

TSAR-2022$^{SharedT.}$ TSAR-2022 (Saggion et al., 2022) supplied datasets in English, Spanish, and Portuguese. These datasets housed target words in contexts taken from Wikipedia articles and journalistic texts, together with 10 candidate substitutions (approx. 20 in raw data) produced by crowd-sourced annotators from the Spain, Brazil and the UK. The candidate substitutions were ranked per their suggestion frequency. The Spanish dataset contained 381 instances, whereas the English and Portuguese datasets both contained 381 instances.

WCL-DHH Alonzo et al. (2022a) aimed to provide reading assistance to Deaf and Hard-of-hearing (DHH) adults. They annotated the original 15,000 English words of the WCL (Maddela & Xu, 2018) dataset with lexical complexity provided by 11 DHH annotators. Annotation was done via a six-point Likert scale.

4.2 Datasets in Other Languages

Spanish The ALexS-2020 shared-task (Zambrano & Ráez, 2020) produced a Spanish dataset containing 723 complex words from recorded transcripts. Merejildo (2021) provided the Spanish CWI corpus (ES-CWI). A group of 40 native-speaking Spanish annotators selected complex words within 3,887 sentences taken from academic texts. The EASIER corpus (Alarcón et al., 2021b) contains 5,310 Spanish complex words in sentences taken from newspapers with a total of 7,892 candidate substitutions. EASIER-500 is a smaller version of this dataset.

Portuguese The PorSimples dataset (Aluísio & Gasperin, 2010) houses sentences taken from Brazilian newspapers. The dataset is contains nine sub-corpora separated by degree of simplification and text genre. The PorSimplesSent dataset (Leal et al., 2018) was created from the previous PorSimples dataset. It has strong and natural simplifications of PorSimples's original sentences. SIMPLEX-PB (Hartmann & Aluísio, 2020) provides features for 730 complex words in context.

French ReSyf contains French synonyms that have been ranked by an SVM per their reading difficulty (Billami et al., 2018). It contains 57,589 instances amounting to 148,648 candidate substitutions. FrenchLys is a LS tool designed by Rolin et al. (2021). It has its own dataset with sentences sampled from french schoolbooks and a French TS dataset: ALECTOR. Substitute candidates were suggested by 20 French speaking annotators.

Japanese The Japanese Lexical Substitution (JLS) dataset (Kajiwara & Yamamoto, 2015) has 243 target words, each shown in 10 different sentences. Crowd-sourced annotators suggested and ranked candidate substitutions. The JLS Balanced Dataset (Kodaira et al., 2016) added to the prior JLS dataset to make it more representative of multiple genres and has 2,010 generalized instances. Nishihara & Kajiwara (2020) created a new dataset (JWCL & JSSL) that increased the Japanese Education Vocabulary List (JEV). It contains 18,000 Japanese words separated into three levels of difficulty: easy, medium, or difficult.

Chinese Personalized-ZH (Lee & Yeung, 2018a) provides 600 Chinese words ranked by eight learners of Chinese using a 5-point lickert-scale. HanLS was introduced by Qiang et al. (2021). It has 534 Chinese complex words. 5 native-speaking annotators provided and ranked candidate substitutions. On average, each complex word has 8 candidate substitutions.

Russian Abramov & Ivanov (2022) provide a parallel translation of the Bible instances found within the original English CompLex dataset (Shardlow et al., 2020). This new Russian LCP dataset consists of 931 distinct words shown in 3,364 different contexts. Annotation was conducted by 10 crowd-sourced annotators located in Russia using a 5-point Likert scale. A later study by Abramov et al. (2023) expanded this dataset by adding several features to each complex word.

5 Discussion and Conclusion

Since the 2017 survey on LS (Paetzold & Specia, 2017b), deep learning approaches have provided new headway in LS. MLM became a popular SG method, with the clear majority of LS studies employing a MLM objective. However, LLMs, such as GPT-3, now surpass the performance of all other approaches when fed a series of prompts, especially when using an ensemble of prompts. LS systems that employ minimal selection and ranking apart from ranking their model's prediction scores, have outperformed more technical and feature-oriented ranking methods (Table 4). However, an exception is made with regards to equivalence score (Li et al., 2022), which has been shown to be effective at SR.

Recent advances in deep learning will be incorporated into future LS systems, such as the most recent generation of LLMs. Prompt learning and LLMs, such as Llama 3 (Touvron et al., 2023), Mistral (Jiang et al., 2023), and others, have proven to deliver state-of-the-art performance becoming increasingly popular in NLP. Using an ensemble of various prompts for selection and ranking has also been shown to advance LS performance further (North et al., 2023). New metrics will also similar to equivalence score will undoubtedly be beneficial.

5.1 Concluding Remarks and Open Challenges in LS

The advent of deep learning, along with the recent advancements in LLMs and prompt engineering, has greatly transformed the way we approach LS. Past LS systems depended on statistical methods, n-gram models, lexical approaches, rule-based techniques, and word embeddings to identify complex words and replace them with simpler alternatives (Paetzold & Specia, 2017b). Now, deep-learning approaches automatically generate, select, and rank candidate substitutions using the methods described throughout this survey. However, here are various open research questions that are yet to be explored in LS research. In this section, we conclude this survey by outlining key areas for the future development of LS systems.

Evaluation

The automatic evaluation metrics that are used to evaluate LS are not perfect (See Section 3 for definitions). Potential@K and MAP@K are both lenient metrics which are designed to indicate whether a system may have the capacity to simplify something. Potential@K indicates whether any returned simplifications are correct regardless of their ranking and MAP@K compares the ranking of a set of substitutions to the gold ranking. This is helpful for tracking progress of systems and identifying whether one system or approach may be better suited to a scenario as compared to another. However in a simplification pipeline, the only important candidate is the top-ranked candidate. If an unsuitable simplification is ranked as the top candidate (leading to it being used as a replacement for the original word), this will be detrimental to the overall quality of the final text, but will not be penalised by the metrics we have discussed. Automated metrics that aim to capture quality using a single

numerical score often do not correlate with human judgments. We believe that exploring more faithful resources and metrics, as well as directly evaluating LS systems with intended user groups is a promising avenue for future work. This can be done by considering variation in data annotation instead of aggregated labels produced by multiple annotators as in most LS datasets currently available.

Wider research conducted for Text Simplification has began to combine annotation done by subject matter experts and crowd-sourced annotators, instead of fully relying on more generalized annotation protocols. Rahman et al. (2024) employed two nurse oncologists and a nurse practitioner as subject matter experts to provide highly representative sentence-level simplifications of healthcare-related material. In addition, the quality of gold labels provided by Text Simplification datasets is now being assessed through human evaluation. Gala et al. (2020) conducted several reading comprehension tests by asking dyslexic readers to recall elements of simplified and non-simplified texts. Results showed that their gold sentence-level simplifications were less likely to result in reading errors verifying that they were easier to read for their intended target demographic. We encourage LS researchers to adopt similar practices.

Explainability Lexical simplifications are inherently more explainable than sentence simplification as the operations are applied at the word level. However, the decision process about which words should be simplified is often hidden behind a black box model. Research aiming to improve explainability and interpretability of these decisions will allow researchers to better understand the challenges and opportunities of modern NLP techniques applied to LS. An example direction for future research may entail the embedding of features within prompts traditionally associated with lexical complexity to better understand the decision-making process behind LLMs for LS. Features, such as word frequency, familiarity, concreteness,and others, may be used to generate potential simplifications. Correlations between these features and the quality of the produced simplifications may in turn shed light onto which features LLMs consider when determining viable simplifications for a given target word.

Personalization Different target populations have different simplification needs thus one model does not fit all. The simplification needs of a second language learner compared to a victim of a stroke, compared to a child are each very different (Gooding & Tragut, 2022). As shown in previous research (North & Zampieri, 2023), English L2 speakers have simplification needs that are different from English L1 speakers and generally dependant on their own L1. For example, speakers of Portuguese or Spanish would have no issue with the word *necessitate* in English as in their language the word *necessitar* with the same meaning of *necessitate* exists. However, *necessitate* would be considered more complex than its synonym *need* by an English LS system due to the fact that it is longer and it has lower frequency than *need*. This would trigger an unnecessary substitution for speakers of Portuguese or Spanish L1 that not help readability and could even hinder their text comprehension given that *need* is not a word of Romance origin. Modeling these needs and using them to personalize LS systems will allow for personalized simplifications that can adequately meet the needs of specific user groups.

Personalized systems are already being developed to identify complex words for the precursor task of LCP before LS (Fig. 1). Lee & Yeung (2018a) created a personalized LCP system for identifying complex words for Chinese as foreign language learners. Koptient & Grabar (2022) implemented a LCP system to rate the complexity of medical jargon for non-expert patients. Ortiz Zambrano et al. (2019) and Ortiz Zambrano & Montejo-Ráez (2021) published a new resource and built a new LCP system for identifying complex words

spoken during university lectures for students in Ecuador. However, standalone personalized systems that generate candidate substitutes rather than identify complex words have yet to be developed. Moreover, many demographics and domains still lack their own personalized system, regardless of whether that system is designed for LCP, LS, or Text Simplification in general. This necessitates the need for further research into personalization.

Perspectivism Even within a target population, each individual will bring a unique perspective on what needs to be simplified. Systems that are able to tailor their outputs to each user's individual needs will provide adaptive simplifications of potentially higher quality. This will, in turn, improve the evaluation of LS models as discussed in this section and throughout the survey.

Real-time or adaptive machine learning has already been applied within several intelligent tutoring systems found throughout educational platforms (Kabudi et al., 2021). These systems tailor user-content in real-time to provide users with a highly personalized learning experience. Hampton et al. (2018) created the Personal Assistant for Life-Long Learning (PAL3) that uses adaptive machine learning to prevent users from forgetting learned material. Hssina & Erritali (2019) employed real-time machine learning to automatically change lesson content based on the student's profile. Troussas & Virvou (2020) implemented an adaptive recommendation system that suggests varying activities depending on the user's needs and preferences. Despite these use cases, real-time or adaptive lexical simplification has not been adopted within live intelligent tutoring services.

Integration LS is one part of the simplification puzzle, which, in turn, is part of a wider effort of improving readability. Integrating LS systems with explanation generation, text summarization (Peal et al., 2022; Xie et al., 2022), redundancy removal, and sentence splitting will further accelerate the adoption of automated simplification models. This will allow LS technology to reach a wider audience.

As of this moment, LS has only been used in a handful of use cases. LS was originally introduced to aid machine translation. LS was used to reduce the ambiguity of inputted texts. This would result in a machine translation system having higher likelihood of finding a suitable translation in the target language (North et al., 2022b). More recent use cases of LS can be seen throughout research aimed to improve the accessibility of medical-related documents (Koptient & Grabar, 2022) as well as throughout educational technology (Rets & Rogaten, 2020; Zaman et al., 2020). However, real-world production of LS technologies remains limited. We therefore encourage the adoption of LS within future technologies to increase the popularity of this field of research and to provide headway in the aforementioned research areas.

Author Contributions Kai North - Problem formulation, Writing
Tharindu Ranasinghe - Writing, Supervising
Matthew Shardlow - Writing, Supervising
Marcos Zampieri - Writing, Supervising

Funding None

Availability of data and materials Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Code availability N/A.

Declarations

Conflict of interest/Competing interests The authors declare that they have no conflict of interest.

Ethical Approval N/A.

References

Abramov, A. V., & Ivanov, V. V. (2022). Collection and evaluation of lexical complexity data for Russian language using crowdsourcing. *Russian Journal of Linguistics, 26*(2), 409–425. https://doi.org/10.22363/2687-0088-30118

Abramov, A. V., Ivanov, V. V., & Solovyev, V. D. (2023). Lexical Complexity Evaluation based on Context for Russian Language. *Computación y Sistemas*, *27*(1), 127–139.https://doi.org/10.13053/cys-27-1-4528

Al-Thanyyan, S. S., & Azmi, A. M. (2021). Automated Text Simplification: A Survey. *ACM Comput Surv, 54*(2), 1–3. https://doi.org/10.1145/3442695

Alarcón, R., Moreno, L., & Martínez, P. (2021a). Exploration of Spanish Word Embeddings for Lexical Simplification. In: Proceedings of the First Workshop on Current Trends in Text Simplification (CTTS 2021), online, URL https://ceur-ws.org/Vol-2944/paper2.pdf

Alarcón, R., Moreno, L., & Martínez, P. (2021). Lexical Simplification System to Improve Web Accessibility. *IEEE Access, 9*, 58755–5876. https://doi.org/10.1109/ACCESS.2021.3072697

Aleksandrova, D., & Brochu Dufour, O. (2022). RCML at TSAR-2022 Shared Task: Lexical Simplification With Modular Substitution Candidate Ranking. In: Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 259–26https://doi.org/10.18653/v1/2022.tsar-1.29

Alonzo, O., Lee, S., Maddela, M., et al. (2022a). A Dataset of Word-Complexity Judgements from Deaf and Hard-of-Hearing Adults for Text Simplification. In: Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 119–124, https://doi.org/10.18653/v1/2022.tsar-1.11

Alonzo, O., Trussell, J., Watkins, M., et al. (2022b). Methods for Evaluating the Fluency of Automatically Simplified Texts with Deaf and Hard-of-Hearing Adults at Various Literacy Levels. In: Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, https://doi.org/10.1145/3491102.3517566

Aluísio, S. M., & Gasperin, C. (2010). Fostering digital inclusion and accessibility: The porsimples project for simplification of portuguese texts. In: Proceedings of the NAACL HLT 2010 Young Investigators Workshop on Computational Approaches to Languages of the Americas. Association for Computational Linguistics, Los Angeles, California, pp 46–53, URl https://aclanthology.org/W10-1607

Arefyev, N., Sheludko, B., Podolskiy, A., et al. (2020). Always Keep your Target in Mind: Studying Semantics and Improving Performance of Neural Lexical Substitution. In: Proceedings of the 28th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Barcelona, Spain (Online), pp 1242–1255, https://doi.org/10.18653/v1/2020.coling-main.107

Aumiller, D., & Gertz, M. (2022). UniHD at TSAR-2022 Shared Task: Is Compute All We Need for Lexical Simplification? In: Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 251–258, https://doi.org/10.18653/v1/2022.tsar-1.28

Billami, M. B., & François, T., & Gala, N. (2018). ReSyf: a French lexicon with ranked synonyms. In: Proceedings of the 27th International Conference on Computational Linguistics. Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp 2570–2581, URL https://aclanthology.org/C18-1218

Bojanowski, P., Grave, E., Joulin, A., et al. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics, 5*, 135–146. https://doi.org/10.1162/tacl_a_00051

Brown, T. B., Mann ,B., Ryder, N., et al. (2020). Language Models Are Few-Shot Learners. In: 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada, URL https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

Cañete, J., Chaperon, G., Fuentes, R., et al. (2020). Spanish Pre-Trained BERT Model and Evaluation Data. In: Proceedings of PML4DC at the International Conference on Learning Representation (ICLR.), Virtual, URL https://arxiv.org/abs/2308.02976

Carroll, J., Minnen, G., Canning, Y., et al. (1998). Practical Simplification of English Newspaper Text to Assist Aphasic Readers. In: Proccedings of the Fifteenth National Conference on Artificial Intelligence (AAAI-98), Madison, Wisconsin, USA, URL https://users.sussex.ac.uk/~johnca/papers/aaai98.pdf

Clark, K., Luong, M. T., Le, Q. V., et al. (2020). Electra: Pre-training text encoders as discriminators rather than generators. In: 8th International Conference on Learning Representations (ICLR-2020). OpenReview.net, Addis Ababa, Ethiopia, URL https://openreview.net/forum?id=r1xMH1BtvB

Conneau, A., Khandelwal, K., Goyal, N., et al. (2020). Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, pp 8440–8https://doi.org/10.18653/v1/2020.acl-main.747

Devlin, J., Chang, M. W., Lee, K., et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, Minnesota, pp 4171–4186, https://doi.org/10.18653/v1/N19-1423

Devlin, S., & Tait, J. (1998). The use of a psycholinguistic database in the simplification of text for aphasic readers. *Linguistic Databases* pp 161–173

Ermakova, L., Bellot, P., Braslavski, P., et al. (2021). Overview of SimpleText CLEF 2021 Workshop and Pilot Tasks. In: Proceedings of the Conference and Labs of the Evaluation Forum (LREC), Bucharest, Romania, URL https://ceur-ws.org/Vol-2936/paper-199.pdf

Fandiño, A. G., Estapé, J. A., Pámies, M., et al. (2022). Maria: Spanish language models. Procesamiento del Lenguaje Natural 68:39–60. URL https://api.semanticscholar.org/CorpusID:252847802

Ferres, D., & Saggion, H. (2022). ALEXSIS: A dataset for lexical simplification in Spanish. In: Proceedings of the Conference and Labs of the Evaluation Forum (LREC), Marseille, France, pp 3582–3594, URL https://aclanthology.org/2022.lrec-1.383

Gala, N., Tack, A., Javourey-Drevet, L., et al. (2020). Alector: A Parallel Corpus of Simplified French Texts with Alignments of Misreadings by Poor and Dyslexic Readers. In: Proceedings of the Conference and Labs of the Evaluation Forum (LREC). European Language Resources Association, Marseille, France, pp 1353–1361, https://aclanthology.org/2020.lrec-1.169

Gasperin, C., Specia, L., Pereira, T. F., et al. (2009). Learning When to Simplify Sentences for Natural Text Simplification. Proceedings of ENIA https://api.semanticscholar.org/CorpusID:14656741

Gooding, S., & Tragut, M. (2022). One Size Does Not Fit All: The Case for Personalised Word Complexity Models. In: Findings of the Association for Computational Linguistics: NAACL 2022. Association for Computational Linguistics, Seattle, United States, pp 353–365, https://doi.org/10.18653/v1/2022.findings-naacl.27

Hampton, A. J., Nye, B. D., Pavlik, P.I., et al. (2018). Mitigating Knowledge Decay from Instruction with Voluntary Use of an Adaptive Learning System. In: Penstein Rosé, C,, Martínez-Maldonado, R., Hoppe, H. U., et al. (eds) Artificial Intelligence in Education. Springer International Publishing, Cham, pp 119–133, URL https://link.springer.com/chapter/10.1007/978-3-319-93846-2_23

Hartmann, N. S., Aluísio, S. M. (2020). Adaptação Lexical Automática em Textos Informativos do Português Brasileiro para o Ensino Fundamental. Linguamática 12(2):3–27. URL https://www.teses.usp.br/teses/disponiveis/55/55134/tde-29072020-161751/pt-br.php

Horn, C., Manduca, C., & Kauchak, D. (2014). Learning a Lexical Simplifier Using Wikipedia. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Association for Computational Linguistics, Baltimore, Maryland, pp 458–463, https://doi.org/10.3115/v1/P14-2075

Hssina, B., & Erritali, M, (2019), A Personalized Pedagogical Objectives Based on a Genetic Algorithm in an Adaptive Learning System. *Procedia Computer Science*, *151*, 1152–1157. https://doi.org/10.1016/j.procs.2019.04.164, the 10th International Conference on Ambient Systems, Networks and Technologies (ANT 2019) / The 2nd International Conference on Emerging Data and Industry 4.0 (EDI40 2019) / Affiliated Workshops

Jiang, A. Q., Sablayrolles, A., Mensch, A. et al. (2023). Mistral 7B. arXiv:2310.06825

Kabudi, T., Pappas, I., & Olsen, D. H. (2021). AI-enabled adaptive learning systems: A systematic mapping of the literature. *Computers and Education: Artificial Intelligence, 2*, 100017. https://doi.org/10.1016/j.caeai.2021.100017

Kajiwara, T., & Yamamoto, K. (2015). Evaluation Dataset and System for Japanese Lexical Simplification. In: Proceedings of the ACL-IJCNLP 2015 Student Research Workshop, pp 35–40, https://doi.org/10.3115/v1/P15-3006

Kajiwara, T., Matsumoto, H., & Yamamoto, K. (2013). Selecting proper lexical paraphrase for children. In: Proceedings of ROCLING. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP), Kaohsiung, Taiwan, pp 59–73, URL https://aclanthology.org/O13-1007

Kodaira, T., Kajiwara, T., & Komachi, M. (2016). Controlled and Balanced Dataset for Japanese Lexical Simplification. Association for Computational Linguistics, Berlin, Germany, pp 1–7, https://doi.org/10.18653/v1/P16-3001, URL https://aclanthology.org/P16-3001

Koptient, A., & Grabar, N. (2022). Automatic Detection of Difficulty of French Medical Sequences in Context. In: Bhatia A, Cook P, Taslimipoor S, et al (eds) Proceedings of the Conference and Labs of the Evaluation Forum (LREC). European Language Resources Association, Marseille, France, pp 55–66, URL https://aclanthology.org/2022.mwe-1.9

Leal, S. E., Duran, M. S., & Aluísio, S. M. (2018). A Nontrivial Sentence Corpus for the Task of Sentence Readability Assessment in Portuguese. In: Proceedings of the 28th International Conference on Computational Linguistics. Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp 401–413, URL https://aclanthology.org/C18-1034

Lee, J., & Yeung, C. Y. (2018a). Automatic prediction of vocabulary knowledge for learners of chinese as a foreign language. In: 2nd International Conference on Natural Language and Speech Processing (ICNLSP), pp 1–4, URL https://api.semanticscholar.org/CorpusID:46967208

Lee, J., & Yeung, C. Y. (2018b). Personalizing lexical simplification. In: Proceedings of the 28th International Conference on Computational Linguistics. Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp 224–232, URL https://aclanthology.org/C18-1019

Li, X., Wiechmann, D., Qiao, Y, et al. (2022). MANTIS at TSAR-2022 Shared Task: Improved Unsupervised Lexical Simplification with Pretrained Encoders. In: Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 243–250, https://doi.org/10.18653/v1/2022.tsar-1.27

Liu, Y., Ott, M., Goyal, N., et al. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 URL https://api.semanticscholar.org/CorpusID:198953378

Maddela, M., & Xu, W. (2018). A word-complexity lexicon and a neural readability ranking model for lexical simplification. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Brussels, Belgium, pp 3749–3760, https://doi.org/10.18653/v1/D18-1410, URL https://aclanthology.org/D18-1410

McCarthy, D., & Navigli, R. (2007). SemEval-2007 Task 10: English Lexical Substitution Task. In: Proceedings of the International Workshop on Semantic Evaluations. Association for Computational Linguistics, Prague, Czech Republic, pp 48–53, URL https://aclanthology.org/S07-1009

Melamud, O., Goldberger, J., & Dagan, I. (2016). context2vec: Learning Generic Context Embedding with Bidirectional LSTM. In: Proceedings of the Conference on Computational Natural Language Learning. Association for Computational Linguistics, Berlin, Germany, pp 51–61, https://doi.org/10.18653/v1/K16-1006, URL https://aclanthology.org/K16-1006

Merejildo, B. (2021). Creación de un corpus de textos universitarios en español para la identificación de palabras complejas en el área de la simplificación léxica. Master's thesis, Universidad de Guayaquil

Mikolov, T., Chen, K., Corrado, G., et al. (2013). Efficient Estimation of word Representations in Vector Space. In: Proceedings of the International Conference on Learning Representations, URL https://api.semanticscholar.org/CorpusID:5959482

Minaee, S., Mikolov, T., Nikzad, N., et al. (2024). Large Language Models: A Survey. arXiv preprint arXiv:2402.06196 abs/2402.06196. URL https://api.semanticscholar.org/CorpusID:267617032

Nishihara, D., & Kajiwara, T. (2020). Word Complexity Estimation for Japanese Lexical Simplification. In: Proceedings of the Conference and Labs of the Evaluation Forum (LREC). European Language Resources Association, Marseille, France, pp 3114–3120, URL https://aclanthology.org/2020.lrec-1.381

North, K., & Zampieri, M. (2023). Features of Lexical Complexity: Insights from L1 and L2 Speakers. Frontiers in Artificial Intelligence 6(1). https://doi.org/10.3389/frai.2023.1236963

North, K., Dmonte, A., Ranasinghe, T., et al. (2022a). GMU-WLV at TSAR-2022 Shared Task: Evaluating Lexical Simplification Models. In: Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 264–270, https://doi.org/10.18653/v1/2022.tsar-1.30

North, K., Zampieri, M., & Shardlow, M. (2022). Lexical Complexity Prediction: A Survey. *ACM Computing Surveys, 55*(9), 1–42. https://doi.org/10.1145/3557885

North, K., Dmonte, A., Ranasinghe, T., et al. (2023). ALEXSIS+: Improving Substitute Generation and Selection for Lexical Simplification with Information Retrieval. In: Proceedings of the Workshop on Innovative Use of NLP for Building Educational Applications. Association for Computational Linguistics, Toronto, Canada, pp 404–413, https://doi.org/10.18653/v1/2023.bea-1.33, URL https://aclanthology.org/2023.bea-1.33

Ortiz Zambrano, J., MontejoRáez, A., Lino Castillo, K. N., et al. (2019). VYTEDU-CW: Difficult words as a barrier in the reading comprehension of university students. In: The International Conference on Advances in Emerging Trends and Technologies, pp 167–176, URl https://link.springer.com/chapter/10.1007/978-3-030-32022-5_16

Ortiz Zambrano, J. A., & Montejo-Ráez, A. (2021). CLexIS2: A New Corpus for Complex Word Identification Research in Computing Studies. In: Proceedings of the International Conference on Recent Advances in Natural Language Processing. INCOMA Ltd., Held Online, pp 1075–1083, URL https://aclanthology.org/2021.ranlp-1.121

Paetzold, G., & Specia, L. (2016a). SemEval 2016 Task 11: Complex Word Identification. In: Proceedings of the International Workshop on Semantic Evaluations. Association for Computational Linguistics, San Diego, California, pp 560–569, https://doi.org/10.18653/v1/S16-1085, URL https://aclanthology.org/S16-1085

Paetzold, G., & Specia, L. (2017a). Lexical Simplification with Neural Ranking. In: Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, Valencia, Spain, pp 34–40, URL https://aclanthology.org/E17-2006

Paetzold, G. H., & Specia, L. (2015). LEXenstein: A Framework for Lexical Simplification. In: Proceedings of ACL-IJCNLP 2015 System Demonstrations. Association for Computational Linguistics and The Asian Federation of Natural Language Processing, Beijing, China, pp 85–90, https://doi.org/10.3115/v1/P15-4015, URL https://aclanthology.org/P15-4015

Paetzold, G. H., & Specia, L. (2016b). Benchmarking Lexical Simplification Systems. In: Proceedings of the Conference and Labs of the Evaluation Forum (LREC). European Language Resources Association (ELRA), Portorož, Slovenia, pp 3074–3080, URL https://aclanthology.org/L16-1491

Paetzold, G. H., & Specia, L. (2016c). Unsupervised lexical simplification for non-native speakers. In: Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023. Association for Computational Linguistics, Singapore, pp 9368–9379, https://doi.org/10.18653/v1/2023.findings-emnlp.627, URL https://aclanthology.org/2023.findings-emnlp.627

Paetzold, G. H., & Specia, L. (2017). A Survey on Lexical Simplification. *J Artif Int Res, 60*(1), 549–593. https://doi.org/10.5555/3207692.3207704

Peal, M., Hossain, M. S., & Chen, J. (2022). Summarizing consumer reviews. Intell. *Inf Syst, 59*, 193–212. https://doi.org/10.1007/s10844-022-00694-9

Peters, M. E., Neumann, M., Iyyer, M., et al. (2018). Deep Contextualized Word Representations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). Association for Computational Linguistics, New Orleans, Louisiana, pp 2227–2237, https://doi.org/10.18653/v1/N18-1202, URL https://aclanthology.org/N18-1202

Przybyła, P., & Shardlow, M. (2020). Multi-Word Lexical Simplification. In: Proceedings of the 28th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Barcelona, Spain (Online), pp 1435–1446, https://doi.org/10.18653/v1/2020.coling-main.123, URL https://aclanthology.org/2020.coling-main.123

Qiang, J., Li, Y., Yi, Z., et al. (2020). Lexical simplification with pretrained encoders. In: The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20), URL https://cdn.aaai.org/ojs/6389/6389-13-9614-1-10-20200517.pdf

Qiang, J., Lu, X., Li, Y., et al. (2021). Chinese Lexical Simplification. *IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29*, 1819–1828. https://doi.org/10.1109/TASLP.2021.3078361

Rahman, M. M., Irbaz, M. S., North, K., et al. (2024). Health Text Simplification: An Annotated Corpus for Digestive Cancer Education and Novel Strategies for Reinforcement Learning. URL https://arxiv.org/abs/2401.15043, 2401.15043

Rello, L., Baeza-Yates, R., Dempere-Marco, L., et al. (2013). Frequent words improve readability and short words improve understandability for people with dyslexia. In: Human-Computer Interaction – INTERACT 2013. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 203–219, URL https://link.springer.com/chapter/10.1007/978-3-642-40498-6_15

Rets, I., & Rogaten, J. (2020). To simplify or not? Facilitating English L2 users' comprehension and processing of open educational resources in English using text simplification. *Journal of Computer Assisted Learning, 37*(3), 705–717. https://doi.org/10.1111/jcal.12517

Rolin, E., Langlois, Q., Watrin, P., et al. (2021). FrenLyS: A Tool for the Automatic Simplification of French General Language Texts. In: Proceedings of the International Conference on Recent Advances in Natural Language Processing. INCOMA Ltd., Held Online, pp 1196–1205, URL https://aclanthology.org/2021.ranlp-1.135

De la Rosa, J., & Fernández, A. (2022). Zero-shot reading comprehension and reasoning for spanish with BERTIN GPT-J-6B. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Association for Computational Linguistics, Hong Kong, China, pp 5933–5940,https://doi.org/10.18653/v1/D19-1607, URL https://aclanthology.org/D19-1607

Saggion H, Štajner, S., Ferrés, D., et al. (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). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 271–283, https://doi.org/10.18653/v1/2022.tsar-1.31, URL https://aclanthology.org/2022.tsar-1.31

Seneviratne, S., Daskalaki, E., & Suominen, H. (2022). CILS at TSAR-2022 Shared Task: Investigating the Applicability of Lexical Substitution Methods for Lexical Simplification. In: Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 207–212, https://doi.org/10.18653/v1/2022.tsar-1.21

Shardlow, M. (2013). The CW Corpus: A New Resource for Evaluating the Identification of Complex Words. In: Proceedings of the Second Workshop on Predicting and Improving Text Readability for Target Reader Populations. Association for Computational Linguistics, Sofia, Bulgaria, URL https://aclanthology.org/W13-2908

Shardlow, M., Cooper, M., & Zampieri, M. (2020). CompLex — a new corpus for lexical complexity prediction from Likert Scale data. In: Proceedings of READI. European Language Resources Association, Marseille, France, pp 57–62, URL https://aclanthology.org/2020.readi-1.9

Shardlow, M., Evans, R., Paetzold, G., et al. (2021). SemEval-2021 Task 1: Lexical Complexity Prediction. In: Proceedings of SemEval, Online, pp 1–16, https://doi.org/10.18653/v1/2021.semeval-1.1, URL https://aclanthology.org/2021.semeval-1.1

Shardlow, M., Alva-Manchego, F., Batista-Navarro, R. T., et al. (2024). The BEA 2024 Shared Task on the Multilingual Lexical Simplification Pipeline. In: Proceedings of the Workshop on Innovative Use of NLP for Building Educational Applications. Association for Computational Linguistics, Mexico City, Mexico, pp 571–589, URL https://aclanthology.org/2024.bea-1.51

Song, J., Hu, J., Wong, L. P., et al. (2020). A New Context-Aware Method Based on Hybrid Ranking for Community-Oriented Lexical Simplification. In: Proceedings of the International Conference on Database Systems for Advanced Applications, URL https://api.semanticscholar.org/CorpusID:221839918

Souza, F., Nogueira, R., & Lotufo, R. (2020). BERTimbau: pretrained BERT models for Brazilian Portuguese. In: Proceedings of the Intelligent Systems: 9th Brazilian Conference, BRACIS 2020. Springer-Verlag, Rio Grande, Brazil, p 403-417, https://doi.org/10.1007/978-3-030-61377-8_28, URL https://doi.org/10.1007/978-3-030-61377-8_28

Specia, L., Jauhar KSujay, & Mihalcea, R. (2012). Semeval - 2012 task 1: English lexical simplification. In: Proceedings of SemEval. Association for Computational Linguistics, Montréal, Canada, pp 347–355, URL https://aclanthology.org/S12-1046

Touvron, H., Martin, L., Stone, K., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv:2307.09288 URL https://arxiv.org/abs/2307.09288

Trask, A., Michalak, P., & Liu, J. (2015). sense2vec - A Fast and Accurate Method for Word Sense Disambiguation In Neural Word Embeddings. ArXiv abs/1511.06388. URL http://arxiv.org/abs/1511.06388

Troussas, C., & Virvou, M. (2020). Introduction. In: Advances in Social Networking-based Learning: Machine Learning-based User Modelling and Sentiment Analysis. Springer International Publishing, Cham, pp 1–16, URL https://link.springer.com/book/10.1007/978-3-030-39130-0

Uchida, S., Takada, S., & Arase, Y. (2018). CEFR-based Lexical Simplification Dataset. In: Proceedings of the Conference and Labs of the Evaluation Forum (LREC). European Language Resources Association (ELRA), Miyazaki, Japan, URL https://aclanthology.org/L18-1514

Vásquez-Rodríguez, L., Nguyen, N., Ananiadou, S., et al. (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). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 218–224,https://doi.org/10.18653/v1/2022.tsar-1.23

Watanabe, W. M., Junior, A. C., Uzêda, V. R., et al. (2009). Facilita: Reading assistance for low-literacy readers. In: Proceedings of the 27th ACM International Conference on Design of Communication, p 29-36, https://doi.org/10.1145/1621995.1622002

Whistely, P. J., Mathias, S., & Poornima, G. (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). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 213–217, https://doi.org/10.18653/v1/2022.tsar-1.22

Wilkens, R., Alfter, D., Cardon, R., et al. (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). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), pp 231–238, https://doi.org/10.18653/v1/2022.tsar-1.25

Xie, F., Chen, J., & Chen, K. (2022). Extractive text-image summarization with relation-enhanced graph attention network. *Intell Inf Syst, 61*, 325–341. https://doi.org/10.1007/s10844-022-00757-x

Yang, Z., Dai, Z., Yang, Y., et al. (2019). XLNet: Generalized Autoregressive Pretraining for Language Understanding. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems, Red Hook, NY, USA, https://dl.acm.org/doi/10.5555/3454287.3454804

Yeung, C. Y., & Lee, J. (2018). Personalized text retrieval for learners of Chinese as a foreign language. In: Proceedings of the 27th International Conference on Computational Linguistics. Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp 3448–3455, URL https://aclanthology.org/C18-1292

Yimam, S. M., Biemann, C., Malmasi, S., et al. (2018). A Report on the Complex Word Identification Shared Task 2018. In: Proceedings of the Workshop on Innovative Use of NLP for Building Educational Applications. Association for Computational Linguistics, New Orleans, Louisiana, pp 66–78, https://doi.org/10.18653/v1/W18-0507, URL https://aclanthology.org/W18-0507

Zaman, F., Shardlow, M., Hassan, S. U., et al. (2020). HTSS: A novel hybrid text summarisation and simplification architecture. *Information Processing and Management, 57*(102351), 1–13. https://doi.org/10.1016/j.ipm.2020.102351

Zambrano, J. A. O., Ráez, A. M. (2020). Overview of ALexS 2020: First Workshop on Lexical Analysis at SEPLN. In: Proceedings of ALexS, URL https://api.semanticscholar.org/CorpusID:225063101