Testing Poor Man's Approach for Baseline Selection: the Case of Text Simplification

Justina Mandravickaitė, Danguolė Kalinauskaitė, Danguolė Kotryna Kapkan and Eglė Rimkienė

***Abstract*—Text simplification means a reduction of text complexity while keeping essential information. It is relevant in various areas of society, such as improving information accessibility for people with cognitive disorders, non-native speakers or children. We report initial experiments of text simplification for the Lithuanian language, focusing on the simplification of administrative or institutional texts. We developed a parallel corpus of original and simplified sentences and used a poor man's approach, a.k.a. machine translation combined with available models, as we worked in a low-resource scenario, to establish a strong baseline for further experiments in the future. Therefore, we chose 3 text simplification models for English (CTRL44, SAS and KiS) and used machine translation to translate our corpus into this language to evaluate the tendencies of these models in terms of simplification. For evaluation, we chose BLEU, ROUGE, SARI and ROLD and performed statistical analysis of simplification operations (proportions of words added, deleted and reordered).**

***Index Terms*—text simplification, model evaluation, baseline selection, plain language.**

I. INTRODUCTION

Text simplification aims to modify the content and structure of a text in order to make it easier to read and understand [8]. By the tasks performed during the text simplification process, the lexical or syntactic complexity of a text is attempted to reduce, while trying to preserve original meaning and information. Lexical simplification focuses on replacing complex words with simpler ones, while syntactic simplification is based on a system of certain rules, which comprises specific operations that are applied sentence-by-sentence to a text to make its syntactic structure simpler. The result of such a simplification of texts is Plain Language.

In its original purpose, Plain Language was first and foremost a means to open expert content for lay people (non-experts), for example, by providing people without legal or medical training access to the respective expert communication and information [15], [51]. Since the 1960s Plain Language has also been proposed for people with communication impairments [2]. It should be noted that in some European countries, it is used instead of Easy-to-Understand Language (or Easy Language) as an instrument for communicative inclusion. However, the purpose of Easy Language is different: Easy Language is designed for specific groups of people (including people with dementia-type illnesses, people affected by aphasia, prelingual hearing loss, as well as functional illiterates and language learners with and without disabilities) to access communication and information [11]. In fact, Plain Language is conceived as a continuum that bridges the gap between Easy Language on the one side and expert language (or standard language respectively) on the other [23].

In our research, Plain Language is related to institutional documents and, therefore, we seek to simplify the language of such documents for non-professionals by choosing simple words, structures, and constructions, and avoiding professional jargon. This need arises from the fact that institutional texts are usually created not by text experts, but by domain experts, which tends to have implications regarding their accessibility to a wider audience. Plain Language thus serves as a means of making complex texts accessible to non-specialists. To put it in simple words, Plain Language targets as large an audience as possible through clarity and avoiding overly technical language.

Therefore, we report our experiments of text simplification for the Lithuanian language, focusing on the simplification of administrative or institutional texts. We developed a parallel corpus of original and simplified sentences in accordance with the Plain Language rules and used a poor man's approach, a.k.a. machine translation combined with 3 text simplification models for English, to establish a strong baseline for further experiments in the future.

The remainder of this paper is structured as follows: Section II briefly reviews related works, Section III describes data, Section IV introduces methods, Section V presents the experimental setup, Section VI reports the results, Section VII provides conclusions and discusses future work.

II. RELATED WORK

A. Plain Language: Linguistic View

Most recent Plain Language definitions are increasingly less specific and focus on the outcome, i.e., text readability. However, the Plain Language movement started from Plain Language guidelines or principles that essentially are lists of rules to be applied to simplify a complicated text. These rules are related to text design and formatting, vocabulary, and syntax. Some of the plain language principles apply cross-linguistically [1], [16], [25], while others are language-specific.

The shift from specific simplification rules towards a general communicative goal is motivated by the idea that instead of producing complicated texts to simplify them as a next step, the Plain Language movement should seek to educate writers of administrative, legal, medical, scientific and other types of texts to be inclusive, understand the needs of their audience and directly produce plain language texts in the first place. This goal

All authors are with the Faculty of Informatics, Vytautas Magnus University, Kaunas, Lithuania. email: {firstname.lastname}@ vdu.lt

can be achieved when the writers and the general public are already familiar with the concept of Plain Language and with its principles, i.e., when plain language is more or less established in the community of speakers of a language.

While in some countries, such as Sweden, the Swedish Plain Language movement has been at work for decades [27], for Lithuanian, Plain Language has not been widely promoted yet, while Easy Language research and promotion is also very recent [13]. In such context, a good first step is to define Lithuanian Plain Language rules for text design and formatting, vocabulary, and syntax, and then start promoting Plain Language by automatically simplifying already available administrative texts, thus adapting them to the needs of a broad audience. This approach would allow us to quickly and efficiently process and immediately make available a large number of texts, otherwise that would not be possible due to the lack of resources. We have followed this approach in developing a parallel corpus of original and simplified sentences for Lithuanian (see Section III).

B. Automatic Text Simplification

Text simplification is the task where the vocabulary and sentence complexity of a text are simplified while maintaining the text's original information. Text simplification is relevant for improving information accessibility for people with cognitive disorders as well as for non-native speakers and children [38].

Technically text simplification can be approached in several ways. The lexical approach aims to make the text simpler by substituting difficult words with their more easily understandable alternatives. This is generally achieved by identifying complex words, choosing substitutions for them, refining these substitutions, and ranking them according to simplicity [4]. This can be done in a rule-based manner or driven by data. The approaches used for lexical text simplification range from simple synonym dictionaries [42] to machine learning [5] and deep learning [31].

The syntactic approach, on the other hand, aims to change sentence structures to ones that are simpler for readers to understand (e.g., passive sentences are changed to active sentences, and the number of clauses is lessened). The most successful tools have been rule-based, but data-driven approaches have been also developed. Rule-based syntactic simplification is achieved by parsing the sentence, identifying difficult sentence structures, transforming these structures based on the rewrite rules, and finally post-processing to make the text more coherent [18]. This approach targets specific syntactic phenomena identified by linguists [35]. The rules can either be hand-crafted or identified by patterns via training on a corpus of original and manually simplified sentences. Under a data-driven approach, either statistical or machine learning models are used, and statistical machine translation is often adopted for the text simplification task [39]. Machine translation using neural networks, such as LSTM, has been used in many studies, as text simplification task can be formulated as translation task, where a complex text is translated into a simple text (e.g., [45], [49]).

As text simplification techniques have evolved over recent decades, text simplifications that avoid long, complex, and linked sentences can now be generated by large language models [17], [41]. Moreover, transformers architecture due to the attention mechanism looks at the whole input sequence and selectively extracts essential information [44] and this feature was found useful for text simplification (e.g., [28], [50]). BERT model has been applied for lexical text simplification (e.g., [30]), text simplification using monolingual machine translation [6] or hybrid text simplification approach (e.g., [24]), to name just a few. Furthermore, T5 model has been used for controllable text simplification (e.g., [36]) as well as in text simplification in limited resources' scenarios (e.g., [26]), etc.

As models differ in performance as well as simplification "preferences", in this paper we report initial experiments of text simplification for the Lithuanian language, focusing on the simplification of administrative or institutional texts. We developed a parallel corpus of original and simplified sentences in accordance with the Plain Language rules. As we are working on the limited resources scenario, for establishing a strong baseline we applied a poor man's approach, where we combined machine translation and popular text simplification models for English. We chose several metrics for automatic evaluation. We also performed a statistical analysis to get an insight into models' tendencies towards different simplification operations.

III. DATA

The dataset consists of 2,091 entries and two columns: *original* and *simplified*. Both columns contain textual data, and there are no missing values. The *original* column contains original text phrases, while the *simplified* column contains simplified versions of the corresponding original text fragments (sentences or clauses). All phrases were simplified by 4 human experts.

The data sources were texts from a variety of Lithuanian governmental and non-governmental public institution websites that provide information to the public on services such as social benefits, migration, utilities, copyright, and various other administrative issues. These texts are written in a variety of written Lithuanian that can be described as bureaucratic – some parts are directly copied from legal acts and thus represent legal language, while other parts imitate legal language in style. There are some exceptions among these texts where the writers did try to convey the contents of laws, rules, and regulations in a more accessible language, but in general, the focus is on precision, not on comprehensibility. As a result, the texts are difficult to understand for any non-expert of the subject matter.

During the simplification process, the texts were divided into sentences or smaller elements, such as clauses, and simplified by applying a set of predefined rules. The lexical and syntactic rules applied were mainly derived from cross-linguistic [16], [25] Plain Language principles. In some cases, Plain Language principles or text simplification syntactic rules specific to languages that are fairly structurally close to

Lithuanian were taken into account [52], [12]. Certain rules, for example, the treatment of participles, were defined for Lithuanian specifically. Lexical simplification was based on frequency, according to the Lithuanian frequency dictionary [43], when in doubt.

IV. METHODS

In this study, we evaluated model efficacy to establish a strong baseline using *the poor man's approach*, with a focus on the Lithuanian language, which is considered a low-resource language. This method involves translating data from a low-resource language into a high-resource language, such as English, using machine translation tools. Subsequently, we applied pre-existing models, originally designed for the high-resource language (English), to analyze the translated data. The *poor man's approach* is typically adopted in situations where there is a significant lack of tools or resources available for a particular language, such as Lithuanian.

We evaluated three existing models for the text simplification task, all trained in English, to determine the most suitable model and approach for our study:

1. **Scientific Abstract Simplification (SAS):** This model has been fine-tuned from the foundational model, *flan-t5-large*, through a multi-task approach. The model aims to transform complex scientific abstracts into more readable versions, making scientific information more understandable. During training, each text is given a specific instruction, such as "simplify:", "summarize:", or "contextualize:", before the main content. This tells the model the desired action to apply for the text [46].
2. **CTRL44:** This controllable text simplification model is trained on the IRSD dataset[1]. It employs specific control tokens such as *<ident>*, *<para>*, *<ssplit>*, and *<dsplit>* to guide its simplification strategies. For automated selections, an operation classifier can be used to determine the most appropriate control token for the input, allowing for customized text simplifications based on specific needs [14].
3. **Keep it Simple (KiS):** It is an unsupervised text simplification method that prioritizes fluency, salience, and simplicity. It uses the k-SCST algorithm, which extends the approach of Self-Critical Sequence Training [33] and proposes multiple simplifications out of which it promotes those with above-average rewards. In testing KiS on the English news dataset, it outperformed supervised models by over 4 SARI points and helped users to complete comprehension tasks 18% faster without losing accuracy [19].

V. EXPERIMENTAL SETUP

a) *Data Translation:* In our research exploring automated translations from Lithuanian to English, we used the *deep-translator* library [9], specifically its *GoogleTranslator* class.

b) *Model Application:* For our analysis, we used three pre-trained models that were described earlier, namely:

- *Scientific Abstract Simplification (SAS)*
- *CTRL44*
- *Keep it Simple (KiS)*

It's important to note that these models were applied directly to the translated data, with no additional fine-tuning or training.

c) *Evaluation Metrics:* We used the following metrics to evaluate text simplification models:

- **BLEU (Bilingual Evaluation Understudy) Score:** Measures how many n-grams in the system's output match the reference sentences. A higher BLEU score indicates that the system's output is closer to the reference, suggesting accurate and fluent simplification [29].
- **ROUGE (Recall-Oriented Understudy for Gisting Evaluation) Score:** Measures the overlap of n-grams between the generated text and reference text. Provides a comprehensive understanding of the quality of the simplified text by evaluating its overlap with the reference [21].
- **SARI (System Aggregated Reference-less Evaluation Metric):** Evaluates the quality of simplified sentences by considering three factors: the words that should be kept, the words that should be added, and the words that should be deleted. Offers a nuanced evaluation specifically for text simplification tasks [47].
- **ROLD (Replace-only Levenshtein distance):** the Levenshtein distance, also known as the edit distance, serves as a metric to measure the dissimilarity between two string sequences. It is characterized by the minimal number of operations – specifically, insertions, deletions, or substitutions – that are necessary to transform one string into another [10].

Together, these metrics offer a detailed evaluation of the text simplification models. However, it is essential to consider them collectively to gain a well-rounded understanding of the models' effectiveness and to address the specific weaknesses of each model.

VI. RESULTS

A. Scores of Automatic Evaluation

[11] Available at https://github.com/liamcripwell/control_simp

In this section, we compare the results of the models we selected for our baseline study. Table 1 summarizes and compares the results from the quantitative perspective. Starting with the number of words in the original and simplified corpus, the results revealed that SAS and KiS tended to simplify by addition of words (the number of words increased from 36270 to 40597 and 44225 respectively after simplification), while CTRL44 had a tendency to simplify by BLEU score marks the proportion of n-grams of the output matching the ones in the original sentences in the range between 0 (no match) and 1 (perfect match). This proportion was the highest for results of CTRL44 (0.3383), while the deletion of words (the number of words decreased from 36271 to 30012 after simplification).

Table 1. Quantitative comparison of original and simplified corpus

	SAS	CTRL44	KiS
Original number of words	36 270	36 270	36 270
Number of words after simplification	40 597	30 012	44 225
BLEU	0.2435	0.3383	0.1322
ROUGE-1	0.5212	0.6987	0.4346
ROUGE-2	0.3802	0.5127	0.2684
ROUGE-L	0.4878	0.6642	0.3901
ROLD Avg.	155.0048	121.7001	178.9957
SARI	37.1902	39.0852	36.5734

BLEU score marks the proportion of n-grams of the output matching the ones in the original sentences in the range between 0 (no match) and 1 (perfect match). This proportion was the highest for results of CTRL44 (0.3383), while the lowest – for results of KiS (0.1322). BLEU score for the output of SAS fell in the middle with the value of 0.2435. Originally, BLEU score was developed for the evaluation of machine translation results, and it has been used for the evaluation of results by other NLP tasks as well, with text simplification being among them [32]. As it does not consider meaning, sentence structure and does not agree with human judgments well, many studies recommend using additional metrics to complement BLEU score [40]. Therefore, additionally we used 3 variants of ROUGE score.

ROUGE score measures the overlap of output and reference n-grams in different flavors. It measures the overlap in the range between 0 (no overlap) and 1 (perfect overlap). Originally, it was developed to evaluate the output of text summarization and machine translation [22]. ROUGE has similarities to BLEU, however, if BLEU measures the precision of n-gram match, ROUGE measures the recall [37].

Regarding unigram overlap (ROUGE-1), the output of CTRL44 got the highest value (0.6987), while the output of KiS got the lowest value (0.4346). Rouge-1 score for the output of SAS fell in the middle with the value of 0.5212. For bigram overlap (ROUGE-2) we can see the same tendency – output of CTRL44 had the highest value (0.5127), KiS – the lowest value (0.2684) and SAS fell in between the latter ones with ROUGE-2 score of 0.3802 (see Table 1). ROUGE-L (overlap of Longest Common Subsequence) continues with CTRL44 getting the highest score (0.6642), KiS – the lowest score (0.3901). The output of SAS got ROUGE-L score of 0.4878. For outputs of all 3 models ROUGE scores for unigrams and Longest Common Subsequence were higher than for bigrams.

ROUGE values correlate rather well with human judgments, it is also a language-independent metric. However, it measures syntactical matches, not semantical ones, therefore, ROUGE does not manage well different words with similar meanings, e.g., synonyms [3]. For this reason, we added SARI for our evaluation and comparison.

SARI has been specifically developed for the evaluation of text simplification results. It makes output evaluation against reference sentences based on words kept, words added and words deleted. SARI values fall between 0 and 100 (the higher the value, the better the performance of the model) [34]. Also, studies have found that SARI ranks model outputs and human judgments in the same order as a human evaluator would do it [7].

As Table 1 shows, CTRL44 got the highest score (39.0852), while the other two models scored a little lower with 37.1902 for SAS and 36.5734 – for KiS. The paper that proposed SARI metric reported scores ranging from 26 to 43 for different datasets and models in text simplification [47]. Therefore, our SARI scores fall into that range.

To evaluate the difference between original and simplified sentences we used an average score of ROLD, which is a variation of Levenshtein distance for measuring the difference between two strings by the number of single-character edits [20]. The higher the score, the more different the original and simplified data are. As Table I shows, the output of CTRL44 was the closest to the original corpus with an average distance of 121.7001. The output of KiS was the most distant from the original data with ROLD score of 178.9957. The output of SAS again fell in the middle between two other models with ROLD score of 155.0048.

B. Statistics of Simplification Operations

To have a more comprehensive comparison, we calculated the proportions of text simplification operations (words added, deleted and reordered) as well as the total number of simplification operations made by each of the tested models (see Table 2).

Table 2. Statistics of text simplification operations

	SAS	CTRL44	KiS
Simplification operations in total	193 451	105 169	247 783
Words deleted, %	28.38%	60.85%	27.05%
Words added, %	44.61%	13.68%	48.15%
Words reordered, %	27.01%	25.47%	24.80%

KiS has made the largest number of simplification operations in comparison to the other two models, while CTRL44 has made the smallest number of simplification operations. KiS performed more than twice as many text simplification operations than CTRL44 (247 783 and 105 169 respectively). SAS with 193 451 simplification operations was again in the middle, however, the output of this model was closer to KiS than to CTRL44.

Regarding proportions of text simplification operations, for SAS and KiS word addition made the largest part of total simplification operations (44.61% and 48.15% respectively). Word addition proportion for CTRL44 was only 13.68%. On the other hand, for the latter model, the largest part of simplification operations was made of word deletion (60.85%). Outputs of SAS and KiS had similar proportions of word deletion, which was twice as low in comparison to CTRL44 – 28.38% and 27.05% respectively. Word reordering proportion was similar for all 3 models – 27.01% (SAS), 25.47% (CTRL44) and 24.80% (KiS).

To summarize, SAS and KiS models had a stronger tendency for word addition, while word deletion and reordering were of similar proportions. However, CTRL44 had a stronger tendency for word deletion and a weaker tendency for word addition. Word reordering had a similar scope in all 3 models.

C. *Manual Inspection of the Results*

To have a more comprehensive view of the results, we performed manual analysis as well. Table 3 presents some examples for illustration.

Table 3. Simplified exemplary sentence

(Translated) original sentence	A one-time child benefit in the amount of 11 basic social benefits is allocated to each child born.
(Translated) manually simplified sentence	We give each child a one-time payment in the amount of 11 basic social benefits.
Automatically simplified sentence with SAS	A one-time payment is granted to a child if at least one of the child's parents (adoptive parents) or guardian, when the child's custody is established in the family, and the child meets the requirements of Article 1, Part 2, Clause 1, 2, 7, 9 or 10 of the Child Benefits Law.
Automatically simplified sentence with KiS	A one-time child benefit is allocated to each child born, in the form of an annual income allowance.
Automatically simplified sentence with CTRL44	A one-time child benefit in the amount of 11 basic social benefits is given to each child.

Manual inspection of the outputs of all 3 models revealed that SAS and KiS are prone to adding new information while simplifying original sentences which was not present in the original sentences. Table 3 presents an example of 1 sentence (it was originally written in Lithuanian, then translated into English) and the same sentence manually simplified (for reference) according to determined simplification rules. The last 3 rows of Table 3 show the results of simplifying this 1 sentence done by our selected 3 models. It can be clearly seen that SAS and KiS added new information into the output which was not present in the original sentence. Meanwhile, CTRL44 output retained the meaning of the original sentence.

VII. CONCLUSIONS

We tested 3 text simplification models (CTRL44, SAS and KiS) with a corpus translated from Lithuanian to English as we have been exploring a low-resource scenario for establishing a strong baseline. We also inspected tendencies toward simplification of the selected models. We chose several metrics for automatic evaluation (BLEU, ROUGE, SARI and ROLD). We also performed statistical analysis of simplification operations (words added, deleted, and reordered) to get an insight into models' tendencies for a particular type of simplification.

CTRL44 model scored the highest in all the evaluation metrics. SAS model tended to score lower than CTRL44 but higher than KiS, though in some cases their scores were quite close. KiS tended to have low scores in comparison to the other 2 models. As for ROLD, sentences simplified by CTRL44 were the closest to the reference sentences, while sentences simplified by KiS were the farthest from the reference sentences. Finally, CTRL44 had a strong tendency to delete words, making simplified sentences shorter, while SAS and KiS had a tendency for word addition. Finally, manual inspection of the results of all 3 models showed that SAS and KiS tend to add new information while simplifying original sentences which was not present in the original sentences.

Our future plans include further experiments in text simplification focusing on the simplification of Lithuanian administrative or institutional texts by training and fine-tuning appropriate models and using CTRL44 combined with machine translation as a baseline.

ACKNOWLEDGEMENTS

This research was funded by the Research Council of Lithuania (No. S-LIP-22-77).

REFERENCES

[1] Iso 24495-1:2023, 2020.

[2] M. Adler. The Plain Language Movement. *Oxford University Press*, 2012.

[3] M. Akter, N. Bansal, and S. K. Karmaker. Revisiting automatic evaluation of extractive summarization task: Can we do better than rouge? In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1547–1560, 2022.

[4] S. S. Al-Thanyyan and A. M. Azmi. Automated text simplification: A survey. *ACM Comput. Surv.*, 54(2), mar 2021.

[5] R. Alarcon, L. Moreno, and P. Mart´ınez. Lexical simplification system to improve web accessibility. *IEEE Access*, 9:58755–58767, 2021.

[6] S. Alissa and M. Wald. Text simplification using transformer and BERT. *Computers, Materials & Continua*, 75(2):3479–3495, 2023.

[7] F. Alva-Manchego, L. Martin, C. Scarton, and L. Specia. Easse: Easier automatic sentence simplification evaluation. In *EMNLP-IJCNLP 2019*, pages 49–54, 2019.

[8] F. Alva-Manchego, C. Scarton, and L. Specia. Cross-sentence transformations in text simplification. In *WNLP@ ACL*, pages 181–184, 2019.

[9] N. Baccouri. deep-translator, 2020. Python library.

[10] M. Bachmann. Levenshtein, 2021. Accessed 2023-10-03.

[11] L. Borghardt, S. Deilen, J. Fuchs, A.-K. Gros, S. Hansen-Schirra, A. Nagels, L. Schiffl, and J. Sommer. Neuroscientific research on the processing of easy language. *Frontiers in Communication*, 6:698044, 2021.

[12] D. Brunato, F. Dell'Orletta, G. Venturi, and S. Montemagni. Design and annotation of the first Italian corpus for text simplification. In *Proceedings of the 9th Linguistic Annotation Workshop*, pages 31–41, USA, June 2015. ACL.

[13] J. Bružaitė-Liseckienė, I. Daraškienė, and L. Vilkaitė-Lozdienė. Teksto lengvai suprantama kalba rengimo gairės. *PERLSI - Solution and Impact*, Vilniaus universitetas 70, 2021.

[14] L. Cripwell, J. Legrand, and C. Gardent. Controllable sentence simplification via operation classification. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2091–2103, USA, July 2022. ACL.

[15] H. FitzGibbon, K. King, C. Piano, C. Wilk, and M. Gaskarth. Where are biomedical research plain-language summaries? *Health Science Reports*, 3(3):e175, 2020.

[16] L. Harris, S. Kleimann, and C. Mowat. Setting plain language standards. *Clarity*, 2010.

[17] K. Jeblick, B. Schachtner, J. Dexl, A. Mittermeier, A. T. Stüber, J. Topalis, T. Weber, P. Wesp, B. O. Sabel, J. Ricke, et al. Chatgpt makes medicine easy to swallow: an exploratory case study on simplified radiology reports. *European Radiology*, pages 1–9, 2023.

[18] A. P. Kumar, A. Nayak, M. Shenoy, R. J. Manoj, and A. Priyadarshi. Pattern-based syntactic simplification of compound and complex sentences. *IEEE Access*, 10:53290–53306, 2022.

[19] P. Laban, T. Schnabel, P. Bennett, and M. A. Hearst. Keep it simple: Unsupervised simplification of multi-paragraph text. In *Proceedings of ACL-IJCNLP 2021*, 2021.

[20] H. Liang, J. Liu, W. Du, D. Jin, W. Lei, Z. Wen, and J. Lv. Knowing-how & knowing-that: A new task for machine comprehension of user manuals. In *Findings of the ACL 2023*, pages 10550–10564, 2023.

[21] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[22] C.-Y. Lin and E. Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of HLT-NAACL 2003*, pages 150–157, 2003.

[23] C. Maaß. Easy Language – Plain Language – Easy Language Plus: Balancing Comprehensibility and Acceptability. *Frank Timme*, 2020.

[24] M. Maddela, F. Alva-Manchego, and W. Xu. Controllable text simplification with explicit paraphrasing. *arXiv preprint* arXiv:2010.11004, 2020.

[25] M. Martinho. International standard for clarity - we bet this works for all languages. *Clarity Journal*, 2018.

[26] J. Monteiro, M. Aguiar, and S. Araújo. Using a pre-trained simplet5 model for text simplification in a limited corpus. *Proceedings of the Working Notes of CLEF*, 2022.

[27] A. Nord. Plain language and professional writing: A research overview. *Institutet för språk och folkminnen - The Institute for Language and Folclore, Language Council of Sweden*, 2018.

[28] K. Omelianchuk, V. Raheja, and O. Skurzhanskyi. Text simplification by tagging. *arXiv preprint* arXiv:2103.05070, 2021.

[29] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318, 2002.

[30] J. Qiang, Y. Li, Y. Zhu, Y. Yuan, and X. Wu. Lexical simplification with pretrained encoders. In *Proceedings of the AAAI*, volume 34, pages 8649–8656, 2020.

[31] J. Qiang, Y. Li, Y. Zhu, Y. Yuan, and X. Wu. LSBert: a simple framework for lexical simplification. *arXiv preprint* arXiv:2006.14939, 2020.

[32] E. Reiter. A structured review of the validity of BLEU. *Computational Linguistics*, 44(3):393–401, Sept. 2018.

[33] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE CVPR*, pages 7008–7024, 2017.

[34] V. H. A. Ribeiro, P. Cavalin, and E. Morais. A dynamic multi-criteria multi-engine approach for text simplification. In *IJCNN 2021*, pages 1–8. IEEE, 2021.

[35] C. Scarton. H. Saggion, Automatic text simplification. Synthesis lectures on human language technologies, April 2017. 137 pages. *Natural Language Engineering*, 26(4):489–492, 2020.

[36] K. C. Sheang and H. Saggion. Controllable sentence simplification with a unified text-to-text transfer transformer. In *Proceedings of INLG*. UK, Aberdeen: ACL, 2021.

[37] R. Siautama, A. C. IA, D. Suhartono, et al. Extractive hotel review summarization based on tf/idf and adjective-noun pairing by considering annual sentiment trends. *Procedia Computer Science*, 179:558–565, 2021.

[38] S. Štajner. Automatic text simplification for social good: Progress and challenges. *ACL-IJCNLP 2021*, pages 2637–2652, 2021.

[39] S. Štajner and M. Popović. Automated text simplification as a preprocessing step for machine translation into an under-resourced language. In *Proceedings of RANLP 2019*, pages 1141–1150, 2019.

[40] E. Sulem, O. Abend, and A. Rappoport. BLEU is not suitable for the evaluation of text simplification. In *Proceedings of EMNLP 2018*, pages 738–744, Belgium, 2018. ACL.

[41] R. Sun, W. Xu, and X. Wan. Teaching the pre-trained model to generate simple texts for text simplification. *arXiv preprint* arXiv:2305.12463, 2023.

[42] D. Swain, M. Tambe, P. Ballal, V. Dolase, K. Agrawal, and Y. Rajmane. Lexical text simplification using wordnet. In *ICACDS 2019*, Revised Selected Papers, Part II 3, pages 114–122. Springer, 2019.

[43] A. Utka. Dažninis rašytinės lietuvių kalbos žodynas. *VDU leidykla*, 2009.

[44] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[45] T. Vu, B. Hu, T. Munkhdalai, and H. Yu. Sentence simplification with memory-augmented neural networks. *arXiv preprint* arXiv:1804.07445, 2018.

[46] H. Wang. Scientific abstract simplification, 2023. *Hugging Face Model Hub*.

[47] W. Xu, C. Napoles, E. Pavlick, Q. Chen, and C. Callison-Burch. Optimizing Statistical Machine Translation for Text Simplification. *Transactions of the ACL*, 4:401–415, 07 2016.

[48] X. Zhang and M. Lapata. Sentence simplification with deep reinforcement learning. *arXiv preprint* arXiv:1703.10931, 2017.

[49] S. Zhao, R. Meng, D. He, S. Andi, and P. Bambang. Integrating transformer and paraphrase rules for sentence simplification. *arXiv preprint* arXiv:1810.11193, 2018.

[50] Z. Ződi. The limits of plain legal language: understanding the comprehensible style in law. *International Journal of Law in Context*, 15(3):246–262, 2019.

[51] B. N. E. C. Łukasz Debowski, Bartosz Broda. Jasnopis–a program to compute readability of texts in polish based on psycholinguistic research. *Natural Language Processing and Cognitive Science*, 2015.