Exploring Automatic Text Simplification for Lithuanian

Justina Mandravickaitė and **Eglė Rimkienė** and **Danguolė Kalinauskaitė** and **Danguolė Kotryna Kapkan**
Vytautas Magnus University
Kaunas, Lithuania
name.surname@vdu.lt

Abstract

The purpose of text simplification is to reduce the complexity of the text while retaining important information. This aspect is relevant for improving accessibility for a wide range of readers, e.g., those with cognitive disorders, non-native speakers, as well as children and the general public among others. We report experiments on text simplification for Lithuanian, focusing on simplifying texts of an administrative style to a plain language level to make it easier to understand for common people. We chose mT5 and mBART as foundational models and fine-tuned them for the text simplification task. Also, we tested ChatGPT for this task. We evaluated the outputs of these models quantitatively and qualitatively. All in all, mBART appeared to be most effective for simplifying Lithuanian text, reaching the highest BLEU, ROUGE and BERTscore scores. Qualitative evaluation by assessing the simplicity, meaning retention and grammaticality of sentences simplified by our fine-tuned models, complemented the results of evaluation metrics' scores.

1 Introduction

Text simplification means reducing the vocabulary and syntactic complexity of a text while preserving the essential information of the original text. Therefore, text simplification is relevant for improving the accessibility of information for people with cognitive disorders, as well as for non-native speakers and children (Štajner, 2021). It is important for the general public as well, especially in terms of legal and/or administrative texts as these texts provide communication between institutions and their target audiences, which have very diverse levels of reading comprehension (François et al., 2020).

In this paper, we report text simplification experiments for Lithuanian. We focus on simplifying texts of the administrative (clerical) style. The examples of communication with the general public by public authorities often use quasi-legal language, which can be ineffective in conveying information to non-specialists (François et al., 2020). Therefore such texts are difficult to understand for anyone who is not an expert in that particular field. While texts on the websites of various public administration institutions are intended to disseminate information relevant to the general public, such as social benefits, public utilities, migration, copyright, etc., there is quite often a discrepancy in terms of their declared purpose and reaching their target audience. Text simplification has the potential to address this problem as it "translates" administrative language into a less complex one in terms of vocabulary, sentence structure and other aspects while retaining the essential information from the original content.

Currently, the notion of plain language is most commonly used in written communication of governmental institutions towards the general public. It is defined as communication in which wording, structure, and design are clear so that the intended audience can easily find, understand and use the information it needs (Adler, 2012). So, in our experiments, we explore the simplification of administrative texts to the level of plain language. 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 (Maaß, 2020).

We chose mT5 and mBART as the base models and fined-tuned them, developing text simplification models for Lithuanian texts. We also tested ChatGPT for this task. We chose these models because they support Lithuanian language (many large language models do not support lower-resource languages well) and after assessing computational resources we had available for model fine-tuning. Also, our text simplification experiments performed lexical and syntactic simplification together, thus simplifying sentence structure and replacing complex words or phrases at the same step.

The rest of the paper is structured as follows: Section 2 briefly describes related work, Section 3 describes the data we used, Section 4 – methods used in our experiments, Section 5 – experimental setup, Section 6 presents results. Finally, Section 7 ends this paper with conclusions.

2 Related Work

Text simplification techniques have developed significantly in recent years from rule-based (e.g., Rennes and Jönsson (2015); Suter et al. (2016)) to data-driven approaches (e.g., Štajner and Saggion (2018); Srikanth and Li (2020)). Machine translation via neural networks, such as LSTM, also has been used in many studies because a text simplification task can be formulated as a translation task where a complex text is translated into a simple text (e.g., Vu et al. (2018); Agrawal and Carpuat (2019)).

As Transformers architecture considers the whole input sequence and selectively extracts essential information (Vaswani et al., 2017), it has been successfully used for text simplification (e.g., Zhao et al. (2018); Omelianchuk et al. (2021)), among other NLP tasks. In particular, simplifications that avoid long, complex, and linked sentences can now be generated by large language models (Jeblick et al., 2023; Sun et al., 2023a).

Recent studies have shown that these models can simplify text via the application of different techniques, such as specifying the desired reading grade level or directly indicating necessary simplification operations (Agrawal and Carpuat, 2023). BERT model has been applied for lexical text simplification (e.g., (Qiang et al., 2020)), text simplification using monolingual machine translation (Alissa and Wald, 2023) or hybrid text simplification approach (e.g., Maddela et al. (2020)), among other studies. T5 model has been used for controllable text simplification (e.g., Sheang and Saggion (2021); Basu et al. (2023); Seidl and Vandeghinste (2024)) as well as in text simplification in a situation with limited resources (e.g., Monteiro et al. (2022); Schlippe and Eichinger (2023)), to name a few. BART model has been applied not only for controllable text simplification (e.g., Sheang and Saggion (2021)) but also for paragraph-level (e.g., Devaraj et al. (2021)) and document-level text simplification (e.g., Vásquez-Rodríguez et al. (2023)) thus expanding the task. Various GPT models have been utilized for text simplification as well, especially in low-resource scenarios (e.g., Wen and Fang (2023); Deilen et al. (2023); Li et al. (2023a)).

Some of the newest models for text simplification include SIMSUM for automated document-level text simplification (Blinova et al., 2023), also, SimpleBART (Sun et al., 2023a), which reports a pre-training strategy for text simplification, and KGSimple, an unsupervised approach that uses knowledge graphs to generate compressed text (Colas et al., 2023). In addition to general text simplification, domain-specific text simplification models are emerging, e.g., for simplifying medical texts (Basu et al., 2023) or texts of particular genres (Li et al., 2023b).

What makes text simplification a complex and non-trivial task, is the lack of high-quality data sources and the need for further exploration of the low-resource scenarios (Sun et al., 2023b). Additionally, sometimes domain-specific text simplification may result in lower quality generated text as on, e.g., medical text simplification (Joseph et al., 2023; Flores et al., 2023). Finally, there are challenges related to cultural and commonsense knowledge in text simplification which requires further research in this field (Corti and Yang, 2023).

In this paper, we report experiments in text simplification for Lithuanian, focusing on simplifying administrative texts to a plain language level (Maaß, 2020), which is intended for the general public. We chose several metrics for automatic evaluation. Additionally, the results were assessed by the linguist from a qualitative perspective.

3 Data for Fine-Tuning and Testing

3.1 Data for Fine-Tuning

The final dataset for fine-tuning comprises 2,142 entries with two columns, where the first column contains original sentences or text fragments, equivalent to sentences, while the second column contains manually simplified versions of the corresponding original content[1]. All data were simplified by four experts according to guidelines which are based on the literature on plain language, i.e. simplified version of language, intended for non-specialists (general public) (Alarcon et al., 2021).

The data sources for this dataset were various Lithuanian governmental and non-governmental public institution websites that provide information

[1]The dataset is available upon request.

on services such as social benefits, migration, utilities, copyright, and other issues. The data preparation process involved dividing the texts into sentences or sentence-equivalent text fragments (e.g., clauses) and simplifying them manually following the above-mentioned simplification guidelines.

The lexical and syntactic rules that were applied were mainly derived from cross-linguistic Plain Language principles (Harris, 2010; Martinho, 2018). In some cases, Plain Language principles or text simplification syntactic rules specific to languages that have a similar grammar structure to Lithuanian were taken into account (Brunato et al., 2015; Łukasz Dębowski, 2015). 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 (Utka, 2009), when in doubt. Guidelines for Plain Lithuanian feature three different levels of proposed simplification operations and can be summarised as follows:

1. **Paragraph-level simplifications.** There are two main rules in this group. First, it is sentence shortening: sentences longer than 12 words should be divided into smaller sentences, preferably by turning embedded relative clauses into independent sentences. Second, it is list creation: where possible, homogenous elements should be transformed into vertical lists, which aid text comprehension.

2. **Lexical simplification.** Whenever possible, a more frequent synonym should be selected, disregarding the perceived formal register requirements. Metaphors and acronyms, if not particularly common, should be avoided, while obscure terms should be defined in a separate sentence.

3. **Syntactic simplification.** These include but are not limited to:

    - transformation of the passive voice into active voice;
    - replacing active participle and gerund constructions with relative clauses;
    - avoiding nominalizations;
    - preferring affirmative sentences to negation, especially avoiding double negation;
    - adding demonstrative pronouns and determiners, where possible, to increase clarity.

3.2 Data for Testing

For testing we used 100 sentences not included in our parallel corpus we used for model fine-tuning. Again, we used governmental and non-governmental public institution websites as data sources. We compiled this set following diversity criteria in terms of topics covered as well as different levels of sentence complexity.

4 Methods

4.1 mT5

The foundation of mT5 model is based on the T5 model, which stands for "Text-to-Text Transfer Transformer." Developed by Google, T5 adopts a unified text-to-text framework, where every language processing task is re-framed as a text generation problem. Key principles of the T5 model include (Zhang et al., 2021):

1. **Unified Text-to-Text Framework**: T5 treats all NLP tasks as a text generation problem, where the input and output are always text strings. This approach simplifies the architecture and allows for flexibility in handling NLP tasks.

2. **Pre-training on a Diverse Corpus**: T5 is pre-trained on a large, diverse corpus, C4 (Colossal Clean Crawled Corpus) (Dodge et al., 2021), which provides a broad understanding of language and context.

3. **Encoder-Decoder Architecture**: The model uses an encoder-decoder architecture, similar to the original Transformer model as proposed by Vaswani (Vaswani et al., 2017). The encoder processes the input text and creates a contextual representation, which the decoder then uses to generate the output text.

4. **Fine-Tuning for Specific Tasks**: While T5 is pre-trained on a general corpus, it can be fine-tuned on a specific task or language to enhance its performance.

For our specific task of Lithuanian text simplification, we used the mT5 model, a multilingual variant of the original T5 (Xue et al., 2021). The model architecture and training procedure that is used for mT5 closely follow that of T5. To train mT5, the authors introduced a multilingual variant of the C4 dataset called mC4, which comprises textual data in 101 languages drawn from the public Common Crawl web scrape. It makes mT5 model particularly suitable for languages with fewer resources (Xue et al., 2021), such as Lithuanian.

4.2 mBART

mBART, an extension of the BART (Bidirectional and Auto-Regressive Transformers) model, incorporates both auto-encoder and auto-regressive components to enhance language understanding and generation. This model is not only tailored for machine translation but also highly adaptable for tasks like text simplification. It uses a sequence-to-sequence framework based on the Transformer architecture, which includes both an encoder and a decoder (Lewis et al., 2019). The encoder processes the input text, converting it into contextual embeddings that encapsulate the nuances of the language — Lithuanian in this context. The decoder then reconstructs the text from these embeddings, aiming to produce simplified text that maintains the original meaning while being more accessible.

mBART functions as a denoising autoencoder and is one of the first models to employ a complete sequence-to-sequence framework for multilingual training by denoising full texts. It was pre-trained on a vast corpus of multilingual data using the BART methodology. This training involved a subset of 25 languages from the Common Crawl (CC) corpus (Wenzek et al., 2019), known as CC25, which includes languages from various families and features texts of different lengths. The Lithuanian portion of this dataset comprises 1,835 tokens within a 13.7 GB corpus, highlighting the model's comprehensive exposure to multilingual text (Liu et al., 2020). This extensive pre-training enables mBART to handle complex linguistic tasks, making it a robust tool for text simplification in less supported languages like Lithuanian.

4.3 ChatGPT

ChatGPT is a variant of the GPT (Generative Pre-trained Transformer) family, which itself is part of a broader class of models using transformer architectures (Yenduri et al., 2024). This design is fundamentally built on self-attention mechanisms that allow the model to process words in context to one another across a sentence or document (**?**). The model can dynamically weigh the importance of each word based on its relationship with others, making it highly effective for complex language processing tasks (Rothman, 2022). We tested ChatGPT 3.5 for Lithuanian text simplification to explore low-resource scenarios.

For our study, we used ChatGPT in its standard, as-is configuration available via OpenAI's browser interface. This meant working within the constraints of the model's pre-training, which did not specifically target Lithuanian language structures but included enough multilingual context to allow for general text manipulation tasks in Lithuanian.

4.4 Evaluaton

4.4.1 Metrics

- **BLEU (Bilingual Evaluation Understudy) Score**: measures how many n-grams in the output match the reference sentences. BLEU scores range from 0 to 1. A higher BLEU score indicates that the output is closer to the reference (Papineni et al., 2002).

- **ROUGE (Recall-Oriented Understudy for Gisting Evaluation) Score**: measures the overlap of n-grams between the simplified text and reference text in different flavors (Lin, 2004). It measures the overlap in the range between 0 (no overlap) and 1 (perfect overlap). We chose 3 variants of ROUGE: unigram overlap (ROUGE-1), bigram overlap (ROUGE-2) and Longest Common Subsequence overlap (ROUGE-L).

- **BERTscore**: BERTscore identifies words in candidate and reference phrases based on cosine similarity via the pre-trained contextual embeddings from BERT. It correlates well with human evaluation (Zhang et al., 2019).

4.4.2 Qualitative Analysis

For qualitative, expert-based evaluation of the simplification output, we used 3 common criteria: grammaticality, meaning preservation and simplicity (Nisioi et al., 2017; Alva-Manchego et al., 2020). Grammaticality (or fluency) means assessing whether the simplified text remains grammatical and understandable; meaning preservation refers to the evaluation of whether semantics (or adequacy) is preserved after the simplification; and simplicity points out to whether the simplified text is simpler than the original text (Grabar and Saggion, 2022). These criteria can be assessed without the need for reference data.

The expert has been asked to assess sentences, simplified by the models according to these 3 criteria on a scale from 1 to 5. As all 3 evaluation criteria are not equal (they go in this order: simplicity – meaning retention – grammaticality), we also asked to apply 2 other rules during the evaluation:

- The most important criterion is *simplicity*, so if according to this criterion simplified sentence gets 1, meaning retention and grammaticality are irrelevant (gets the score of 1 as well).
- If for *simplicity* a simplified sentence scores higher than 1, but *meaning retention* scores 1, then the grammaticality is scored 1 (otherwise we would get a grammatically correct but semantically incorrect sentence, i.e., unrelated to the original one).

Without such a hierarchy of criteria, there could be a paradoxical situation where models would be rewarded for simply copying the original content, while they would be penalized for attempting to simplify, although with some errors.

5 Experimental Setup

This study is aimed at the exploration of text simplification for Lithuanian. We used mT5 and mBART, which were directly fine-tuned using a dataset of complex (original) and simplified Lithuanian sentences designed by linguists. The fine-tuning focused on exploring the effects of batch size (bs) and learning rate (lr) variations on performance. The results indicated significant differences in performance between the model configurations. The mBART model with a larger batch size of 8 (mBART-bs8_lr1e-4) consistently outperformed the other configurations. On the other hand, the mT5 model with a smaller batch size (mT5-bs2_lr1e-4) demonstrated stronger performance.

The pre-trained mT5 and mBART were fine-tuned on a Lithuanian corpus, with their encoder-decoder architecture left unchanged to suit the language's nuances. ChatGPT, on the other hand, was not fine-tuned; instead, we used several prompts to test its text simplification capabilities for Lithuanian. We assessed all models using selected metrics to compare their ability to simplify text while preserving the original meaning and intent. The fine-tuning process covered eight epochs, this enabled us to track the progression and improvements in the models' performance as training continued.

6 Results

6.1 Automatic Evaluation

Firstly, we executed experiments with the mT5 and mBART models, focusing on fine-tuning and testing while adjusting key hyperparameters, namely the batch size and learning rate. The outcomes of this fine-tuning process, which was carried out over eight epochs, are visually represented in Figure 1. In this figure, the performance of the mBART model is outlined through various variants of ROUGE, with different configurations indicated by labels such as *bs-8-lr-1e-4*. These labels indicate the hyperparameters used during training — *bs* for batch size and *lr* for learning rate. Each configuration provides insights into how the model's performance is influenced by these hyperparameters.

The *bs8-lr1e-4* and *bs4-lr1e-4* results were selected as the best performing models based on their consistently higher scores across ROUGE metrics, as seen in the graphs. The larger batch size of *bs8-lr1e-4*, in particular, showed superior results, indicating effective learning and generalization capabilities for Lithuanian text simplification, while also avoiding overfitting.

In figure 2 we can see the performance during the fine-tuning of mT5 model. The ROUGE-1 graph, the configuration with a batch size of 2 and a learning rate of 1e-4 (*bs2-lr-1e-4*) achieves the highest score, suggesting that this combination is the most effective for the text simplification task out of the ones tested. The same configuration (*bs2-lr-1e-4*) leads in the ROUGE-2 and ROUGE-L graphs as well, which indicates its effectiveness not just at capturing single word overlaps but also in capturing longer phrase and sentence-level structures. Configurations with larger batch sizes and smaller learning rates improved more slowly, suggesting smaller learning rates require more epochs for comparable performance.

Table 1) summarizes the performance of each model configuration across various metrics. We selected the two best models based on their hyperparameter configurations during fine-tuning and tested them using a dataset that was not used during training and was unseen by the models.

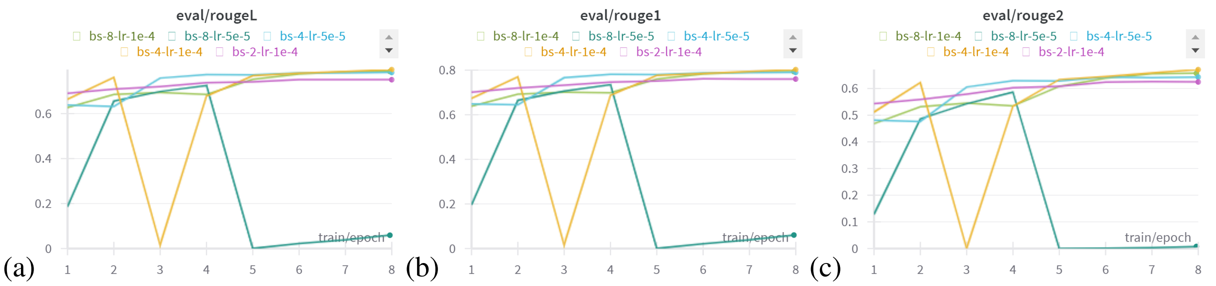

Figure 1: The mBART model's ROUGE scores during fine-tuning with different parameters: (a) ROUGE-L score, (b) ROUGE-1 score, (c) ROUGE-2 score.

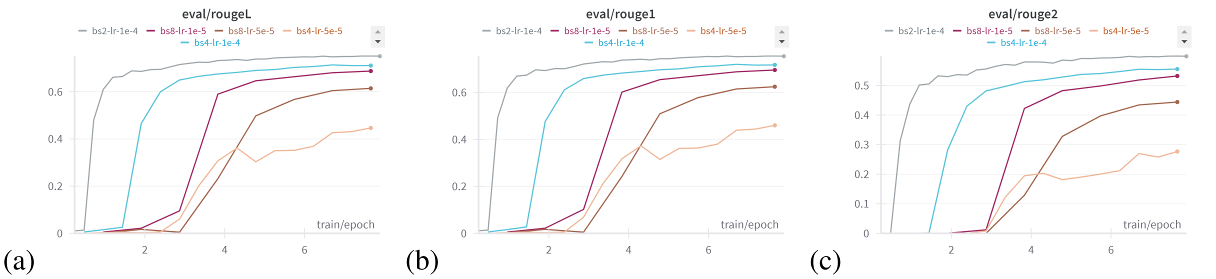

Figure 2: The mT5 model's ROUGE scores during fine-tuning with different parameters: (a) ROUGE-L score, (b) ROUGE-1 score, (c) ROUGE-2 score.

The results indicate significant differences in performance between the model configurations. The mBART model with a larger batch size of 8 (*mBART-bs8_lr1e-4*) consistently outperformed the other configurations across all metrics. This suggests that larger batch sizes may contribute to better model learning and generalization, especially for complex tasks like text simplification.

On the other hand, the mT5 model with a smaller batch size (*mT5-bs2_lr1e-4*) demonstrated stronger performance compared to its larger batch counterpart, particularly noticeable in the BLEU and ROUGE scores. This might be attributed to better handling of the nuances in a less resource-dense language like Lithuanian when trained with more focused, though smaller, data batches.

For testing ChatGPT we used 3 different prompts in the zero-shot scenario, and the average scores of the outputs are presented in Table 1. The results show that according to our selected evaluation metrics, ChatGPT performed better than or close to *mT5-bs4_lr1e-4*, but worse than the other 3 models. This shows potential, however, experimenting with prompts revealed that it is rather difficult to control the simplification to the desired level, e.g., plain language in our case.

Overall, the mBART model with the largest batch size and same learning rate setting appears most effective for simplifying Lithuanian text, highlighting its suitability for languages with fewer linguistic resources available for training.

6.2 Qualitative Evaluation

As automatic evaluation does not cover all text simplification aspects, it has been accompanied by a qualitative evaluation by the linguist, who assessed simplified sentences produced by the models. The generated sentences were assessed by their simplicity, meaning retention and grammaticality. The results are summarised in Table 2.

We can see that the highest simplicity score shared *mBART-bs8_lr1e-4* and ChatGPT (3.92/5.0). Meanwhile, *mBART-bs4_lr1e-4* and *mBART-bs8_lr1e-4* got the highest score for meaning retention (4.12/5.0). As for grammaticality, the just-mentioned *mBART-bs8_lr1e-4* achieved the highest score of 4.25/5.0.

ChatGPT showed potential, especially taking into consideration that we tested it with zero-shot prompting. However, it was rather difficult to control the desired simplification level – in our case, plain language was relevant, targeting the general public, not Easy Language that mostly aims to aid people with special needs (Maaß, 2020). Also,

Table 1: Automatic evaluation scores

	chatGPT	mT5-bs2_lr1e-4	mT5-bs4_lr1e-4	mBART-bs4_lr1e-4	mBART-bs8_lr1e-4
Average BLEU	0.359	0.5697	0.0738	0.5099	**0.6605**
Average ROUGE-1 F-score	0.4556	0.7937	0.3682	0.739	**0.8221**
Average ROUGE-2 F-score	0.228	0.7036	0.2996	0.6288	**0.7265**
Average ROUGE-L F-score	0.396	0.7844	0.352	0.7322	**0.8137**
Average BERTScore F1	0.76	0.9033	0.7137	0.8879	**0.9243**

Table 2: Qualitative evaluation scores

	Simplicity	Meaning retention	Grammaticality
mT5-bs2_lr1e-4	3.26	3.31	3.36
mT5-bs4_lr1e-4	1.99	1.89	1.88
mBART-bs4_lr1e-4	3.81	**4.12**	4.21
mBART-bs8_lr1e-4	**3.92**	**4.12**	**4.25**
chatGPT	**3.92**	3.86	3.78

there was some difficulty in controlling that information not present in an original sentence would not be added to its simplified version.

Although *mT5-bs2_lr1e-4* and *mBART-bs4_lr1e-4* were rather close in terms of automatic evaluation scores, the qualitative assessment revealed clearer differences in simplified sentences. For example, the latter model managed better in terms of grammatically correct sentences, e.g., correct case of parts of speech. Also, *mT5-bs2_lr1e-4* had a mild tendency to cut longer original sentences in the middle thus losing a part of the information.

The latter tendency, however, was rather strong in *mT5-bs4_lr1e-4*. It also struggled in terms of correct Lithuanian grammar, making common spelling mistakes, and jumbling the syntactic structure of the sentences or, in several cases, getting stuck on generating the same phrase over and over.

To summarize, qualitative evaluation added the results of automatic evaluation metrics, showing that mBART was the most successful in simplifying Lithuanian texts. It performed better than other tested text simplification models in terms of simplicity, meaning retention and grammaticality of simplified sentences.

7 Conclusions

In this paper, we report experiments on text simplification for Lithuanian with the focus of simplifying administrative-style texts to a plain language to make it easier to understand for the general public, i.e. non-specialists. We chose mT5 and mBART as foundational models and fine-tuned them for this task. Also, we tested ChatGPT to explore a low-resource scenario. We evaluated the outputs of these models quantitatively (via BLEU, ROUGE and BERTscore scores) and qualitatively (assessing simplicity, meaning retention and grammaticality of simplified sentences). All in all, mBART model appeared to be most effective for simplifying Lithuanian texts. It reached the highest BLEU, ROUGE and BERTscore scores. Qualitative evaluation results complemented the results of quantitative evaluation.

Our future plans include model improvement (e.g., exploring different fine-tuning techniques and more comprehensive experimentation in terms of training parameters) and increasing dataset size via, for example, data augmentation, to increase model performance and generalizability. Also, we plan a more comprehensive analysis of the model decision-making process to take into account such aspects as checking for factuality or model bias.

Limitations

Our study demonstrates promising results for text simplification for Lithuanian. However, it has several limitations we need to acknowledge. Firstly, we evaluated the results focusing on readability (that is, if model-simplified sentences could be easily understood by the experts who evaluated them) and retention of essential information. However, to assess the practical use of the simplified texts, evaluation and analysis could include user feedback and/or reading comprehension tests. Secondly, we limited our experiments to simplifying administrative-style texts. Therefore, models' performance may vary if given texts of different domains and genres. Also, the dataset we used for fine-tuning models is limited in size, thus, models could be improved with more diverse and comprehensive textual data. Furthermore, while quantitative evaluation metrics we used provide valuable insights, they may not fully capture the nuances related to text simplification. So, additional metrics, evaluation criteria and linguistic analysis could offer a more comprehensive assessment of simplified texts as well as models themselves. Addressing these limitations could improve the robustness and applicability of text simplification in real-world scenarios.

Acknowledgments

This research has received funding from the Research Council of Lithuania Lithuanian Studies Programme 2016–2024 under the project Automatic Adaptation of Administrative Texts in Lithuanian to the Needs of Non-Specialists, grant agreement No. S-LIP-22-77.

References

Mark Adler. 2012. *The Plain Language Movement*. Oxford University Press.

Sweta Agrawal and Marine Carpuat. 2019. Controlling text complexity in neural machine translation. 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)*, pages 1549–1564.

Sweta Agrawal and Marine Carpuat. 2023. Controlling pre-trained language models for grade-specific text simplification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12807–12819.

Rodrigo Alarcon, Lourdes Moreno, and Paloma Martínez. 2021. Lexical simplification system to improve web accessibility. *IEEE Access*, 9:58755–58767.

Sarah Alissa and Michael Wald. 2023. Text simplification using transformer and bert. *Computers, Materials & Continua*, 75(2):3479–3495.

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

Chandrayee Basu, Rosni Vasu, Michihiro Yasunaga, and Qian Yang. 2023. Med-easi: finely annotated dataset and models for controllable simplification of medical texts. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press.

Sofia Blinova, Xinyu Zhou, Martin Jaggi, Carsten Eickhoff, and Seyed Ali Bahrainian. 2023. Simsum: Document-level text simplification via simultaneous summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9927–9944.

Dominique Brunato, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2015. Design and annotation of the first Italian corpus for text simplification. In *Proceedings of the 9th Linguistic Annotation Workshop*, pages 31–41, Denver, Colorado, USA. Association for Computational Linguistics.

Anthony Colas, Haodi Ma, Xuanli He, Yang Bai, and Daisy Zhe Wang. 2023. Can knowledge graphs simplify text? In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 379–389.

Lorenzo Corti and Jie Yang. 2023. Artist: Artificial intelligence for simplified text. *arXiv preprint arXiv:2308.13458*.

Silvana Deilen, Sergio Hern'andez Garrido, Ekaterina Lapshinova-Koltunski, and Christiane Maaß. 2023. Using ChatGPT as a CAT tool in easy language translation. In *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 1–10, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Ashwin Devaraj, Byron C Wallace, Iain J Marshall, and Junyi Jessy Li. 2021. Paragraph-level simplification of medical texts. In *Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting*, volume 2021, page 4972. NIH Public Access.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. *arXiv preprint arXiv:2104.08758*.

Lorenzo Jaime Yu Flores, Heyuan Huang, Kejian Shi, Sophie Chheang, and Arman Cohan. 2023. Medical text simplification: Optimizing for readability with unlikelihood training and reranked beam search decoding. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4859–4873.

Thomas François, Adeline Müller, Eva Rolin, and Magali Norré. 2020. Amesure: a web platform to assist the clear writing of administrative texts. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 1–7.

Natalia Grabar and Horacio Saggion. 2022. Evaluation of automatic text simplification: Where are we now, where should we go from here. In *Traitement Automatique des Langues Naturelles*, pages 453–463. ATALA.

Susan Kleimann & Christine Mowat Harris, Lynda. 2010. Setting plain language standards. *Clarity*.

Katharina Jeblick, Balthasar Schachtner, Jakob Dexl, Andreas Mittermeier, Anna Theresa Stüber, Johanna Topalis, Tobias Weber, Philipp Wesp, Bastian Oliver Sabel, Jens Ricke, et al. 2023. Chatgpt makes medicine easy to swallow: an exploratory case study on simplified radiology reports. *European Radiology*, pages 1–9.

Sebastian Joseph, Kathryn Kazanas, Keziah Reina, Vishnesh Ramanathan, Wei Xu, Byron Wallace, and Junyi Jessy Li. 2023. Multilingual simplification of medical texts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16662–16692, Singapore. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Preprint*, arXiv:1910.13461.

Zihao Li, Matthew Shardlow, and Fernando Alva-Manchego. 2023a. Comparing generic and expert models for genre-specific text simplification. In *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 51–67, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Zihao Li, Matthew Shardlow, and Fernando Alva-Manchego. 2023b. Comparing generic and expert models for genre-specific text simplification. In *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 51–67.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Preprint*, arXiv:2001.08210.

Christiane Maaß. 2020. *Easy Language – Plain Language – Easy Language Plus: Balancing Comprehensibility and Acceptability*. Frank Timme.

Mounica Maddela, Fernando Alva-Manchego, and Wei Xu. 2020. Controllable text simplification with explicit paraphrasing. *arXiv preprint arXiv:2010.11004*.

Miguel Martinho. 2018. International standard for clarity - we bet this works for all languages. *Clarity Journal*.

José Monteiro, Micaela Aguiar, and Sílvia Araújo. 2022. Using a pre-trained simplet5 model for text simplification in a limited corpus. *Proceedings of the Working Notes of CLEF*.

Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P Dinu. 2017. Exploring neural text simplification models. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 2: Short papers)*, pages 85–91.

Kostiantyn Omelianchuk, Vipul Raheja, and Oleksandr Skurzhanskyi. 2021. Text simplification by tagging. *arXiv preprint arXiv:2103.05070*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the ACL 2002*, pages 311–318.

Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. 2020. Lexical simplification with pretrained encoders. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8649–8656.

Evelina Rennes and Arne Jönsson. 2015. A tool for automatic simplification of swedish texts. In *20th Nordic Conference of Computational Linguistics*, pages 317–320. Linköping University Electronic Press.

Denis Rothman. 2022. *Transformers for Natural Language Processing: Build, train, and fine-tune deep neural network architectures for NLP with Python, Hugging Face, and OpenAI's GPT-3, ChatGPT, and GPT-4*. Packt Publishing Ltd.

Tim Schlippe and Katharina Eichinger. 2023. Multilingual text simplification and its performance on social sciences coursebooks. In *International Conference on Artificial Intelligence in Education Technology*, pages 119–136. Springer.

Theresa Seidl and Vincent Vandeghinste. 2024. Controllable sentence simplification in dutch. *Computational Linguistics in the Netherlands Journal*, 13:31–61.

Kim Cheng Sheang and Horacio Saggion. 2021. Controllable sentence simplification with a unified text-to-text transfer transformer. In *Proceedings of the 14th International Conference on Natural Language Generation (INLG); 2021 Sep 20-24; Aberdeen, Scotland, UK. Aberdeen: Association for Computational Linguistics; 2021.* ACL (Association for Computational Linguistics).

Neha Srikanth and Junyi Jessy Li. 2020. Elaborative simplification: Content addition and explanation generation in text simplification. *arXiv preprint arXiv:2010.10035*.

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

Sanja Štajner and Horacio Saggion. 2018. Data-driven text simplification. In *Proceedings of the 27th International Conference on Computational Linguistics: Tutorial Abstracts*, pages 19–23.

Renliang Sun, Wei Xu, and Xiaojun Wan. 2023a. Teaching the pre-trained model to generate simple texts for text simplification. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9345–9355, Toronto, Canada. Association for Computational Linguistics.

Renliang Sun, Zhixian Yang, and Xiaojun Wan. 2023b. Exploiting summarization data to help text simplification. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 39–51.

Julia Suter, Sarah Ebling, and Martin Volk. 2016. Rule-based automatic text simplification for german. *Bochumer Linguistische Arbeitsberichte*, page 279.

Andrius Utka. 2009. *Dažninis rašytinės lietuvių kalbos žodynas*. VDU leidykla.

Laura Vásquez-Rodríguez, Matthew Shardlow, Piotr Przybyła, and Sophia Ananiadou. 2023. Document-level text simplification with coherence evaluation. In *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 85–101.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Tu Vu, Baotian Hu, Tsendsuren Munkhdalai, and Hong Yu. 2018. Sentence simplification with memory-augmented neural networks. *arXiv preprint arXiv:1804.07445*.

Zhihao Wen and Yuan Fang. 2023. Augmenting low-resource text classification with graph-grounded pre-training and prompting. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 506–516.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2019. Ccnet: Extracting high quality monolingual datasets from web crawl data. *Preprint*, arXiv:1911.00359.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mt5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498.

Gokul Yenduri, M Ramalingam, G Chemmalar Selvi, Y Supriya, Gautam Srivastava, Praveen Kumar Reddy Maddikunta, G Deepti Raj, Rutvij H Jhaveri, B Prabadevi, Weizheng Wang, Athanasios V. Vasilakos, and Thippa Reddy Gadekallu. 2024. Gpt (generative pre-trained transformer) – a comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions. *IEEE Access*, pages 1–1.

Jipeng Zhang, Hai Zhao, and Jordan Boyd-Graber. 2021. Contextualized rewriting for text simplification. In *Transactions of the Association for Computational Linguistics*, volume 9, pages 1525–1540.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Sanqiang Zhao, Rui Meng, Daqing He, Saptono Andi, and Parmanto Bambang. 2018. Integrating transformer and paraphrase rules for sentence simplification. *arXiv preprint arXiv:1810.11193*.

Bartłomiej Nitoń Edyta Charzyńska Łukasz Dębowski, Bartosz Broda. 2015. Jasnopis–a program to compute readability of texts in polish based on psycholinguistic research. *Natural Language Processing and Cognitive Science*.