Simplifying Lithuanian texts into Easy-to-Read language using large language models

Simona Kuoraitė
Vilnius University
Faculty of Mathematics and Informatics
Institute of Informatics
Vilnius, Lithuania
simona.kuoraite@mif.stud.vu.lt

dr. Valentas Gružauskas
Vilnius University
Faculty of Mathematics and Informatics
Institute of Informatics
Vilnius, Lithuania
valentas.gruzauskas@mif.vu.lt

Abstract

This paper explores the task of simplifying Lithuanian texts into Easy-to-Read language. Easy-to-Read is a form of language written in short, clear sentences and simple words, adapted for people with intellectual disabilities or limited language skills. The aim of this work is to investigate how the large language model Lt-Llama-2-7b-hf, pre-trained on Lithuanian language data, can be adapted to the task of simplifying Lithuanian texts into Easy-to-Read language. To achieve this goal, specialized datasets were developed to fine-tune the model, and experiments were carried out. The model was tested by comparing texts in their original language and texts with a prompt adapted to the task. The results were evaluated using the SARI metric for assessing the quality of simplified texts and a qualitative evaluation of the large language model. The results show that the fine-tuned model sometimes simplifies text better than a model that was not fine-tuned, but that a larger and more extensive dataset would be needed to achieve significant results, and that more research should be carried out on fine-tuning the model for this task.

1 Introduction

In recent years, there has been increasing attention on accessibility for all individuals. According to the World Health Organization, more than a billion people in the world, around 16% of the global population, have a disability (Glo, 2022). Among them, some individuals have cognitive disabilities, learning difficulties, or limited language proficiency, which makes accessing information challenging (Miesenberger and Petz, 2014). Easy to Read (ETR) language is a form of language designed to improve information accessibility by simplifying texts using short, clear sentences and simple words, adapting them for people who struggle with understanding standard texts. While ETR guidelines exist, the process of manually adapting texts remains time-consuming and resource-intensive. In Lithuania, ETR language has only recently gained recognition, and the availability of accessible content in the Lithuanian language remains limited. One of the main challenges is the lack of professionals or volunteer organizations capable of translating texts into ETR language. Without automated tools to assist in simplifying texts, the process is slow. The introduction of transformer-based architectures has significantly improved natural language processing (NLP) (Lauriola et al., 2022), enabling large language models (LLMs) like BERT (Devlin et al., 2019), GPT (Brown et al., 2020), T5 (Raffel et al., 2023) and others to generate text quicker and with higher quality (Vaswani et al., 2017). These advancements have also made it possible to adapt pre-trained models for specific tasks, such as simplifying texts to ETR. Transformers utilize a self-attention mechanism, which allows them to focus on relationships between different parts of a text sequence, understanding the importance of each word in the context. This makes them more effective than Recurrent Neural Networks (RNNs) (Karita et al., 2019). Until recently, NLP technologies for the Lithuanian language lagged behind, limiting progress in this field. However, the recent development of LLMs such as Lt-Llama-2 presents new opportunities for text simplification in Lithuanian (Nakvosas et al., 2024). This study explores how Lt-Llama-2 can be adapted for the text simplification task by fine-tuning the pre-trained Lt-Llama-2-7b-hf model for simplifying Lithuanian text into ETR.

Proceedings of the 1st Workshop on Artificial Intelligence and Easy and Plain Language in Institutional Contexts (AI & EL/PL), pages 30–37, June 23, 2025

2 Methodology

2.1 Datasets

Some Lithuanian texts are already being simplified into ETR by specialists. These texts can be found online, often as PDF files, with the original texts also being publicly available. These documents were used as the main source for fine-tuning and testing the model for the task of simplifying Lithuanian text.

For the experiments, we focused on ETR content of the 2nd level, which is the middle level of simplification, aimed at people with cognitive challenges (e.g. people with mild intellectual disabilities). These texts are also useful for people who are not native speakers of Lithuanian, but have already acquired a basic knowledge of the language. (Bružaitė-Liseckienė et al., 2021). The simplified texts were compared to the original counterparts, and datasets were created of simplified and original text pairs. While creating the dataset, the main challenge was ensuring that each record in the dataset had both the original and the corresponding simplified text, removing any texts where the simplified version added or omitted context compared to the original.

The final dataset consisted of 125 records, each containing at least one sentence. Overall, the original texts contained 2287 words, while the simplified texts contained 1974 words. The distribution of original and simplified texts, as well as their sources, is displayed in Table 1.

2.2 Text simplification

For the text simplification to ETR task, we chose the transformer based model Lt-Llama-2-7b-hf (Nakvosas et al., 2024), pre-trained on a large amount of Lithuanian data by Neurotechnology. This model was chosen because of the lack of strong LLMs pre-trained on the Lithuanian language. To evaluate whether fine-tuning the model would improve its ability to simplify texts, we conducted experiments with both the pre-trained model and the fine-tuned model. The results from both models were compared.

Additionally, we tested the effects of providing a prompt to both models during the text simplification task and compared those results as well. In the context of prompt engineering, several techniques were used to enhance the performance of the models. One common and effective method is "Think Step By Step" (Chain-of-Thought, CoT) (Kojima et al., 2023). This approach involves adding the phrase "think step by step" at the end of the prompt, guiding the model to break down complex tasks into more simple steps. Another widely used technique is "few-shot" prompting (Sivarajkumar et al., 2024). Using this technique, the provided prompt had a few examples of original and simplified texts so the model would understand the expected outcome better. In total, as shown in Table 2, 4 experiments were conducted.

2.3 Optimization algorithm

The Paged AdamW algorithm (Loshchilov and Hutter, 2019) was used for the optimization, adapted for 8-bit precision computing. This optimizer helped to significantly reduce memory usage and increase training efficiency, which was particularly important when working with a large language model and limited resources.

2.4 Learning Rate Configuration

For the model fine-tuning process, a learning rate of 3×10^{-5} was chosen to ensure a stable and balanced learning process. To further enhance adaptation, a warm-up phase was incorporated. During the first 30 steps, the learning rate was gradually increased from a very low value to the fixed value of 3×10^{-5}. This gradual increase helped prevent abrupt weight updates at the start of the training when the model was not yet sufficiently adapted to the text simplification task (Popel and Bojar, 2018).

Additional studies, such as (Smith et al., 2018), emphasize the importance of not only selecting an appropriate learning rate but also adjusting the batch size to ensure faster and more efficient model training. In line with these findings, we used a batch size of 8, expecting improved model performance and reduced fine-tuning time.

2.5 Evaluation of simplified texts

For the evaluation of simplified texts, 10% of the dataset was chosen. The texts were evaluated using both automatic metrics and by using an LLM as a judge. While LLM-based evaluation can provide a scalable alternative to human judgment (Gu et al., 2025), it may not always align with human perception and could exhibit biases or inconsistencies (Ferrer et al., 2021), in some cases, providing overly high or low scores. Therefore, incorporating human evaluation in future research would be advisable to ensure a more comprehensive understanding of the quality of the simplified texts.

Text source	Number of records	Number of words in the original text	Number of words in the simplified text
Annual report of the President of the Republic of Lithuania	31	1002	767
A guide to housekeeping and building a social circle	16	284	205
Ministry of Defence Guidelines on Emergency and Preparing for Wartime	61	766	758
A guide to the fight for women's rights	17	253	244

Table 1: Distribution of original and simplified texts in the dataset

Experiment No.	Experiment description
EXP1	Only pre-trained model
EXP2	Pre-trained model tested using prompt
EXP3	Fine-tuned model
EXP4	Fine-tuned model tested using prompt

Table 2: Experiments with the Lt-Llama-2-7b-hf model

2.5.1 Simplified text evaluation using SARI metric

To evaluate the quality of the simplified texts automatically, we used the System Output Against Reference Sentences for Text Simplification (SARI) metric (Xu et al., 2016). SARI is a widely used metric for evaluating simplified texts. It measures the quality of simplified text by assessing three key operations: addition, keeping, and deletion of words in the simplified sentence. The metric provides a mean score based on these operations.

$$\text{SARI} = d_1 F_{\text{add}} + d_2 F_{\text{keep}} + d_3 P_{\text{del}} \tag{1}$$

where

$$d_1 = d_2 = d_3 = \frac{1}{3} \tag{2}$$

$$P_{\text{operation}} = \frac{1}{k} \sum_{n=1}^{k} p_{\text{operation}}(n) \tag{3}$$

$$R_{\text{operation}} = \frac{1}{k} \sum_{n=1}^{k} r_{\text{operation}}(n) \tag{4}$$

$$F_{\text{operation}} = \frac{2 \times P_{\text{operation}} \times R_{\text{operation}}}{P_{\text{operation}} + R_{\text{operation}}} \tag{5}$$

operation $\in \{\text{del}, \text{keep}, \text{add}\}$ and k where k is the highest n-gram order.

2.5.2 Simplified text evaluation using LLM as a judge

To assess the quality of the simplified texts, we also used an LLM, specifically OpenAI's GPT-4o-mini. The model was asked to evaluate simplified texts using three criteria: clarity, context retention and simplicity. The simplified texts were assessed on a scale of 0 to 10 for each criterion, as well as an overall score. The evaluation was conducted in Google Colab by calling the API with a carefully crafted prompt, which included the original text, the professionally simplified text, and the model-generated simplified text from the experiments. The criteria for evaluation, specified in the prompt, were:

- Clarity: To assess how easily the text can be understood by people with intellectual disabilities or limited reading skills.
- Context: To assess whether the simplified text retains the meaning of the original text and whether important details have been lost.
- Simplicity: To assess whether the text is written in clear, short sentences and simple words.

3 Results

3.1 Fine-tuning Lt-Llama-2-7b-hf

Before conducting the experiments, we fine-tuned the pre-trained Lt-Llama-2-7b-hf model with 90% of the data from the created dataset. Figure 1 displays how Cross-Entropy Loss changes in the process of fine-tuning the model for both training and validation datasets which were split into 80% and 10% of the original dataset size respectively.

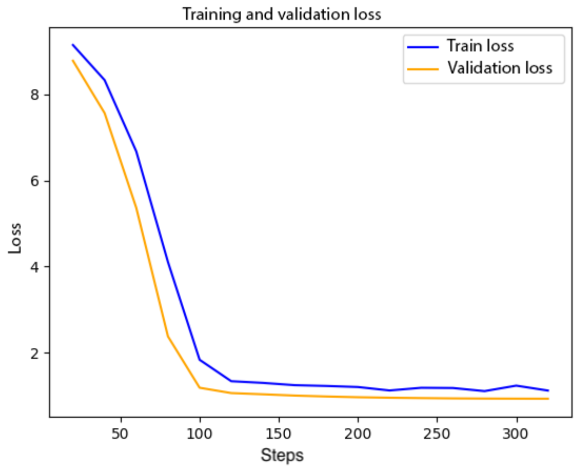

Figure 1: Variation of model learning loss for training and validation datasets over learning time

The X axis displays the number of iterations, which are computed as follows:

$$\text{Total training steps} = \frac{N}{B \times G} \times E \qquad (6)$$

where:

- N is the number of examples in the dataset,
- B is the batch size,
- G is the gradient accumulation steps,
- E is the number of epochs.

While the Y axis displays the value of the loss. In the graph, we could observe that at the start of the fine-tuning process, the loss is high for both training and validation data. However, the loss decreases rapidly subsequently, indicating that the model quickly learns to discriminate between a large number of text features and then the learning process slows down. Although in the further iterations, the loss is decreasing very slowly, it decreases for both datasets equally, which lets us assume that even though the dataset is small, the model is not memorizing text features and overfitting.

3.2 Evaluation of simplified texts

After fine-tuning the model for the text simplification task, the model was given data from the test dataset to simplify. A preliminary analysis of the simplified texts for all 4 experiments shows that regardless of fine-tuning, the results were sometimes aleatory with completely unrelated texts like not-simplified, just paraphrased texts or rewriting the prompt. That could happen because of the small training dataset or the inability of the model to adapt for this specific task. On the other hand, in some of the generated texts, it was noticeable that sentences were a slightly shorter or some more simple words were used.

Experiment No.	SARI result mean
EXP1	52.063
EXP2	44.233
EXP3	52.435
EXP4	55.648

Table 3: The mean of SARI results for test dataset for different experiments

3.3 Evaluation using SARI

Table 3 displays SARI evaluations for all 4 experiments. The results for SARI are expressed between 0 and 100, and as the table shows, they remain relatively low. Nevertheless, it is noticeable that the fine-tuned model performs slightly better than the pre-trained LLM, pointing towards potential benefits from fine-tuning. Similarly, the results where higher when including a prompt for simplifying text, indicating that using a prompt can enhance the results of the text simplification task.

3.4 Evaluation using GPT-4o-mini as a judge

After evaluating the model using the SARI metric, a more subjective assessment was performed using GPT-4o-mini as an evaluator to gain further insights into the quality of the simplified texts. The results, presented in Table 4, show ratings on a scale from 1 to 10, assigned by GPT-4o-mini for two samples created using the four different techniques.

The evaluations indicate that the results of the different experiments vary from average (e.g., 4/10 for EXP1, Sample 1) to very high (e.g., 9/10 for EXP2, Sample 2). The lowest ratings were given to the results of EXP1 and EXP3, particularly for Sample 1, which was poorly rated across all criteria, with context being the most negatively impacted. This might seem like a reasonable evaluation, as the original text contained important information on where to seek help in case of emergency, which was missing in the simplified version.

On the other hand, the highest overall ratings for simplified texts were given to the results of EXP2 and EXP4 with Sample 2. These results highlight a critical insight: large language models like GPT-4o-mini, while powerful, may not always be fully reliable as evaluators. The results for these experiment and sample pairs were actually just rewrites of the original prompt instead of simplified sentences, but the model evaluated them as successfully simplified.

4 Discussion

During the fine-tuning of the Lt-Llama-2-7b-hf model, the loss values showed a gradual decrease, demonstrating that the model effectively learned to adapt to both training and validation data. As the number of iterations increased, the loss decreased quickly at first, but began to slow down, indicating that the model was learning the features necessary for simplification tasks.

When testing the fine-tuned model on the text simplification task, the results indicated that the model was not yet fully adapted to simplify texts efficiently. While some outputs were simplified to shorter sentences with simpler words, other results were unclear or consisted of paraphrased text or the provided prompt, deviating from the expected simplification. Both the SARI metric and GPT-4o-mini evaluations confirmed that the fine-tuned and non-fine-tuned models produced similar results, with relatively low scores across all experiments. The best results were obtained for the EXP4 experiment with an average SARI value on 55.648. This shows that the adapted model balanced word addition, keeping, and deletion better than the non-adapted model, but achieved only the average possible SARI score. For the GPT-4o-mini model evaluation, EXP2 and EXP4 performed best overall, particularly for Sample2, which consistently received higher ratings (up to 9/10 for clarity and simplicity). In contrast, Sample1 results across all experiments remained noticeably weaker, indicating that model performance varied significantly depending on input content rather than experimental configuration alone.

In terms of model fine-tuning, in this study, we focused on fine-tuning the Lt-Llama-2 model for the text simplification task using the Paged AdamW optimizer and a gradual learning rate warm-up to ensure memory efficiency and stable training on a relatively small dataset. While this approach is widely used and effective for similar tasks, it's worth noting that alternative fine-tuning strategies, such as weight freezing or layer-wise learning rate adjustment, could also be explored. These techniques can help optimize model performance and further reduce memory consumption, particularly when working with larger datasets. For instance, freezing certain layers during fine-tuning (or "salting" the weights) could enable more efficient transfer learning by focusing on specific aspects of the model's knowledge while avoiding unnecessary updates. This is especially beneficial when training on smaller datasets, as it prevents overfitting and ensures faster convergence.

Moreover, there are additional methods worth considering for improving fine-tuning an LLM for the text simplification task, such as transfer learning and progressive document-level simplification. Transfer learning allows fine-tuning a pre-trained model to a new task by leveraging knowledge from larger, more diverse datasets, which could be explored in future work. Similarly, progressive simplification, as discussed in studies like (Fang et al., 2025), emphasizes simplifying text at different levels of complexity, potentially improving model accuracy and usability, especially for challenging linguistic tasks.

By integrating these approaches, we could not only improve model performance for simplifying Lithuanian texts, but also expand its applicability to a broader range of texts and simplification levels. As suggested by (Parthasarathy et al., 2024), incorporating these methods into a fine-tuning pipeline could help mitigate challenges and lead to breakthroughs in text simplification tasks, making the model more robust and adaptable to various types of input.

As part of our future research, we are preparing a proposal to extend this work by incorporating Human Feedback Reinforcement Learning (HF-RL). This approach would allow us to fine-tune the model using direct human feedback, improving its ability to generate more accurate and useful simplifications. Additionally, we plan to explore multi-stage fine-tuning, where we will combine open-source datasets with our own domain-specific data. This will help us create a more comprehensive fine-tuning process, potentially improving model performance in text simplification tasks.

4.1 Comparison to other research

In comparison to similar research, while text simplification in the Lithuanian language remains limited, a notable study focused on the simplification of administrative texts into plain language rather than

Experiment No.	**Example No.**	Clarity	Context	Simplicity	General Rating
EXP1	Sample1	4/10	3/10	5/10	4/10
EXP1	Sample2	7/10	5/10	8/10	6/10
EXP2	Sample1	6/10	5/10	7/10	6/10
EXP2	Sample2	9/10	7/10	9/10	8.5/10
EXP3	Sample1	4/10	3/10	5/10	4/10
EXP3	Sample2	8/10	7/10	9/10	8/10
EXP4	Sample1	4/10	5/10	6/10	5/10
EXP4	Sample2	8/10	7/10	9/10	8/10

Table 4: Example results evaluation by GPT-4o-mini

Study	Language	Model	Dataset Size
(Mandravickaitė et al., 2025)	Lithuanian	T5, mBART, Lt-Llama-2	~2142 pairs
(Martínez et al., 2024)	Spanish	LLaMA-2	~2081 pairs
(Barbu et al., 2025)	Estonian	LLaMA 3.1, OpenNMT	~50,416 pairs

Table 5: Overview of fine-tuning datasets used in related studies

ETR (Mandravickaitė et al., 2025). Their approach involved fine-tuning transformer-based models, including T5, mBART, and Lt-Llama-2—the only non-multilingual model in the task - on a dataset of complex and simplified administrative texts. The results indicated that "in many cases, instead of simplifying the provided sentences, the fine-tuned model simply expanded them by adding information that was not present in the original complex sentences". While T5 and mBART showed better results, with SARI scores ranging from 54.12 to 72.98, the fine-tuned Lt-Llama-2 underperformed. The study emphasized the importance of high-quality training data and task-specific fine-tuning challenges, also highlighted in our research.

In addition to the Lithuanian-focused study, two significant studies in other languages provide valuable comparisons for our work. The first study (Martínez et al., 2024) investigated simplifying Spanish texts into ETR using the Llama-2 model. Their approach involved fine-tuning the Llama-2 model on complex and simplified Spanish sentences, including a translation approach, where complex Spanish text was translated to English, simplified, and translated back to Spanish. The results showed improvements in readability and accessibility, with qualitative evaluations confirming the model's ability to simplify content while preserving its meaning.

In comparison, while our study focuses on the Lithuanian language, the successful application of Llama-2 for text simplification to ETR in Spanish suggests the model's flexibility. Although the datasets and languages differ, the findings imply that with adequate fine-tuning and dataset preparation, Llama-2 could potentially be applied to Lithuanian text simplification tasks as well, as well as opening the possibility of simplifying text using a translation technique.

Another relevant study (Barbu et al., 2025) investigated Estonian text simplification using LLMs. This research is relevant since the Estonian language, like Lithuanian, is a less-resourced language with limited LLM tools. The study involved fine-tuning Llama on a custom dataset, combining both translated data and GPT-4-generated simplifications, and comparing it to other LLMs such as DRESS, OpenNMT, and T5. A comparison of Llama 3.1 and OpenNMT models revealed that while OpenNMT achieved a slightly higher BLEU score (30.05 vs. 27.04), indicating better alignment with reference texts, Llama 3.1 outperformed OpenNMT on the SARI metric (49.72 vs. 47.43), suggesting more effective text simplification. Additionally, Llama 3.1 had a slightly lower FKGL score (8.71 vs. 9.02), indicating slightly easier readability. Despite these similarities in automatic metric performance, manual evaluations by three native Estonian speakers rated Llama 3.1 significantly higher (3.03 vs. 1.6 on a 4-point scale), demonstrating its superior ability to simplify texts to ETR standards in terms of grammar, readability, meaning preservation, and simplification.

As summarized in Table 5, our dataset was considerably smaller compared to other studies in the field. While (Mandravickaitė et al., 2025) used administrative texts and (Martínez et al., 2024) leveraged both real and translated data, our study was limited to a smaller corpus of ETR-specific texts. This difference in dataset size and diversity may partially explain the lower performance of our fine-tuned model, especially since effective LLM adaptation often requires tens of thousands of training examples to generalize well.

5 Conclusion

In conclusion, even though large language models such as Lt-Llama-2-7b-hf have significant potential for text simplification tasks, the results showed that these models, even when fine-tuned, require further refinement to perform well in text simplification tasks. While fine-tuning loss results indicated that the model was adapting to the text simplification task and the fine-tuned model achieved better results than the non-fine-tuned one, the overall performance was still moderate, with the highest SARI score of 55.648 for the EXP4 experiment. To improve the model's performance, further experiments should focus on optimizing the fine-tuning parameters and increasing the dataset size. This would allow the model to adapt to a wider range of text simplification tasks.

References

2022. *Global Report on Health Equity for Persons with Disabilities*, 1st ed edition. World Health Organization, Geneva.

Eduard Barbu, Meeri-Ly Muru, and Sten Marcus Malva. 2025. Improving Estonian Text Simplification through Pretrained Language Models and Custom Datasets. *arXiv preprint*. Citation-key: simplifying-estonian-text arXiv:2501.15624 [cs].

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. *arXiv preprint*. ArXiv:2005.14165 [cs].

Justina Bružaitė-Liseckienė, Inga Daraškienė, and Laura Vilkaitė-Lozdienė. 2021. Teksto lengvai suprantama kalba rengimo gairės.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint*. ArXiv:1810.04805 [cs].

Dengzhao Fang, Jipeng Qiang, Yi Zhu, Yunhao Yuan, Wei Li, and Yan Liu. 2025. Progressive Document-level Text Simplification via Large Language Models. *arXiv preprint*. ArXiv:2501.03857 [cs].

Xavier Ferrer, Tom van Nuenen, Jose M. Such, Mark Coté, and Natalia Criado. 2021. Bias and Discrimination in AI: a cross-disciplinary perspective. *IEEE Technology and Society Magazine*, 40(2):72–80. ArXiv:2008.07309 [cs].

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. A Survey on LLM-as-a-Judge. *arXiv preprint*. ArXiv:2411.15594 [cs].

Shigeki Karita, Nanxin Chen, Tomoki Hayashi, Takaaki Hori, Hirofumi Inaguma, Ziyan Jiang, Masao Someki, Nelson Enrique Yalta Soplin, Ryuichi Yamamoto, Xiaofei Wang, Shinji Watanabe, Takenori Yoshimura, and Wangyou Zhang. 2019. A Comparative Study on Transformer vs RNN in Speech Applications. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 449–456, SG, Singapore. IEEE.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large Language Models are Zero-Shot Reasoners. *arXiv preprint*. ArXiv:2205.11916 [cs].

Ivano Lauriola, Alberto Lavelli, and Fabio Aiolli. 2022. An introduction to Deep Learning in Natural Language Processing: Models, techniques, and tools. *Neurocomputing*, 470:443–456.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. *arXiv preprint*. ArXiv:1711.05101 [cs].

Justina Mandravickaitė, Eglė Rimkienė, Danguolė Kotryna Kapkan, Danguolė Kalinauskaitė, Antanas Čenys, and Tomas Krilavičius. 2025. Automatic Text Simplification for Lithuanian: Transforming Administrative Texts into Plain Language. *Mathematics*, 13(3):465. Citation-key: text-simplification-in-Lithuanian.

Paloma Martínez, Alberto Ramos, and Lourdes Moreno. 2024. Exploring Large Language Models to generate Easy to Read content. *Frontiers in Computer Science*, 6:1394705. Citation-key: simplifying-in-spanish.

Klaus Miesenberger and Andrea Petz. 2014. Easy to Read on the Web – State of the Art and Research Directions. *Procedia Computer Science*, 27:318–326.

Artūras Nakvosas, Povilas Daniušis, and Vytas Mulevičius. 2024. Open Llama2 Model for the Lithuanian Language. *arXiv preprint*. ArXiv:2408.12963 [cs].

Venkatesh Balavadhani Parthasarathy, Ahtsham Zafar, Aafaq Khan, and Arsalan Shahid. 2024. The Ultimate Guide to Fine-Tuning LLMs from Basics to Breakthroughs: An Exhaustive Review of Technologies, Research, Best Practices, Applied Research Challenges and Opportunities. *arXiv preprint*. ArXiv:2408.13296 [cs].

Martin Popel and Ondřej Bojar. 2018. Training Tips for the Transformer Model. *The Prague Bulletin of Mathematical Linguistics*, 110(1):43–70. ArXiv:1804.00247 [cs].

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *arXiv preprint*. ArXiv:1910.10683 [cs].

Sonish Sivarajkumar, Mark Kelley, Alyssa Samolyk-Mazzanti, Shyam Visweswaran, and Yanshan Wang. 2024. An Empirical Evaluation of Prompting Strategies for Large Language Models in Zero-Shot Clinical Natural Language Processing: Algorithm Development and Validation Study. *JMIR Medical Informatics*, 12:e55318.

Samuel L. Smith, Pieter-Jan Kindermans, Chris Ying, and Quoc V. Le. 2018. Don't Decay the Learning Rate, Increase the Batch Size. *arXiv preprint*. ArXiv:1711.00489 [cs].

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

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing Statistical Machine Translation for Text Simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.