Computational Linguistics in the Netherlands Journal 14 (2025) 85-111 Submitted 12/2024; Published 07/2025

Generating Simplified Dutch Texts for Pupils Through N-Shot Learning

Wout Sinnaeve WOUT.SINNAEVE@LIVE.BE
Joni Kruijsbergen* JONI.KRUIJSBERGEN@UGENT.BE
Orphée De Clercq* ORPHEE.DECLERCQ@UGENT.BE

*Language and Translation Technology Team, Ghent University, Belgium

Abstract

Text simplification (TS) aims to improve text readability while retaining its original meaning, aiding individuals with limited literacy skills or reading comprehension challenges. While substantial progress has been made in TS for English, there is a notable lack of research for Dutch, in part caused by the absence of Dutch parallel simplification corpora. This study investigates the effectiveness of N-shot learning using generative open-source large language models (LLM) for TS in Dutch, circumventing the need for extensive parallel corpora. Various N-shot learning techniques are assessed for their performance in generating simplified Dutch texts for pupils. The readability and appropriateness of these texts is evaluated using automatic readability assessment models and human evaluations. Results indicate that while one-shot learning using a Dutch monolingual generative LLM shows the highest performance among the tested methods, the overall effectiveness is poor, with metrics close to random guess probabilities. Human evaluation further highlights significant issues and that the generated outputs often do not match the intended readability levels and appropriateness for specific educational contexts. These findings suggest that current N-shot learning methodologies are not effective for Dutch TS, emphasising the need for more refined approaches and better training data to improve performance in this task.

1. Introduction

Text Simplification (TS) refers to the process of modifying linguistic features of a text with the goal of improving its readability while retaining its original meaning. It is used to aid individuals with limited literacy skills, such as children or non-native speakers (Paetzold and Specia 2016, Xia et al. 2016). Additionally, it can be used to aid individuals with various reading comprehension challenges, such as aphasia (Carroll et al. 1998), dyslexia (Rello et al. 2013) and autism (Evans et al. 2014).

Over the years, various approaches have been developed for TS, each with its own set of methodologies and applications. Initially, TS research focused on lexical simplification, which involves substituting complex words with simpler synonyms (Al-Thanyyan and Azmi 2021). This was followed by syntactic simplification, which addresses the complexity of sentence structures (Shardlow 2014). More recently, the advent of neural networks and machine translation techniques has significantly advanced the field, allowing more nuanced and effective simplifications (Kulmizev and Nivre 2022).

Despite these advancements, the majority of TS research and resources have been centered on the English language (Al-Thanyyan and Azmi 2021, Seidl and Vandeghinste 2024). This focus has resulted in a lack of comparable resources and models for other languages, including Dutch. Furthermore, the absence of a large parallel corpus of simplified Dutch texts has particularly hindered the development of effective TS models for Dutch (Daelemans et al. 2004, Sevens et al. 2018, Seidl and Vandeghinste 2024). Consequently, researchers have explored alternative methods, such as leveraging neural machine translation to create synthetic Dutch parallel corpora from English datasets (Seidl and Vandeghinste 2024).

In recent years, the field of Natural Language Processing (NLP) has undergone a paradigm shift with the rise of large language models (LLM). These models, especially the current generation of

generative (decoder) models, have revolutionised the ability to perform tasks directly through N-shot learning (Brown et al. 2020). This advancement has opened new avenues for NLP tasks, offering potential solutions that do not rely heavily on extensive parallel corpora (Radford et al. 2019).

With this in mind, the current study aims to investigate whether N-shot learning using LLMs can effectively address the issue of Dutch TS, i.e., the deliberate modification of text to meet specific readability criteria without altering the text's core meaning or content (Agrawal and Carpuat 2024). This involves evaluating various N-shot learning methods for the task of TS in Dutch, utilising both multilingual and monolingual open-source language models. By addressing these objectives, this study seeks to answer the following research question: "Can large language models be leveraged to perform text simplification aimed towards pupils through N-shot learning?" Specifically, this research seeks to:

1. Assess the performance of few-shot, one-shot and zero-shot learning techniques in generating simplified Dutch texts aimed at Dutch pupils.

2. Assess the quality of the generated texts through manual evaluation of the output, focusing on common errors and limitations in the outputs of the LLMs.

3. Evaluate the reliability and suitability of the generated simplified texts through automatic readability assessment models developed for the purpose of this study, and human evaluations by teachers.

Our results demonstrate that one-shot learning using synthetic examples with the Dutch monolingual LLM GEITje-ultra (Vanroy 2024) achieved the highest relative performance among all tested approaches. However, the overall effectiveness of all evaluated methods proved to be limited. Automatic readability assessments, coupled with human evaluations, indicate substantial issues with the reliability, coherence and appropriateness of the simplified outputs. The findings suggest that the current capabilities of N-shot learning with LLMs fall short in achieving text simplification for Dutch texts targeted toward pupils.

The remainder of this paper is organised as follows: Section 2 reviews related work in TS, focusing on Dutch-language efforts and controlled simplification methods while also introducing prior work on readability prediction and N-shot learning. Section 3 describes the data collection and preparation processes. Section 4 details the methodology, including the N-shot learning approaches and evaluation techniques. In Section 5 the results are presented together with a comparative analysis of automatic assessments and human evaluations. Finally, Section 6 discusses the implications of the findings and outlines directions for future research.

2. Related Work

2.1 Main Approaches in Text Simplification

Automatic text simplification (TS) has a long-standing history, with it being the subject of research since the late 90s. The field's development can be categorised into four primary approaches (Al-Thanyyan and Azmi 2021). The first of these, lexical simplification, proposed TS by identifying and substituting complex words with simpler, more understandable, synonyms while maintaining the original syntax. This approach can be further subdivided into a rule-based approach, relying on predefined linguistic rules and a data-driven approach utilising machine learning strategies to learn lexical simplification rules from parallel corpora (Al-Thanyyan and Azmi 2021).

Second, syntactic simplification aims to simplify complex syntactic structures in a text, generally involving three steps. First, a sentence is analysed to determine its structure and parse tree. Subsequently, modifications are made to the parse tree according to a set of rewrite rules. Finally, a

regeneration phase might be implemented, where additional changes are made to enhance a text's coherence, relevance and readability (Shardlow 2014). Syntactic simplification can also be subdivided into a rule-based and data-driven approach (Al-Thanyyan and Azmi 2021).

Third, inspired by the success of machine translation (MT) methodologies, researchers began treating TS as a monolingual MT problem. Initially, these studies would rely on statistical machine translation (SMT) (Al-Thanyyan and Azmi 2021). However, in 2016, with the transition to neural networks as the primary modelling paradigm (Kulmizev and Nivre 2022), a TS method using neural machine translation (NMT) was proposed (Wang et al. 2016). In most cases, both SMT and NMT methods rely on parallel data, either in the form of parallel corpora or aligned sentence pairs (Al-Thanyyan and Azmi 2021, Seidl and Vandeghinste 2024).

Finally, while evolutions within these approaches from rule-based to data-driven methods resolved many of the performance issues presented by the rule-based models, data driven approaches were directly dependent on the availability of parallel data (Al-Thanyyan and Azmi 2021). With this in mind, researchers developed hybrid systems, combining hand-written rules for common syntactic simplifications and a data-driven lexical simplification module in an attempt to mediate these limitations (Siddharthan and Mandya 2014).

Particularly with the advent of neural networks and machine learning techniques, the state of the art in TS has advanced significantly (Zhang and Lapata 2017). Modern TS approaches leverage deep learning models, such as transformers, which have shown significant performance in generating simplified texts while maintaining semantic integrity (Scarton and Specia 2018, Martin et al. 2022). These models are trained on large corpora, allowing them to analyse and process complex linguistic patterns and produce high-quality simplified outputs (Wang et al. 2016).

With regard to languages, TS is most extensively researched and developed in English due to the abundance of available data and resources. However, there is a growing interest in TS for other languages, including Spanish, Japanese, German and Dutch (Seidl and Vandeghinste 2024). In line with this, the field has also seen advancements in the development of multilingual TS models. The MUSST model, for example, employs a modular approach for simplifications in English, Italian and Spanish and can be extended to other languages (Scarton et al. 2017).

2.2 Controlled Text Simplification

While TS works well to provide suitable output for all types of simplifications, simplified texts are consumed by a wide range of audiences with varying needs (Martin et al. 2022). Therefore, controlled TS methods were developed with the goal of controlling the degree (e.g. specified sentence length, word length) and type (e.g. lexical, syntactic) of simplification outputted by TS models, by allowing users to model the complexity of the output language towards a specific audience (Sheang and Saggion 2021, Kew and Ebling 2022, Agrawal and Carpuat 2024).

Controlled TS modifies the output by employing control tokens, special tokens representing a target attribute (Agrawal and Carpuat 2024). Depending on how and when these tokens are employed, the distinction can be made between decoding-based and learning-based controlled TS approaches (Martin et al. 2022, Seidl and Vandeghinste 2024).

In decoding-based controlled TS, constraints are introduced to the system during inference (Martin et al. 2022). Output length, for example, can be constrained by inserting a minimum or maximum length control token. At the sequence level, such tokens could be employed to prevent the decoder from generating an end-of-sequence token before reaching the predefined length (Kikuchi et al. 2016, Martin et al. 2022). Similarly, at the word level, a maximum word length control token could force the encoder to replace long, complex words with shorter ones (Sheang and Saggion 2021).

Learning-based approaches, on the other hand, model the system on a specified attribute by modifying the training process (Martin et al. 2022, Agrawal and Carpuat 2024). With the learning-based approach, control tokens are inserted into the input data, allowing the encoder to learn a hidden representation of the target, in turn causing the decoder to adapt its output accordingly

(Agrawal and Carpuat 2024). Attributes such as output length can be controlled by incorporating a length vector into the input. This vector guides the model during the encoding stage to learn the desired output length, similar to how a decoding-based approach would operate by using a vector during the decoding process to control the output (Mallinson et al. 2018).

2.3 Text Simplification for Dutch

Most TS research in the past has been geared towards English (Al-Thanyyan and Azmi 2021, Seidl and Vandeghinste 2024). Because of this, much of the current research does not focus on introducing new techniques for TS, but instead focuses on applying existing techniques to other languages. Here, the primary challenge often lies in finding suitable resources (Al-Thanyyan and Azmi 2021).

Early Dutch TS focused on simplification of Dutch subtitles through sentence length reduction. To accomplish this, Daelemans et al. (2004) proposed two methods: a data-driven method which leveraged a parallel corpus to learn sentence reduction and a rule-based method which relied on hand-crafted deletion rules. Similarly, work by Vandeghinste and Pan (2004) employed sentence analysis tools to generate several compressed versions of a sentence. Subsequently, the most probable and most grammatical output was selected by leveraging a parallel subtitle corpus and employing a set of rules extracted using bootstrapping.

More recent work addresses both syntactic and lexical simplification in Dutch. Sevens et al. (2018) developed a rule-based syntactic simplification model for sentence simplification, leveraging the speed and reliability of modern-day parsers to employ syntactic parsing. Bulté et al. (2018), in turn, employed a pipeline approach to develop an automated lexical simplification model. They extracted complex tokens based on age of acquisition and corpus frequency features and substituted these by simpler synonyms. Subsequently, they employed a trigram probability model to check the appropriateness of the output.

As can be noted from the prior work on Dutch TS, the absence of a large parallel corpus of simplified Dutch has led to research trying to develop and employ methods to achieve TS without relying on parallel corpora (Daelemans et al. 2004, Sevens et al. 2018, Bulté et al. 2018, Seidl and Vandeghinste 2024). In line with this, the most recent work on Dutch TS by Seidl and Vandeghinste (2024) employed neural machine translation to translate English parallel datasets to Dutch, creating a synthetic corpus which allowed the use of data-driven methods for controlled TS in Dutch. Results demonstrated that substantial simplification is possible with a synthetic dataset containing only 2,000 parallel rows, although optimal performance necessitated a minimum of 10,000 rows.

2.4 Automatic Readability Assessment

Automatic Readability Assessment (ARA) refers to the task of estimating the reading and comprehension difficulty of a text for a specified audience (Sato et al. 2008, Vajjala 2022). Typically, it is treated as a supervised machine learning problem that requires a gold standard training corpus, annotated with labels indicating either reading level categories or numbers indicating a graded scale. ARA training data can be derived from a variety of sources, both expert and non-expert annotated. (Vajjala 2022). A common method for compiling such annotated datasets is to collect texts from educational textbooks (Heilman et al. n.d., Sato et al. 2008, François 2014, Pilán et al. 2016), the underlying idea being that there is a directly proportional relation between a level of education and the complexity of the reading materials used at that level (Vajjala 2022).

The feature extraction process in ARA is crucial as it directly impacts the performance of the machine learning models. Common features include lexical (e.g., word length, word frequency), syntactic (e.g., sentence length, parse tree depth) and semantic features (e.g., word embeddings, topic modeling) (Vajjala and Meurers 2012, Collins-Thompson 2014). Advanced methods may also incorporate discourse features to capture a text's cohesion and coherence (Graesser et al. 2004).

In recent years, the development of neural network-based models has significantly advanced the field of ARA. These models, including recurrent neural networks and transformers, have demonstrated superior performance compared to traditional feature-based approaches by learning complex representations of text directly from the data (Vaswani et al. 2017, Devlin et al. 2019). For example, Vajjala and Meurers (2014) employed a multi-task learning approach using neural networks to improve readability classification by leveraging additional linguistic tasks.

With regard to Dutch, some feature-based ARA models have been developed. De Clercq and Hoste (2016) built a readability prediction system for English and Dutch using both expert annotation and annotations extracted through crowdsourcing. Similarly, Dascalu et al. (2017) introduced a new Automatic Essay Scoring method for Dutch by modifying the ReaderBench framework, which was originally developed for English, to work for Dutch texts.

Fine-tuning pre-trained language models such as BERT has shown promising results in ARA tasks across various languages. By adapting these models to specific readability assessment tasks through transfer learning, researchers can leverage large-scale pre-trained representations, significantly improving the accuracy of readability predictions (Devlin et al. 2019, Madrazo Azpiazu and Pera 2020).

2.5 N-shot Learning

N-shot learning is a machine learning technique that leverages prior knowledge to generalise to new tasks (Wang et al. 2020). This is achieved by providing the model with a natural language description, along with no examples, one example or a few. Depending on the number of examples, we differentiate between zero-shot, one-shot and few-shot learning (Brown et al. 2020).

Requiring only a few supervised data samples or even none at all, the method was proposed to tackle low performance with small datasets (Wang et al. 2020). While the concept of N-shot learning has been around for a while (Fink 2004, Fei-Fei et al. 2006), interest in N-shot learning has surged with the development of large language models (LLM), as it can leverage the extensive pre-training of these models, enabling them to effectively generalise from limited examples, showing significant performance on a variety of tasks (Wang et al. 2020, Brown et al. 2020).

In recent years, zero-shot and few-shot learning approaches have also been utilised in the field of text simplification (TS). For instance, Scarton and Specia (2018) demonstrated that zero-shot TS models managed to outperform state-of-the-art TS approaches. In terms of few-shot learning, meta-learning has also been used to perform TS on low-resource domains (Garbacea and Mei 2022). This meta-learning approach allows models to quickly generalize across new tasks using knowledge gained from previous tasks (Yin 2020, Brown et al. 2020). Similarly, transfer-learning has been employed to perform TS on low-resource languages (e.g. French, Japanese, Russian, Basque, Danish) (Ryan et al. 2023) and domains (e.g. medical records, specialised research, historical documents) (Garbacea and Mei 2022). This methodology aims to enhance the performance of models in a target domain by leveraging the knowledge derived from distinct yet related source domains.

Based on the success of these studies, we hypothesised that N-shot learning may also be functional for TS, as it allows the model to adapt to new simplification tasks with minimal data. However, we know that controlled TS requires a more specific type of simplification, which could introduce complications as it necessitates adhering to predefined rules or guidelines which may not always be effectively captured. Therefore, while N-shot learning presents a powerful tool for addressing low-resource challenges, we made sure to carefully select the examples and descriptions provided in order to maximise the models' potential for successful TS using N-shot learning techniques.

3. Data

This study aims to evaluate the effectiveness of few-shot, one-shot and zero-shot learning techniques in creating simplified Dutch texts for pupils with low reading proficiency in Dutch. To this purpose,

a collection of Dutch reading materials was compiled and used as input for the generative LLMs with the task of controlled text simplification using N-shot learning techniques. Additionally, a set of ARA models were trained to assess the quality of these simplifications. For the training and fine-tuning of these ARA models, a second corpus was compiled.

To serve as input to the generative LLM for simplification, a corpus was compiled consisting of textbook and teacher-selected reading materials aimed at Dutch L1 speakers, and more specifically pupils in the Flemish educational system. To ensure a representative distribution in terms of the types of texts in the corpus, a list of recommended text types was compiled from Flemish curriculum goals[1]. Based on this list, a sample of 50 texts was taken, covering 20 unique text types. An overview of these text types as well as the number of texts present in the corpus can be found in Table 1.

Table 1: Overview of text types and number of texts in the LLM corpus

Category	Text Types	# Texts
Narrative Fiction	Drama, Epic, Short story, Novel, Science fiction, Fairy tale	14
Opinion Pieces	Argumentative, Column, Opinion, Review, Speech	8
Professional Writing	Blog post, Resume, Website, Commercials	4
Journalistic Writing	Article, Interview, Magazine	19
Informative Writing	Informative, Essay	5

Thirty-three out of the fifty texts were extracted from five different textbooks used in the fourth, fifth and sixth form of Flemish general education. The remaining seventeen texts were collected from KlasCement[2], a teacher co-creation platform for teaching materials. These texts were selected based on their relevance to fourth, fifth and sixth form general education, as well as their peer-review score. Text length throughout the dataset ranges from 200 to 2,002 words – with on average 842.20 words per text – and amounts to 42,110 words in total.

For training ARA models, a total of 1,000 texts were chosen from two different corpora and equally distributed across four readability levels, i.e. 250 texts per level. These texts were sampled from two existing corpora: an in-house corpus that has been specifically compiled for assessing readability in the framework of the recently launched Flemish centralised tests [3] and the Wablieft corpus (Vandeghinste et al. 2019).

The first 750 texts were selected from the in-house corpus, consisting of a collection of texts extracted from various Dutch textbooks at different levels of Flemish education. For this study, texts were selected across three educational levels: fourth form primary education, sixth form primary education and first-grade secondary education A-stroom[4](Vlaamse Overheid 2024). Training the model on texts from various educational levels will not only optimise the ARA model for evaluating educational texts, but ensures a progressive increase in readability levels (Vajjala 2022). Moreover, it provides a meaningful frame of reference for envisaged end-users.

The final 250 texts were selected from the Wablieft corpus, a collection of texts from the Belgian easy-to-read Wablieft newspaper[5]. The newspaper is written in Dutch and targeted towards people with a limited functional literacy (Vandeghinste et al. 2019). As opposed to the in-house corpus, texts found in the Wablieft corpus are not specifically tailored to pupils, thus providing a readability level which may be more appropriate when working with older adolescents or even adults.

1. For each curriculum objective (leerplandoelstelling), the curriculum (leerplan) provides an overview of materials which could be employed during teaching. For the reading targets, this overview consists of a series of relevant text types.
2. https://www.klascement.net/
3. https://steunpunttoetsen.be/
4. The educational path that students follow after obtaining their primary education certificate.
5. https://www.wablieft.be/nl/krant

Only texts with a word count within a range of 100 to 1,230 were selected, allowing us to obtain enough texts per readability level, while excluding texts which are excessively short or long. Once all viable texts had been extracted from both source corpora 250 texts were randomly sampled per level. The word count statistics per level in this ARA corpus are presented in Table 2.

Table 2: Data statistics of the ARA Dataset

		Level 1	Level 2	Level 3	Level 4
Word count	Avg	372.03	411.23	346.66	226.74
	St.Dev.	253.93	261.32	266.15	157.73
	Min.	100	106	100	100
	Max.	1040	1229	1192	1121

4. Methodology

The study aims to carry out controlled TS by employing N-shot prompting techniques using two open-source generative LLMs, a multilingual and a monolingual Dutch one (Section 4.1). In order to evaluate the performance without the availability of parallel data, the simplifications are assessed in two manners. First, an automatic assessment was performed relying on automated readability assessment (Section 4.2). To this end, we first trained and compared two ARA approaches: a more traditional feature-based approach and a neural fine-tuning approach. Besides this, a human evaluation study was conducted (Section 4.3).

4.1 N-shot Learning Using Generative LLMs

For the controlled TS without the use of a parallel corpus, we employed LLAMA[6] and GEITje-ultra[7], two open-source generative LLMs, to generate simplifications at different educational levels.

For the multilingual LLAMA model, we employed the LLAMA 3 8B Instruct model, a state-of-the-art auto-regressive language model developed by META. The model uses an optimised transformer architecture and comes in two versions: a version with 8 billion parameters and a version with 70 billion parameters (META, 2024). To optimise the comparability of the LLAMA model's performance to the performance of the 7 billion parameter GEITje-ultra model, we utilised the 8 billion parameter version of LLAMA 3 over the 70 billion version. In terms of the configuration, we set both the temperature and the top_p at 0.3. These values were selected because they prioritise accuracy, which was considered crucial given that the objective was to generate four simplifications which were only marginally different from each other. At the same time, these settings provided enough randomness to prevent constant repetitions.

The second model, the GEITje 7B ultra model, is a conversational AI model designed for Dutch. The model is fine-tuned on a synthetic dataset with 56M tokens and aligned through AI feedback (Vanroy 2024). Considering the number of test instances which had to be run and the computational power required to run the model, we loaded the model in 8bit mode to optimise computational efficiency and ensure that the tests could be completed within the expected timeframe. This model was run using the default parameters.

To guide the TS process towards a specified educational level, we employed N-shot learning techniques, testing a zero-shot, one-shot and few-shot methodology. Depending on the N-shot method, the model was provided with a particular system role for each level.

6. https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct
7. https://huggingface.co/BramVanroy/GEITje-7B-ultra

The first of these methods, the zero-shot method, involved giving the model the task of simplifying a text to the desired level without providing any examples to guide it. For the first three levels, this system role consisted of the instruction: "Simplify the following text to a reading level appropriate for an x-year-old, while keeping the content appropriate for an older audience," where x would either be ten, twelve or fourteen, correlating to the average age of pupils in fourth year primary, sixth year primary and first-grade of secondary education, respectively. For the fourth level, the model was simply given the system role "Simplify the following text to a reading level that would be appropriate for a text from the Wablieft corpus." This method assumes that the LLMs had seen at least part of the Wablieft corpus, or the Wablieft newspaper as part of their training data. However, we cannot know for sure whether this was the case.

For the one-shot method, we provided the model with one example of a text at the desired level whereas in the few-shot method, we provided the model with three examples. As examples, we employed two different strategies. The first strategy involved simply providing the model with the example texts which were selected from the ARA corpus. The second strategy involved generating more complex versions of these example texts using GPT-4, allowing us to create synthetic examples. In these synthetic examples, the generated complex version was given to the LLM as the so-called original text, while the actual original text was provided as the simplified version.

For these methods, the following system role was provided: "You are a text simplification model for Dutch with level x. (An example — Some examples) of a level x text: $< example(s) >$." It should be noted that for the few-shot method, we employed text fragments, as opposed to the complete texts, in order to mitigate the length of the system role. For each method, a user role was also provided, consisting of the instruction: "Simplify the following text to level x: $< input\_text >$." Please note that all roles outlined above are translated versions of the instructions, with the instructions always being provided to the models in Dutch. A verbatim example of a regular few-shot prompt from the LLAMA model can be found in Appendix B. Additionally, a verbatim example of a synthetic few-shot prompt from the GEITje model can be found in Appendix C.

Example of a Synthetic One-Shot System Role

You are a text-simplification model for Dutch with level 0. Here is an example of a simplification to a level 0 text:

Original text:

<generated complex version >

Simplified version to level 0:

<original example from the corpus >

4.2 Automatic Evaluation

For the readability assessment, we compared the performance of a more traditional feature-based approach to a neural fine-tuning approach. Both approaches were trained on the ARA corpus which was split into a train and test set with a 90/10 split. The readability levels to be assigned were Level 1: fourth form primary education, Level 2: sixth form primary education, Level 3: first-grade secondary education (A-stroom) and Level 4: Wablieft corpus.

We first trained a **feature-based** ARA model using supervised machine learning techniques. As input, the ARA model used a set of textual features extracted from the texts using T-scan [8]. These features capture various linguistic properties such as sentence length, word length and syntactic complexity (Pander Maat et al. 2014).

Given that T-scan returned a total of 458 features per text, we performed feature selection using the SelectKBest method from scikit-learn to automatically select the top K relevant features with

8. https://github.com/CentreForDigitalHumanities/tscan

respect to the target readability levels. Selection was done using the Chi-Square test of independence, a statistical test which evaluates patterns of observations to determine if categories occur more frequently than would be expected by chance (Starbuck 2023). Given their low computational requirements, we opted to explore several machine learning algorithms for model training: Logistic Regression, Decision Trees, Random Forests, Support Vector Machines (SVM), Gradient Boosting, K-Nearest Neighbours (KNN) and Multinomial Naive Bayes.

Performance of each model was evaluated through k-fold cross-validation using F1-score as the evaluation metric. To find the optimal number of features, we performed a grid search over a range of K features[9]. For each K, we trained all models using the top K features and recorded their five-fold cross-validated F1-scores. From these results, we selected the value of K that yielded the highest F1-score for each model. This process resulted in a different optimal K for each model. Finally, we evaluated the performance of each model on a held-out test set, using the respective optimal K features determined during the validation phase.
Next, we **fine-tuned** the Dutch RobBERT 2023 (Delobelle et al. 2020)[10] language model on the classification task at hand. We made sure the same training and test splits were used. The model was fine-tuned on the raw text data but we made sure that all texts with a token count exceeding 512 tokens were truncated. Truncation occurred on the first sentence boundary which would lower the token count below 512. This ensured that no texts were excessively long, while retaining proper sentence boundaries. In total, 25% of the training data and 22% of the test data was truncated.

We employed the RobBERT 2023 Large tokenizer from the Hugging Face Transformers library to tokenize the data and defined a custom PyTorch dataset class to encapsulate the tokenized data and labels. For the classification task, the RobBERT 2023 Large model for classification was used as a base-model and fine-tuned on the training data. The model was chosen both because of its state-of-the-art results on various task, as well as its increased performance with smaller datasets compared to other state-of-the-art Dutch models (Delobelle et al. 2020).

The configuration of the training process was done using the TrainingArguments class from the Transformers library. This configuration was set to a training and evaluation batch size of eight, a weight decay of 0.01, five epochs for training and 250 steps for warmup. The Trainer class was utilised for both training and evaluation. Once the training was complete, the performance of the trained model was tested on the held-out test set.

The results on the held-out test set are presented in Table 3. Overall, we observe that the fine-tuned model outperforms all other models in terms of accuracy (Acc), precision (P), recall (R) and F1-score (F1), indicating its superior ability to classify readability levels. While simpler models like logistic regression and decision trees have moderate to low success, the SVM and Random Forest models also show strong performance. This indicates that feature-based approaches still hold value, especially in contexts where computational resources may limit the deployment of more complex neural models, where sustainability is a concern due to the substantial energy consumption of neural networks (Strubell et al. 2020) or where interpretability is a key factor. In addition to the evaluation metrics, confusion matrices for each of the models can be found in Appendix A.

These results are not without limitations. The best F1-score of 78.67% still suggests notable room for improvement.

Nevertheless, this performance can effectively be employed in educational tools to tailor reading materials to students' comprehension levels. This could aid educators with the process of selecting appropriate reading materials in terms of their readability level. In our experiments we used this best model to carry out the automatic evaluation of the simplifications.

To assess the effectiveness of each N-shot method, we first evaluated the methods' performance on a corpus sample. This decision was primarily driven by considerations of time efficiency and

9. $k \in \{10, 20, 30, \ldots, 370\}$
10. https://huggingface.co/DTAI-KULeuven/robbert-2023-dutch-large

Table 3: Automatic Readability Assessment Models' Performance

	Acc	P	R	F1	# Features
Logistic Regression	.5600	.5329	.5600	.5414	300
Decision Tree	.4700	.4574	.4700	.4617	250
Random Forest	**.6100**	**.6039**	**.6100**	**.6021**	**250**
SVM	.5800	.5578	.5800	.5639	130
Gradient Boosting	.5600	.5546	.5600	.5563	300
KNN	.5300	.5310	.5300	.5170	50
Multinomial Naive Bayes	.5200	.5115	.5200	.5104	110
Neural Model	**.7900**	**.7916**	**.7900**	**.7867**	N/A

computational resources. To this purpose we randomly selected a diverse but representative subset of ten texts from our 50-text corpus, each representing a distinct text type. The ten selected texts were subsequently used to generate simplifications using the different N-shot methods, one time with LLAMA and another time with GEITje-ultra.

Once the generation was completed, all outputs were tokenized using the Dutch RobBERT 2023 tokenizer (Delobelle et al. 2020)[11] and evaluated using the best performing ARA model. These ARA predicted readability levels were then compared to the target levels, i.e. the level to which the large language model (LLM) was asked to simplify. Based on the correlation between the predicted and desired levels, several metrics were calculated for each method allowing us to identify the best-performing N-shot method. This method was then applied to generate controlled simplifications for the entire corpus.

4.3 Manual Evaluation

4.3.1 Identification of Common Errors in LLM output

After all simplified texts were generated using the best performing N-shot learning approach and LLM model, each of the 200 generated simplifications were manually evaluated by a trained linguist and native speaker of Dutch, focussing on six errors which are known to commonly occur when employing LLMs.

As a first error, we checked whether the model provided four different texts, one for each of the target levels, or whether it had simply generated the same text two or more times. Evaluating the diversity of the outputs is important, as LLMs have shown to sometimes struggle to generate varied outputs, often producing outputs that are too similar to one another (Holtzman et al. 2020). Additionally, we hypothesised that asking the LLM to generate four versions of a single text, with only slightly varying levels of complexity, could further exacerbate this issue.

Next, we checked whether the generated outputs actually contained a modified version of the input text. For the task of controlled TS, it is crucial that the generated texts retain the core information of the input text, a challenge often noted in text transformation tasks (See et al. 2017). With this in mind, we checked if the generated output consisted of a summary of the input text, rather than a simplified version; or if the generated output was simply a comment on, or a question about the input text.

The outputs were also examined for completeness, ensuring each text had a clear beginning, body and end. Texts that ended abruptly were considered incomplete. This criterion helped assess the coherence and overall usability of the generated texts. Incomplete or abruptly ending texts are

11. The same tokenizer that was employed for fine-tuning the ARA model

again a common issue in LLM outputs, which can affect the overall quality and coherence (Wiseman et al. 2017).

Additionally, the texts were evaluated to determine if they were summaries or descriptions of the input text rather than rewritten versions. The objective was for the model to generate new versions of the input text, preserving the original meaning while rephrasing the content. LLMs often face challenges in balancing between maintaining the original meaning and rephrasing the content effectively, which can affect the usefulness of the generated text (Kryscinski et al. 2019).

Each text was also assessed for obvious hallucinations, instances of factually incorrect or fabricated content that had no basis in the input text. Given that LLMs are known to be prone to hallucinations, where they generate plausible sounding but inaccurate or nonsensical information (Ji et al. 2023), this assessment is crucial in evaluating the factual accuracy and reliability of the outputs.

Finally, we checked whether the outputs were simplifications of the example text provided in the system role, rather than the actual input text which needed to be simplified. Misapplication of simplification processes can lead to outputs that are not aligned with the intended input, a known limitation in simplification models (Xu et al. 2015).

4.3.2 Human Evaluation of the Outputs' Relative Complexity by Teachers

Based on the results from the manual evaluation discussed above, all texts which were deemed completely unusable were excluded. From the remaining texts, a total of 21 simplifications, attained from 7 original texts, were randomly selected for the teacher evaluation. In order to limit the time required for filling out each question, we selected fragments of the generated outputs, rather than providing the full texts. This allowed us to present the evaluators with a larger number of texts within a similar timeframe.

For the teacher evaluation, we focused exclusively on the first three levels, since these were based on actual educational materials, allowing us to confidently assume that Level 1 (fourth form primary) is simpler than Level 2 (sixth form primary), which in turn is simpler than Level 3 (first-grade secondary). The fourth readability level, which was based on texts from the Wablieft corpus, was not included, as we could not decisively assume its complexity compared to the other levels.

Two surveys[12] were constructed for this evaluation process. The first survey targeted teachers from the fourth and sixth years of Flemish primary education, as well as the first grade of Flemish secondary education. It consisted of seven parts, each containing three text fragments. Each part represented one of the seven original texts which were simplified across the three readability levels (Levels 1-3). For each part, teachers were asked two questions. Question A required them to rank the three text fragments from easiest to most difficult by means of a Likert scale with three options. If teachers considered two or more texts to have the same readability level, they could indicate this by assigning them the same score. For the outputs to be considered accurate, Level 1 fragments should be the easiest fragments, while Level 3 fragments should be the most complex.

Question B asked teachers to indicate how appropriate they deemed each text for use at their specific level of education, using a 4-point Likert scale which ranged from "Very inappropriate" to "Very appropriate". For this question, the output can be considered accurate if fourth form teachers find Level 1 texts most appropriate, sixth form teachers find Level 2 texts most appropriate and first-grade teachers find Level 3 texts most appropriate.

The second survey was aimed at teachers of the third-grade of secondary education. They were presented the same survey. For question A, they were also asked to rank the texts from easiest to most complex. Question B, however, varied slightly for this group. Given that none of the simplified texts were aimed towards a third-grade secondary education level, the third-grade teachers were not

12. All surveys can be found in the online repository: https://github.com/WoutSin/Controlled-Text-Simplification-for-Dutch-using-Generative-Large-Language-Models.git

asked to indicate how appropriate they deemed each text for use at their specific level of education. Instead, they were asked to rate how appropriate they deemed each text to be for pupils with low Dutch reading proficiency. They could again indicate this using a Likert scale ranging from "Very inappropriate" to "Very appropriate".

The purpose of this human teacher evaluation was to assess the quality of the generated output in terms of the simplification itself and to determine whether the LLMs were able to generate texts of a desired readability level. By involving educators in the evaluation process, we were able to conduct a more thorough assessment of both the readability and appropriateness of the simplified texts. Firstly, by having teachers rank the simplified texts in terms of their complexity, we could verify whether they exhibited a clear progression in complexity across the three educational levels. Secondly, having teachers evaluate the appropriateness of each text for their specific educational level allowed us to assess whether the generated outputs were simplified to the correct level. Finally, by having third-grade secondary education teachers indicate whether they considered the texts appropriate to use as alternatives for pupils with a low Dutch reading proficiency level.

The surveys were sent out to six Flemish schools. In the end, a total of ten teachers participated in the human evaluation: four teachers from the fourth form primary education, three teachers from the sixth form primary education, one teacher from the first-grade secondary education and two teachers from the third-grade secondary education.

5. Results

As discussed above, we checked the viability of text simplification for Dutch using N-shot prompting techniques with two generative open-source LLMs. The effectiveness of these models, in the absence of parallel corpora, was assessed through an automatic readability assessment model (Section 5.1). Additionally, generated outputs were manually assessed (Section 5.2), including manual error checking of the output and an additional human assessment by Flemish teachers verify the appropriateness and practical usability of the simplified texts in an educational setting.

5.1 Automatic Evaluation

The performance of the various N-shot learning methods for text simplification, using GEITje-ultra and LLAMA, were evaluated using our best ARA model. Once the model had predicted the readability level of the simplified texts, key performance metrics, including accuracy (Acc), precision (P), recall (R) and F1-score (F1) were calculated for evaluation.

Table 4 presents the comparative performance metrics for each N-shot learning approach using the neural ARA model. The results indicate that the synthetic one-shot method using GEITje-ultra achieved the highest performance across most metrics, with an accuracy of 30.0%, precision of 35.0%, recall of 30.0% and F1-score of 29.0%. Overall, the performance is relatively low, with each model attaining results close to the probability of random guess. These results suggest that N-shot learning, in its current implementation, might not be suitable for controlled TS tasks.

When considering the various N-shot learning approaches, with and without a synthetic example, the maximum difference between the GEITje-ultra and LLAMA also comes down to just two percent. Notably, GEITje-ultra, the monolingual Dutch model, benefited from synthetic examples, while LLAMA, a multilingual model, experienced performance degradation with synthetic examples. This contrasting effect highlights how different models may respond differently to various prompting techniques. In the case of this study, synthetic examples present pairs of texts: one complex (synthetically generated) and one simplified (the original example). Non-synthetic examples, on the other hand, only show what a text at a specific level looks like.

GEITje-ultra seemed to benefit from these synthetic examples, likely because the clear contrast between complex and simplified versions helped the model better interpret the simplification process. LLAMA's performance, as mentioned, was negatively impacted by synthetic examples. This

discrepancy might be due to LLAMA's training data and architecture, which may perhaps not be optimised for learning from such structured pairs of texts. For instance, the use of a simplification example may require the model to make very language-specific adjustments that it is less adept at performing compared to a monolingual model like GEITje-ultra. While there could be several reasons as to why a measure can positively impact one model, but not another, these results do highlight the importance of careful consideration and experimentation when applying different techniques to various models. The one-shot performance of GEITje-ultra, for instance, was 1.53 times better with a synthetic example, while the performance of the LLAMA model was 1.85 times worse.

Table 4: Performance of N-shot Models and Distribution of Predicted Readability Levels

Method	Performance				Distribution			
	Acc	P	R	F1	Level 1	Level 2	Level 3	Level 4
Zero-shot GEITje	0.225	0.230	0.230	0.170	0.225	0.700	0.050	0.025
Zero-shot LLAMA	0.200	0.170	0.200	0.150	0.100	0.750	0.050	0.100
One-shot GEITje	0.250	0.200	0.250	0.190	0.175	0.700	0.100	0.025
One-shot LLAMA	0.275	0.310	0.270	0.270	0.150	0.525	0.125	0.200
One-shot GEITje (synt)	**0.300**	**0.350**	**0.300**	**0.290**	0.150	0.575	0.175	0.100
One-shot LLAMA (synt)	0.200	0.200	0.200	0.170	0.075	0.550	0.175	0.200
Few-shot GEITje	0.225	0.200	0.230	0.170	0.200	0.725	0.075	0.000
Few-shot LLAMA	0.275	0.430	0.270	0.270	0.050	0.525	0.200	0.225
Few-shot GEITje (synt)	0.300	0.470	0.300	0.260	0.200	0.625	0.150	0.025
Few-shot LLAMA (synt)	0.250	0.180	0.250	0.190	0.025	0.625	0.200	0.150

The distribution of the predicted readability levels, i.e. the percentage of generated outputs that were classified as a particular level (1-4) by the ARA model, was also calculated across all different N-shot learning models. This was done by counting the number of times each readability level was predicted and dividing this by the total number of predictions. According to these numbers, all models – regardless of N-shot method – exhibited a tendency to generate simplifications of Level 2. Moreover, the results show that the GEITje-ultra model tends to generate simplifications of Level 1 and Level 2 more frequently compared to the LLAMA model. The LLAMA model, in turn, appears to have a higher tendency to generate Level 3 and Level 4 simplifications.

After being identified by the neural ARA model as the best performing N-shot method on a subset of the corpus, the GEITje-ultra model, utilising the one-shot learning approach with a synthetic example, was employed to generate simplified versions of the full corpus. The results for these generated outputs indicate an accuracy of 28.5%, a precision of 31.0%, a recall of 29.0% and an F1-score of 24.0%. These results are lower than the performance which was attained on the subset of the corpus using the same prompts and suggest that the performance of GEITje-ultra for the task of controlled TS is close to the probability of a random guess, indicating that the model is not effectively generating appropriate simplified texts.

That being said, a potential limitation on the LLM's performance is the length of the system and user role inputs. In zero-shot learning, the system role input is restricted to a limited number of characters. However, in one-shot and few-shot learning scenarios, the system role inputs are significantly longer, with average lengths ranging from 2345.50 to 6354.75 characters, as they include one or three examples respectively. Similarly, due to this study's focus on text-level simplification, the user role contained the full input text, which averaged 842.20 words, along with a description of the task. This often led to a substantial amount of text inputted to the LLM through the user-role. This large number of characters inputted to the LLMs through both system and user roles could

negatively impact its performance, as LLMs are known to show drops in reasoning performance as input grows (Levy et al. 2024).

5.2 Human Evaluation

5.2.1 Manual Identification of Common Errors in LLMs

All 200 simplifications which were generated using the synthetic one-shot learning approach for GEITje-ultra, were also manually checked for common errors[13]. Table 5 lists the frequency of these errors and reveals that 4.5% of the outputs were repeated multiple times. Additionally, 15.5% of the outputs did not contain a modified version of the input text. In 13.0% of these cases, the output was a summary of the input text. In the remaining cases, the output was either a comment or question regarding the input text. A significant 95.0% of the generated outputs were incomplete, suggesting a major issue in the model's ability to generate comprehensive responses of this length. Finally, both hallucinations and simplification of the readability level example, as opposed to the input text, occurred at a low frequency of 2.0%.

Table 5: Frequency of Common Errors in the Generated Output

	Frequency
LLM generated the same output multiple times	.045
LLM did not generate a modified version of the input text	.155
Generated output was a summary of the input text	.130
Generated output was incomplete	.950
Generated output contained obvious hallucinations	.020
Generated output was a simplified version of the example text	.020

Based on this analysis, 76 of the original 200 texts (38%) were deemed unusable due to the high number of errors in the generated output. The 124 remaining texts were derived from 31 original texts, each of which had been simplified to four different readability levels. Subsequently, 21 out of these 124 texts, derived from seven different original input texts, were used for the human evaluation of the outputs' relative complexity by teachers.

5.2.2 Human Evaluation of the Outputs' Relative Complexity by Teachers

First, all teachers were asked to rank the texts from simplest to most complex, with the option to give the same rating to two or more fragments if there was no distinguishable difference in terms of readability.

Table 6 presents the assessment results. Given that the three levels correspond to target complexity levels for the model's outputs, where Level 1 should be the simplest and Level 3 the most complex, the first three columns and rows can be interpreted as a confusion matrix. Each fragment was evaluated by ten teachers, resulting in 70 evaluations per level and 210 in total.

The results show that Level 1 fragments were identified as the simplest in 40.0% of the cases. Level 2 fragments were considered intermediate in 20.0% of cases, being perceived as the most complex fragments 35.7% of the time. Finally, Level 3 fragments were correctly identified as the most complex 30.0% of the time. This shows that while Levels 1 and 3 were most commonly classified as the simplest and most complex respectively, i.e. their target complexity levels, Level 2 fragments were often perceived as most complex compared to their other classifications.

13. A complete overview of the manual error checking can be found in the repository: https://github.com/WoutSin/Controlled-Text-Simplification-for-Dutch-using-Generative-Large-Language-Models/tree/main/5.%20Manual%20Error%20Checking%20Data

Table 6: Human Evaluation of the Simplified Outputs' Relative Complexity

	Simplest	In between	Most complex	No difference
Level 1 Fragment	28	21	11	10
Level 2 Fragment	15	14	25	16
Level 3 Fragment	15	16	21	18

Next, the fourth form, sixth form and first-grade teachers were asked to rate the appropriateness of each generated output as reading materials at their respective levels of education. The third-grade teachers, on the other hand, were asked to rate the materials in terms of how appropriate they were for third-grade pupils with a lower Dutch reading level. Table 7 presents a contingency table of appropriateness ratings, showing the proportion of times teachers at the various educational levels assigned each rating to generated simplifications of a particular target readability level.

These results indicate that the Level 1 target output was always deemed the most appropriate, regardless of the educational level. For the interpretation of these results with regard to what they can tell us about the performance of the large language model (LLM) for text simplification (TS), it is important to keep in mind that a Level 1 fragment is meant to correlate to a fourth form primary education level, a Level 2 fragment is meant to correlate to a sixth form primary education level and a Level 3 fragment is meant to correlate to a first-grade secondary education level. In other words, the first three rows of the table can be interpreted as a 3x3 confusion matrix.

With this in mind, the Level 1 fragments, aimed at fourth form primary education, were, on average, considered the most appropriate fragments by fourth form teachers. The Level 2 fragments, correlating to a sixth form primary education level, were, on average, considered the second most appropriate by sixth form teachers, with Level 1 fragments being considered the most appropriate. The Level 3 fragments, correlating to a first-grade secondary education level, were considered the third most appropriate by first-grade teachers, followed by the Level 1 and Level 2 fragments, respectively. Finally, third-grade teachers deemed the Level 1 fragments most appropriate on average, while Level 2 and Level 3 fragments attained the same average score.

Table 7: Contingency Table of Appropriateness Ratings

Teacher Education Level	Target Readability Level	Appropriateness Distribution			
		Very Inappropriate	Inappropriate	Appropriate	Very Appropriate
Fourth Form	Level 1	**0.000**	**0.143**	**0.500**	**0.357**
	Level 2	0.071	0.357	0.393	0.179
	Level 3	0.036	0.357	0.393	0.214
Sixth Form	Level 1	0.000	0.095	0.476	0.429
	Level 2	**0.000**	**0.143**	**0.667**	**0.190**
	Level 3	0.000	0.381	0.476	0.143
First Grade	Level 1	0.000	0.286	0.714	0.000
	Level 2	0.000	0.429	0.571	0.000
	Level 3	**0.143**	**0.571**	**0.286**	**0.000**
Third Grade	Level 1	0.071	0.214	0.500	0.214
	Level 2	0.071	0.500	0.286	0.143
	Level 3	0.071	0.500	0.286	0.143

6. Discussion and Conclusion

This study set out to investigate the capabilities of Large Language Models (LLM) for controlled text simplification (TS) in Dutch using N-shot learning. Through a combination of automated assessments and human evaluations, the study aimed to determine the effectiveness and reliability of LLMs in performing this specific task. The evaluation methods included a neural Automatic Readability Assessment (ARA) model, manual error checking and human evaluation by teachers, providing a comprehensive analysis of the models' performance.

In order to achieve our goal, a series of feature-based models, as well as a fine-tuned ARA model, were first trained and tested to evaluate their performance in classifying readability levels of Dutch texts. The results indicated that the fine-tuned ARA model outperformed traditional machine learning models, achieving a higher accuracy (79.00%), precision (79.16%), recall (79.00%) and F1-score (78.67%). This model was selected for the automatic assessment of the generated simplifications' readability level.

Subsequently, various N-shot learning techniques for controlled TS using GEITje-ultra and LLAMA were evaluated on a subset of our corpus using the best ARA model. This evaluation revealed varying impacts on performance depending on the employed method. The synthetic one-shot method using GEITje-ultra achieved the highest performance metrics among the tested approaches. However, these results were still relatively poor, indicating substantial room for improvement. Notably, GEITje-ultra benefitted from synthetic examples, while LLAMA experienced performance degradation with these examples, highlighting how different models may respond differently to various prompting techniques. Overall, the performance of all models was low, suggesting that N-shot learning, in its current implementation, is not suitable for controlled TS tasks.

Finally, the best-performing N-shot learning method, as identified by the evaluation using the fine-tuned ARA model, was employed to generate simplifications on the full corpus. These simplifications were then assessed through the ARA model, manual error checking and human evaluation by teachers. The findings of these evaluations also indicate significant limitations in the current methodologies for using LLMs for TS. The ARA model's evaluation revealed low performance across all metrics, suggesting that the GEITje-ultra model's performance is close to random guess probability. These results collectively suggest that LLMs, under current methodologies, do not yield effective results for TS through N-shot learning. The research question, "Can large language models be leveraged to perform text simplification aimed towards pupils through N-shot learning?" is therefore answered negatively based on the presented evidence.

6.1 Discussion

That being said, it is crucial to contextualise these findings within the intended use of LLMs. Given that these are designed to be used conversationally, with iterative back-and-forth interactions, the results do not necessarily imply that LLMs are incapable of TS; rather, they highlight that achieving this goal through a single prompt in N-shot learning is unlikely. Successful application for TS would likely require more sophisticated prompt engineering and iterative refinement processes. So, while the findings of this study indicate that the current approach of using LLMs for TS through N-shot learning shows significant limitations, this does not preclude the potential for these models to be effective in TS with appropriate methodologies and prompt engineering strategies.

Previous research in TS has shown that modern approaches, particularly those utilising deep learning models like neural machine translation, have significantly advanced the field. These models, trained on large corpora, are adept at maintaining semantic integrity while simplifying texts (Wang et al. 2016, Zhang and Lapata 2017). However, the limitations of generative LLMs in TS, as evidenced by the GEITje-ultra model's performance, suggest that these models struggle with the specific requirements of controlled simplification when using N-shot learning. This contrasts with findings that emphasise the success of neural networks in TS tasks, highlighting a gap in the application of

these models for controlled TS without iterative refinement (Sheang and Saggion 2021, Agrawal and Carpuat 2024).

The study's findings also align with the broader challenges of applying TS to languages other than English. Dutch TS research has often focused on methods that do not rely on large parallel corpora due to their scarcity (Daelemans et al. 2004, Sevens et al. 2018, Bulté et al. 2018). This aligns with the difficulties observed in generating appropriate Dutch simplifications with the GEITje-ultra model, suggesting that the lack of robust Dutch parallel corpora may contribute to the model's poor performance. Moreover, the synthetic corpus approach used by Seidl and Vandeghinste (2024) underscores the necessity of creative methodologies to overcome data limitations, a factor that may need more emphasis in future research using LLMs for Dutch TS.

Additionally, the study's insights into the poor human evaluation scores reflect ongoing concerns in TS literature regarding the alignment of simplified outputs with specific audience needs (Martin et al. 2022, Kew and Ebling 2022). The variability in teachers' assessments of readability levels further highlights that LLMs struggle to produce outputs that consistently match the intended complexity and appropriateness for the provided educational contexts, echoing the need for refined control mechanisms in TS models (Agrawal and Carpuat 2024). While N-shot learning provides examples of the task to the model, results from this study suggest that it may lack the nuanced control over output attributes that more targeted methodologies, such as the use of control tokens, can offer.

6.2 Limitations and Future Work

While this study provides insights into the challenges of leveraging generative LLMs for TS through N-shot learning, several limitations must be acknowledged to contextualise the findings and guide future research. This research focused specifically on Dutch language simplification. Given that Dutch presents unique challenges due to the scarcity of parallel simplification corpora, the findings may not be entirely generalizable to other languages, especially those with more extensive resources and data available, as the linguistic characteristics and availability of training data can significantly affect the performance of LLMs (Brown et al. 2020).

Furthermore, the evaluation relied on a combination of automatic readability assessment (ARA), manual error checking and human evaluations. While comprehensive, these methods have inherent limitations. Despite the ARA model's performance metrics being relatively good, false classifications can still impact the reliability of the automated assessments. Moreover, human evaluations, though critical, can be subjective and vary significantly based on the individual rater's perspective and criteria. Additionally, only a total of 10 teachers participated in the human evaluation, making this a relatively small-scale human evaluation. This limited sample size may not fully capture the diversity of educational contexts and user needs, potentially affecting the generalizability of the findings. Future research could seek to include a larger and even more diverse group of evaluators to provide a more robust understanding of LLMs' effectiveness across different educational settings.

Based on the findings of this study, future research could focus on exploring additional alternative methodologies for TS beyond N-shot learning with LLMs. One potential area for further investigation is the application of iterative prompt engineering and back-and-forth interactions with LLMs to refine and improve the quality of simplified texts. Studies could experiment with various prompt engineering strategies, incorporating user feedback loops to enhance the accuracy and readability of the outputs. Additionally, examining the use of control tokens and other targeted techniques could offer more precise control over the output attributes, potentially addressing the nuanced requirements of controlled TS that N-shot learning currently appears to struggle with.

Another area for further research is the development and evaluation of synthetic parallel corpora tailored to languages with limited existing resources, such as Dutch. Given the challenges highlighted by the study in generating appropriate Dutch simplifications, creating high-quality synthetic data could enhance the fine-tuning and performance of LLMs in these contexts. Moreover, comparing

the performance of LLMs on controlled TS tasks across different languages with varying levels of resource availability could provide valuable insights into the generalizability of these models and inform the design of more robust, perhaps even language-agnostic simplification systems.

Finally, the large number of characters inputted to the LLMs through both system and user roles may have negatively impacted its performance. Because of this, future research could focus on employing N-shot learning techniques using LLMs for sentence-level simplification, as opposed to text-level simplification. The use of sentences, as opposed to full texts, would significantly reduce the length of the system and user roles inputted to the LLM, which could positively impact its performance (Levy et al. 2024). Additionally, this methodology can still be used to simplify entire texts by first performing sentence-level tokenization of the input text.

In conclusion, this study provided insights into the challenges and limitations of using LLMs for controlled TS through N-shot learning, emphasising the need for more refined approaches to harness the full potential of these models for TS tasks, while also highlighting the difficulties of working with under-resourced languages like Dutch.

Acknowledgements

We would like to thank all teachers who contributed voluntarily to the human evaluation study and the Steunpunt Centrale Toetsen Onderwijs for granting us access to an in-house dataset. Additionally, we would like to thank the reviewers for their valuable insights. This work was supported by Ghent University under grant BOF.STG.2022.0012.01.

References

Agrawal, S. and M. Carpuat (2024), Do Text Simplification Systems Preserve Meaning? A Human Evaluation via Reading Comprehension, *Transactions of the Association for Computational Linguistics* **12**, pp. 432–448, MIT Press.

Al-Thanyyan, Suha S and Aqil M Azmi (2021), Automated Text Simplification: A Survey, *ACM Computing Surveys (CSUR)* **54** (2), pp. 1–36, ACM New York, NY, USA.

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris 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, *Advances in Neural Information Processing Systems*, Vol. 33, pp. 1877–1901.

Bulté, Bram, Leen Sevens, and Vincent Vandeghinste (2018), Automating Lexical Simplification in Dutch, *Computational Linguistics in the Netherlands Journal* **8**, pp. 24–48.

Carroll, John, Guido Minnen, Yvonne Canning, Siobhan Devlin, and John Tait (1998), Practical Simplification of English Newspaper Text to Assist Aphasic Readers, *Proceedings of the AAAI-98 workshop on integrating artificial intelligence and assistive technology*, Madison, WI, pp. 7–10.

Collins-Thompson, Kevyn (2014), Computational Assessment of Text Readability: A survey of current and future research, *ITL-International Journal of Applied Linguistics* **165** (2), pp. 97–135, John Benjamins.

Daelemans, Walter, Anja Höthker, and Erik F Tjong Kim Sang (2004), Automatic Sentence Simplification for Subtitling in Dutch and English, *Proceedings of the Fourth International Conference*

on Language Resources and Evaluation (LREC'04), European Language Resources Association (ELRA).

Dascalu, Mihai, Wim Westera, Stefan Ruseti, Stefan Trausan-Matu, and Hub Kurvers (2017), ReaderBench Learns Dutch: Building a Comprehensive Automated Essay Scoring System for Dutch Language, *Artificial Intelligence in Education*, Springer International Publishing, pp. 52–63.

De Clercq, Orphée and Véronique Hoste (2016), All mixed up? Finding the optimal feature set for general readability prediction and its application to English and Dutch, *Computational Linguistics* **42** (3), pp. 457–490, MIT Press One Rogers Street, Cambridge, MA 02142-1209.

Delobelle, Pieter, Thomas Winters, and Bettina Berendt (2020), RobBERT: a Dutch RoBERTa-based Language Model, *Findings of the Association for Computational Linguistics: EMNLP 2020*, Association for Computational Linguistics, pp. 3255–3265.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019), BERT: Pre-training of deep bidirectional transformers for language understanding, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Association for Computational Linguistics, pp. 4171–4186.

Evans, Richard, Constantin Orasan, and Iustin Dornescu (2014), An evaluation of syntactic simplification rules for people with autism, *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, Association for Computational Linguistics, pp. 131–140.

Fei-Fei, Li, Robert Fergus, and Pietro Perona (2006), One-shot learning of object categories, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28** (4), pp. 594–611, IEEE.

Fink, Michael (2004), Object classification from a single example utilizing class relevance metrics, *Advances in Neural Information Processing Systems*, Vol. 17.

François, Thomas (2014), An analysis of a French as a Foreign Language Corpus for Readability Assessment, *Proceedings of the Third Workshop on NLP for computer-assisted language learning*, pp. 13–32.

Garbacea, Cristina and Qiaozhu Mei (2022), Adapting Pre-trained Language Models to Low-Resource Text Simplification: The Path Matters, *Conference on Lifelong Learning Agents*, PMLR, pp. 1103–1119.

Graesser, Arthur C, Danielle S McNamara, Max M Louwerse, and Zhiqiang Cai (2004), Coh-Metrix: Analysis of text on cohesion and language, *Behavior research methods, instruments, & computers* **36** (2), pp. 193–202, Springer.

Heilman, Michael, Kevyn Collins-Thompson, Jamie Callan, and Maxine Eskenazi (n.d.), Combining Lexical and Grammatical Features to Improve Readability Measures for First and Second Language Texts, *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, Association for Computational Linguistics, pp. 460–467.

Holtzman, Ari, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi (2020), The Curious Case of Neural Text Degeneration, *Proceedings of International Conference on Learning Representations.*

Ji, Ziwei, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung (2023), Survey of Hallucination in Natural Language Generation, *ACM Computing Surveys* **55** (12), pp. 1–38, ACM New York, NY.

Kew, Tannon and Sarah Ebling (2022), Target-Level Sentence Simplification as Controlled Paraphrasing, *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, Association for Computational Linguistics, pp. 28–42.

Kikuchi, Yuta, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura (2016), Controlling Output Length in Neural Encoder-Decoders, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, pp. 1328–1338.

Kryscinski, Wojciech, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher (2019), Neural Text Summarization: A Critical Evaluation, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Association for Computational Linguistics, Hong Kong, China, pp. 540–551. https://aclanthology.org/D19-1051.

Kulmizev, Artur and Joakim Nivre (2022), Schrödinger's Tree—On Syntax and Neural Language Models, *Frontiers in Artificial Intelligence* **5**, pp. 796788, Frontiers Media SA.

Levy, Mosh, Alon Jacoby, and Yoav Goldberg (2024), Same Task, More Tokens: the Impact of Input Length on the Reasoning Performance of Large Language Models, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, pp. 15339–15353.

Madrazo Azpiazu, Ion and Maria Soledad Pera (2020), An Analysis of Transfer Learning Methods for Multilingual Readability Assessment, *Adjunct Publication of the 28th ACM Conference on User Modeling, Adaptation and Personalization*, pp. 95–100.

Mallinson, Jonathan, Rico Sennrich, and Mirella Lapata (2018), Sentence Compression for Arbitrary Languages via Multilingual Pivoting, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, pp. 2453–2464.

Martin, Louis, Angela Fan, Éric De La Clergerie, Antoine Bordes, and Benoît Sagot (2022), MUSS: Multilingual Unsupervised Sentence Simplification by Mining Paraphrases, *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, European Language Resources Association (ELRA), pp. 1651–1664.

Paetzold, Gustavo and Lucia Specia (2016), Unsupervised Lexical Simplification for Non-Native Speakers, *Proceedings of the AAAI Conference on Artificial Intelligence*.

Pander Maat, HLW, Rogier Kraf, APJ Van den Bosch, Nick Dekker, M van Gompel, S de Kleijn, Ted Sanders, and K van der Sloot (2014), T-Scan: a new tool for analyzing Dutch text, *Computational Linguistics in the Netherlands Journal* **4**, pp. 52 – 74.

Pilán, Ildikó, Sowmya Vajjala, and Elena Volodina (2016), A Readable Read: Automatic Assessment of Language Learning Materials based on Linguistic Complexity, *International Journal of Computational Linguistics and Applications* **7**, pp. 143 – 159.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. (2019), Language Models are Unsupervised Multitask Learners, *OpenAI blog* **1** (8), pp. 9.

Rello, Luz, Ricardo Baeza-Yates, Laura Dempere-Marco, and Horacio Saggion (2013), Frequent Words Improve Readability and Short Words Improve Understandability for People with Dyslexia, *Human-Computer Interaction – INTERACT 2013*, Springer Berlin Heidelberg, pp. 203–219.

Ryan, Michael J, Tarek Naous, and Wei Xu (2023), Revisiting non-English Text Simplification: A Unified Multilingual Benchmark, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, pp. 4898–4927.

Sato, Satoshi, Suguru Matsuyoshi, and Yohsuke Kondoh (2008), Automatic Assessment of Japanese Text Readability Based on a Textbook Corpus, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, European Language Resources Association (ELRA).

Scarton, Carolina, Alessio Palmero Aprosio, Sara Tonelli, Tamara Martín Wanton, and Lucia Specia (2017), MUSST: A Multilingual Syntactic Simplification Tool, *Proceedings of the IJCNLP 2017, System Demonstrations*, pp. 25–28.

Scarton, Carolina and Lucia Specia (2018), Learning Simplifications for Specific Target Audiences, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Association for Computational Linguistics, pp. 712–718.

See, Abigail, Peter J Liu, and Christopher D Manning (2017), Get to the point: Summarization with Pointer-Generator Networks, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, p. 1073–1083.

Seidl, Theresa and Vincent Vandeghinste (2024), Controllable Sentence Simplification in Dutch, *Computational Linguistics in the Netherlands Journal* **13**, pp. 31–61.

Sevens, Leen, Vincent Vandeghinste, Ineke Schuurman, and Frank Van Eynde (2018), Less is more: A rule-based syntactic simplification module for improved text-to-pictograph translation, *Data & Knowledge Engineering* **117**, pp. 264–289, Elsevier.

Shardlow, Matthew (2014), A Survey of Automated Text Simplification, *International Journal of Advanced Computer Science and Applications* **4** (1), pp. 58–70.

Sheang, Kim Cheng and Horacio Saggion (2021), Controllable Sentence Simplification with a Unified Text-to-Text Transfer Transformer, *Proceedings of the 14th International Conference on Natural Language Generation*, pp. 341–352.

Siddharthan, Advaith and Angrosh Annayappan Mandya (2014), Hybrid text simplification using synchronous dependency grammars with hand-written and automatically harvested rules, *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, Association for Computational Linguistics, pp. 722–731.

Starbuck, Craig (2023), *The Fundamentals of People Analytics: With Applications in R*, Springer Nature.

Strubell, Emma, Ananya Ganesh, and Andrew McCallum (2020), Energy and Policy Considerations for Modern Deep Learning Research, *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34, pp. 13693–13696.

Vajjala, Sowmya (2022), Trends, Limitations and Open Challenges in Automatic Readability Assessment Research, *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, European Language Resources Association (ELRA), pp. 5366–5377.

Vajjala, Sowmya and Detmar Meurers (2012), On Improving the Accuracy of Readability Classification Using Insights from Second Language Acquisition, *Proceedings of the Seventh Workshop on Building Educational Applications using NLP*, Association for Computational Linguistics, pp. 163–173.

Vajjala, Sowmya and Detmar Meurers (2014), Readability Assessment for Text Simplification: From Analysing Documents to Identifying Sentential Simplifications, *ITL-International Journal of Applied Linguistics* **165** (2), pp. 194–222, John Benjamins.

Vandeghinste, Vincent and Yi Pan (2004), Sentence Compression for Automated Subtitling: A Hybrid Approach, *Text Summarization Branches Out*, pp. 89–95.

Vandeghinste, Vincent, Bram Bulté, and Liesbeth Augustinus (2019), Wablieft: An Easy-to-Read Newspaper Corpus for Dutch, *Proceedings of CLARIN Annual Conference*, pp. 188–191.

Vanroy, Bram (2024), GEITje 7B Ultra: A Conversational Model for Dutch, *arXiv preprint arXiv:2412.04092*.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin (2017), Attention is All You Need, *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 6000–6010.

Vlaamse Overheid (2024). https://www.vlaanderen.be/naar-het-secundair-onderwijs.

Wang, Tong, Ping Chen, Kevin Amaral, and Jipeng Qiang (2016), An Experimental Study of LSTM Encoder-Decoder Model for Text Simplificatio, *arXiv preprint arXiv:1609.03663*.

Wang, Yaqing, Quanming Yao, James T Kwok, and Lionel M Ni (2020), Generalizing From a Few Examples: A Survey on Few-Shot Learning, *ACM Computing Surveys* **53** (3), pp. 1–34, ACM New York, NY, USA.

Wiseman, Sam, Stuart M Shieber, and Alexander M Rush (2017), Challenges in data-to-document generation, *arXiv preprint arXiv:1707.08052*.

Xia, Menglin, Ekaterina Kochmar, and Ted Briscoe (2016), Text Readability Assessment for Second Language Learners, *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, Association for Computational Linguistics, pp. 12–22.

Xu, Wei, Chris Callison-Burch, and Courtney Napoles (2015), Problems in Current Text Simplification Research: New Data Can Help, *Transactions of the Association for Computational Linguistics* **3**, pp. 283–297.

Yin, Wenpeng (2020), Meta-Learning for Few-Shot Natural Language Processing: A Survey, *arXiv preprint arXiv:2007.09604*.

Zhang, Xingxing and Mirella Lapata (2017), Sentence Simplification with Deep Reinforcement Learning, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, pp. 584–594.

Appendix A. Confusion Matrices from the ARA Models' Evaluation

Table 8: CFM1: Logistic Regression Confusion Matrix

	Level 1	Level 2	Level 3	Level 4
Level 1	16	1	5	3
Level 2	10	5	8	2
Level 3	3	8	13	1
Level 4	1	2	0	22

Table 9: CFM2: Decision Tree Confusion Matrix

	Level 1	Level 2	Level 3	Level 4
Level 1	8	6	9	2
Level 2	7	9	8	1
Level 3	3	8	9	5
Level 4	3	0	1	21

Table 10: CFM3: Random Forest Confusion Matrix

	Level 1	Level 2	Level 3	Level 4
Level 1	16	2	5	2
Level 2	8	11	3	3
Level 3	4	7	12	2
Level 4	2	1	0	22

Table 11: CFM4: Support Vector Machine Confusion Matrix

	Level 1	Level 2	Level 3	Level 4
Level 1	14	2	6	3
Level 2	7	8	6	4
Level 3	4	8	12	1
Level 4	0	1	0	24

Table 12: CFM5: Gradient Boosting Confusion Matrix

	Level 1	Level 2	Level 3	Level 4
Level 1	13	4	7	1
Level 2	7	11	5	2
Level 3	4	9	10	2
Level 4	0	3	0	22

Table 13: CFM6: K-Nearest Neighbours Confusion Matrix

	Level 1	Level 2	Level 3	Level 4
Level 1	13	7	3	2
Level 2	5	15	3	2
Level 3	10	10	5	0
Level 4	3	1	1	20

Table 14: CFM7: Multinomial Naïve Bayes Confusion Matrix

	Level 1	Level 2	Level 3	Level 4
Level 1	15	0	7	3
Level 2	10	7	3	5
Level 3	6	5	12	2
Level 4	1	6	0	18

Table 15: CFM8: Neural Model Confusion Matrix

	Level 1	Level 2	Level 3	Level 4
Level 1	21	1	3	0
Level 2	3	19	1	2
Level 3	4	6	15	0
Level 4	0	1	0	24

Appendix B. Example of a Regular Few-Shot System Role from the LLAMA Model

<|begin_of_text|><|start_header_id|>system<|end_header_id|>

Je bent een text-simplification model voor Nederlands met niveau 0. Enkele voorbeelden van een niveau 0 tekst:

Voorbeeld 1:
De vlucht van Frankfurt naar Windhoek duurt veel langer: ongeveer tien uur. “We krijgen een warme maaltijd aan boord”, zegt opa. “Daar komen ze er al mee aan”. Ik klap mijn tafeltje naar beneden en inspecteer de potjes en doosjes op mijn dienblad. Wat zou mama vanavond eten? “Smakelijk, Silke!” zegt opa, die een stuk kip naar zijn mond brengt. “Ik weet niet of ik alles op krijg”, zucht ik. “Van mama moet ik mijn bord leegeten”. “In een vliegtuig gelden andere regels”, stelt opa me gerust. “Je eet waar je zin in hebt en de rest laat je staan”. Als de steward mijn dienblad heeft weggenomen, voel ik me loom worden. Ik druk op een knop om mijn rugleuning te verstellen en leun naar achteren. Mijn ogen vallen dicht. Af en toe is er turbulentie en schudt het vliegtuig door elkaar. Daar heeft de piloot ons voor gewaarschuwd.

Voorbeeld 2:
Spoorwegmaatschappij NMBS past tijdens de zomervakantie haar aanbod aan. Zo rijden er elke dag tientallen extra treinen van en naar de kust en andere toeristische bestemmingen, en zijn er ook voordelige zomerprijzen. Daarnaast rijden er minder piekuurtreinen, omdat er tijdens de vakantieperiode minder mensen naar het werk moeten of naar de les.

Van 1 juli tot 1 september rijden er zowel op weekdagen als tijdens het weekend extra treinen van en naar de grote kuststations (Oostende, Knokke, Blankenberge, De Panne). Iedere weekdag rijden er in totaal dertig extra treinen. Op weekenddagen en feestdagen zijn dat er 32. Indien er extra mooi weer wordt voorspeld, kunnen daar zowel op een weekdag als tijdens het weekend nog zes extra treinen bij komen.

Ook naar andere toeristische bestemmingen (Walibi, Pairi Daiza, Ardennen) wordt het treinaanbod tijdens de zomervakantie versterkt. Op een weekdag gaat het om dertien extra treinen van en naar de stopplaatsen van Bierges-Walibi en Houyet (afvaart van de Lesse). Op een weekenddag rijden er 24 extra treinen van en naar Bierges-Walibi, Houyet (afvaart van de Lesse) en Cambron-Casteau (Pairi Daiza).

Voorbeeld 3:
Als de zon hoog staat, de wolken juist hangen en de lucht precies dik genoeg is, dan heb je één kans op een miljoen datje schaduw begint te praten. Dat gebeurt bij Lars.

Schim, zijn schaduw, praat met de stem uit zijn hoofd. Lars wil alles weten over de wereld van Schim en wordt zelf een schaduw. Schim kruipt op zijn beurt in het lichaam van Lars. Hij geniet van alles wat hij kan voelen, ruiken, proeven, zien en horen. Mama weet dat er iets niet klopt. Zijn nichtje Geike houdt al helemaal niet van gespook. Komt Lars ooit nog wel terug?

<|eot_id|><|start_header_id|>user<|end_header_id|>

Vereenvoudig de volgende tekst naar niveau 0:
[TEXT]

<|eot_id|><|start_header_id|>assistant<|end_header_id|>

Appendix C. Example of a Synthetic Few-Shot System Role from the GEITje Model

Je bent een text-simplification model voor Nederlands met niveau 0. Hier zijn enkele voorbeelden van een versimpeling naar een niveau 0 tekst:

Voorbeeld 1:

Origineel: De intercontinentale vlucht die ons van het bruisende Frankfurt naar het exotische Windhoek zal brengen, is een aanzienlijk langere onderneming, met een geschatte duur van ongeveer tien uur. "We zullen worden voorzien van een warme maaltijd tijdens onze reis", kondigt mijn grootvader aan met een geruststellende glimlach. "Kijk, ze beginnen al met het serveren".

Ik klap mijn kleine tafeltje naar beneden en begin de verschillende potjes en doosjes die op mijn dienblad liggen te inspecteren, elk gevuld met een ander onderdeel van de maaltijd. Ik vraag me af wat mijn moeder vanavond zou eten, thuis in ons vertrouwde huis. "Eet smakelijk, Silke!" zegt mijn grootvader, terwijl hij een stuk kip naar zijn mond brengt.

Ik zucht zachtjes. "Ik weet niet of ik alles op krijg. Mijn moeder staat erop dat ik mijn bord altijd leeg eet". Mijn grootvader stelt me gerust met zijn wijze woorden: "In een vliegtuig gelden andere regels. Je eet waar je zin in hebt en de rest laat je staan".

Nadat de steward mijn dienblad heeft weggenomen, voel ik een gevoel van loomheid over me heen komen. Ik druk op een knop om mijn rugleuning te verstellen en leun naar achteren, mijn ogen vallen langzaam dicht. Af en toe is er turbulentie en schudt het vliegtuig door elkaar, een fenomeen waar de piloot ons vooraf voor heeft gewaarschuwd. Maar ondanks deze kleine onderbrekingen, blijf ik in een staat van rust, wetende dat we veilig zijn in de bekwame handen van onze piloot.

Versimpelde versie naar niveau 0: De vlucht van Frankfurt naar Windhoek duurt veel langer: ongeveer tien uur. "We krijgen een warme maaltijd aan boord", zegt opa. "Daar komen ze er al mee aan". Ik klap mijn tafeltje naar beneden en inspecteer de potjes en doosjes op mijn dienblad. Wat zou mama vanavond eten? "Smakelijk, Silke!" zegt opa, die een stuk kip naar zijn mond brengt. "Ik weet niet of ik alles op krijg", zucht ik. "Van mama moet ik mijn bord leegeten". "In een vliegtuig gelden andere regels", stelt opa me gerust. "Je eet waar je zin in hebt en de rest laat je staan". Als de steward mijn dienblad heeft weggenomen, voel ik me loom worden. Ik druk op een knop om mijn rugleuning te verstellen en leun naar achteren. Mijn ogen vallen dicht. Af en toe is er turbulentie en schudt het vliegtuig door elkaar. Daar heeft de piloot ons voor gewaarschuwd.

Voorbeeld 2

Origineel: Gedurende de zomermaanden ondergaat het dienstenaanbod van de Belgische spoorwegmaatschappij NMBS een aanzienlijke transformatie. Er worden dagelijks tientallen extra treinen ingezet om reizigers naar de kust en andere toeristische bestemmingen te vervoeren, en er worden ook voordelige zomertarieven aangeboden. Daarnaast wordt het aantal piekuurtreinen verminderd, aangezien er tijdens de vakantieperiode minder mensen naar hun werk of naar school moeten reizen.

Vanaf 1 juli tot en met 1 september worden er zowel op weekdagen als in het weekend extra treinen ingezet om reizigers naar de grote kuststations (Oostende, Knokke, Blankenberge, De Panne) te brengen. Op weekdagen worden er in totaal dertig extra treinen ingezet, terwijl er op weekend- en feestdagen twee extra treinen worden toegevoegd, waardoor het totaal op 32 komt. Als er bijzonder mooi weer wordt voorspeld, kunnen er zowel op een weekdag als tijdens het weekend nog zes extra treinen worden toegevoegd.

Ook het treinaanbod naar andere toeristische bestemmingen, zoals Walibi, Pairi Daiza en de Ardennen, wordt tijdens de zomervakantie versterkt. Op een weekdag worden er dertien extra treinen ingezet naar de haltes van Bierges-Walibi en Houyet (voor de afvaart van de Lesse). Op een weekenddag worden er 24 extra treinen ingezet naar Bierges-Walibi, Houyet (voor de afvaart van de Lesse)

en Cambron-Casteau (Pairi Daiza). Dit verhoogde aanbod zorgt ervoor dat reizigers gemakkelijker toegang hebben tot deze populaire toeristische bestemmingen tijdens de zomermaanden.

Versimpelde versie naar niveau 0: Spoorwegmaatschappij NMBS past tijdens de zomervakantie haar aanbod aan. Zo rijden er elke dag tientallen extra treinen van en naar de kust en andere toeristische bestemmingen, en zijn er ook voordelige zomerprijzen. Daarnaast rijden er minder piekuurtreinen, omdat er tijdens de vakantieperiode minder mensen naar het werk moeten of naar de les.

Van 1 juli tot 1 september rijden er zowel op weekdagen als tijdens het weekend extra treinen van en naar de grote kuststations (Oostende, Knokke, Blankenberge, De Panne). Iedere weekdag rijden er in totaal dertig extra treinen. Op weekenddagen en feestdagen zijn dat er 32. Indien er extra mooi weer wordt voorspeld, kunnen daar zowel op een weekdag als tijdens het weekend nog zes extra treinen bij komen.

Ook naar andere toeristische bestemmingen (Walibi, Pairi Daiza, Ardennen) wordt het treinaanbod tijdens de zomervakantie versterkt. Op een weekdag gaat het om dertien extra treinen van en naar de stopplaatsen van Bierges-Walibi en Houyet (afvaart van de Lesse). Op een weekenddag rijden er 24 extra treinen van en naar Bierges-Walibi, Houyet (afvaart van de Lesse) en Cambron-Casteau (Pairi Daiza).

Voorbeeld 3:

Origineel: Wanneer de zon haar hoogste punt in de hemel bereikt, de wolken zich op een specifieke manier positioneren en de luchtdichtheid precies goed is, dan ontstaat er een zeldzaam fenomeen waarbij je schaduw begint te praten. Dit is precies wat er gebeurt bij een jongen genaamd Lars.

Schim, de naam die Lars aan zijn pratende schaduw heeft gegeven, communiceert met de stem die hij in zijn hoofd hoort. Gedreven door nieuwsgierigheid naar de wereld van Schim, ondergaat Lars een transformatie en wordt hij zelf een schaduw. Op zijn beurt neemt Schim de kans om in het lichaam van Lars te kruipen. Hij geniet intens van alles wat hij kan voelen, ruiken, proeven, zien en horen - sensaties die voor een schaduw onbekend zijn.

Lars' moeder merkt dat er iets niet klopt. Zijn nichtje Geike, die absoluut niet van bovennatuurlijke verschijnselen houdt, is ook onrustig. De vraag blijft hangen: zal Lars ooit nog terugkeren naar zijn normale zelf? Het antwoord blijft onzeker in deze mysterieuze en fascinerende wereld van schaduwen en menselijke vormen.

Versimpelde versie naar niveau 0: Als de zon hoog staat, de wolken juist hangen en de lucht precies dik genoeg is, dan heb je één kans op een miljoen datje schaduw begint te praten. Dat gebeurt bij Lars.

Schim, zijn schaduw, praat met de stem uit zijn hoofd. Lars wil alles weten over de wereld van Schim en wordt zelf een schaduw. Schim kruipt op zijn beurt in het lichaam van Lars. Hij geniet van alles wat hij kan voelen, ruiken, proeven, zien en horen. Mama weet dat er iets niet klopt. Zijn nichtje Geike houdt al helemaal niet van gespook. Komt Lars ooit nog wel terug?

Vereenvoudig de volgende tekst naar niveau 0:

[TEXT]