Sequence-to-Sequence Automatic Sentence Simplification

Development & Evaluation

Noof Abdullah Alfear

Doctor of Philosophy

University of York
Department of Computer Science

June 2024

Abstract

Sentence simplification involves the transformation of complex sentences into simpler versions, addressing syntactic and lexical aspects through operations such as rephrasing, addition, deletion, splitting, or substitution. The Sequence-to-Sequence (Seq2Seq) model has demonstrated significant efficacy across various natural language processing tasks, making it a primary focus of our research.

The thesis's first focus is the evaluation aspect of sentence simplification models. We are trying to minimize the dependency on human to evaluate the models outcomes. Our study examines the correlation between existing text simplification evaluation metrics and human judgment. It identifies evaluation metrics that exhibit strong alignment with human assessments, offering insights into metrics suitable for future research.

The second focus of this thesis is different Seq2Seq models with different configurations from dataset to models architecture. The thesis compares the performance of Gated Recurrent Units (GRUs) and Long Short-Term Memory (LSTM) units within Seq2Seq models for sentence simplification tasks. It evaluates which recurrent unit type, GRU or LSTM, achieves superior performance in encoder-decoder architectures equipped with attention mechanism. Also, the impact of text embedding techniques on Seq2Seq sentence simplification models is explored. The significance of embedding methods in determining the quality of simplification outcomes is discussed, highlighting their role in model performance. Moreover, the research investigates the potential of fine-tuning pre-trained Large Language Models (LLMs) for simplifying complex sentences. It assesses the integration of LLMs into Seq2Seq models and their adaptation to the simplification task, aiming to leverage the benefits of pre-trained language representations.

The third focus of this thesis is examining the efficacy of fine-tuning domain-specific LLMs on domain-specific datasets to enhance the quality of simplified sentences. It explores whether domain-specific LLMs outperform general LLMs when tailored and fine-tuned for specific domains, contributing insights into domain-specific sentence simplification.

Through these investigations, the thesis contributes to advancing the understanding and development of Seq2Seq models for automated sentence simplification, addressing various facets from evaluation metrics to model architecture and fine-tuning strategies.

Dedication

I would like to dedicate this thesis to my beloved father Abdullah who sadly passed away during my study, supportive mother, Fowziah, loving husband, Abdullah, and amazing daughters: Yara, Deem and Dana . . . You will always inspire me to achieve more in life . . .

Table of Contents

List of Figures

List of Tables

Acknowledgements

I praise and thank Allah SWT for his greatness and for giving me the strength during my PhD journey.

I would like to express my sincere gratitude to my supervisor Dr. Dimitar Kazakov for the continuous support, patience and motivation. Additionally, I extend my thanks to Prof. Hend Al-Khalifah, for her insightful encouragement, valuable feedback and support during my study, which significantly contributed to the enhancement of this thesis.

Special thanks also go to the research administrators and IT staff of the Computer Science Department, who have been consistently helpful and responsive to my requests.

This research was conducted using the Viking Cluster, a high-performance computing facility provided by the University of York. I extend my gratitude to the University of York High Performance Computing service and the Research Computing team for their valuable computational support.

I am deeply thankful to King Saud University for their financial support, which made it possible for me to pursue my studies at the University of York.

At the end, I wish to extend my deepest gratitude to my beloved parents, husband, and daughters for believing in me, and to my sister, brothers and friends for supporting me throughout my PhD journey. Without their encouragement and backing, I would not have been able to achieve what I have today.

Declaration

I declare that this thesis is a presentation of original work and I am the sole author. This work has not been previously presented for an award at this or any other university. All sources are acknowledged as references.

Some parts of this thesis have been published and presented in international peer-reviewed venue:

- Our meta-evaluation study of sentence evaluation metrics (Section 4.1), was published in the proceedings of LREC-COLING 2024 [11].

Chapter 1

Introduction

1.1 Background and Motivation

Artificial Intelligence (AI) represents an interdisciplinary field bridging science and engineering, enabling machines to perform tasks typically requiring human intelligence. This expansive domain encompasses sub-fields such as robotics, gaming, Natural Language Processing (NLP), medical diagnostics, among others [12].

Natural Language Processing (NLP) is a field of AI research that focuses on the utilization of computers to understand and manipulate natural language in various forms, including text, images, and speech, to accomplish a range of tasks with high performance. NLP has been applied across multiple domains, including Machine Translation, Information Retrieval, Question Answering, and Text Summarization [13].

A recent application of NLP is Automatic Text Simplification (ATS). ATS is the process of automatically transforming complex text into a simpler form with correct grammar, while preserving the original meaning. Complex text is defined as text with a complex grammatical structure or that contains difficult lexicons or phrases. Complex grammatical structures are sentences that contain intricate arrangements of clauses, phrases, or syntactic elements, making them more challenging to process. This complexity often arises from features such as multiple clauses (independent and dependent), embedded phrases, or the use of passive voice. For instance:
"Despite the challenging weather conditions, the explorers, who had been on their journey for weeks, continued their trek, determined to reach their destination."
This sentence illustrates complex grammatical structures due to the following reasons: the sentence begins with a dependent clause ("Despite the challenging weather conditions") that sets up the context, followed by an independent clause ("the explorers continued their trek"). The phrase "who had been on their journey for weeks" is a relative clause embedded within

the sentence to provide additional descriptive information about the explorers. The participial phrase "determined to reach their destination" adds more detail, further complicating the sentence structure. These elements combine to make the sentence grammatically intricate, requiring readers to parse several layers of meaning.
While difficult lexicons or phrases refer to the use of rare, specialized, or abstract vocabulary that is not commonly encountered in everyday communication. Such language can demand advanced comprehension or familiarity with specific domains. For example:
"The juxtaposition of the protagonist's hubris with his eventual downfall is emblematic of the classical tragic hero."
This sentence illustrates difficult lexicons and phrases because words like "juxtaposition," "hubris," and "emblematic" are not typically used in casual conversation. These terms are often encountered in academic or literary contexts, making them less accessible to general readers. By combining such specialized vocabulary, the sentence exemplifies how difficult lexicons can increase complexity and challenge readers unfamiliar with the terminology.
Complex text poses challenges for comprehension by certain individuals. People such as children, second language learners, or those with low literacy, as well as individuals with cognitive impairments like dyslexia, aphasia, autism, or Down syndrome, often struggle to understand complex text. Similarly, machines engaged in NLP tasks, such as parsing, machine translation, and information retrieval, may encounter challenges with complex text. In these instances, simplified text enhances reading comprehension and improves the performance and functionality of NLP tasks. The simplification can be achieved by two integral phases - lexical and syntactic. Lexical simplification entails substituting complex words or phrases with simpler synonyms that align with the context and are easier to understand. Conversely, syntactic simplification involves modifying complex grammatical structures in sentences, such as passive voice, lengthy sentences, appositions, relative clauses, coordination, and subordination. This process can involve splitting sentences, deletion, reordering, and often entails morphological changes [14]. A common technique to automate the Text Simplification process is by applying machine learning algorithms [12]. This involves enabling machines to learn and build a model that incorporates information extracted from the given data. Subsequently, this model is utilized to automatically perform the process in the future with unseen text. The model varies depending on the applied algorithm and may involve decision trees, linear regression, or neural networks. Deep learning is a recent machine learning approach that proved its effectiveness across various application domains including Natural Language Processing (NLP) [15]. It can be defined as multi-layer neural networks with a large number of parameters organized in various network architectures. Deep learning, as a subfield of machine learning, plays a significant role in addressing Automatic Text

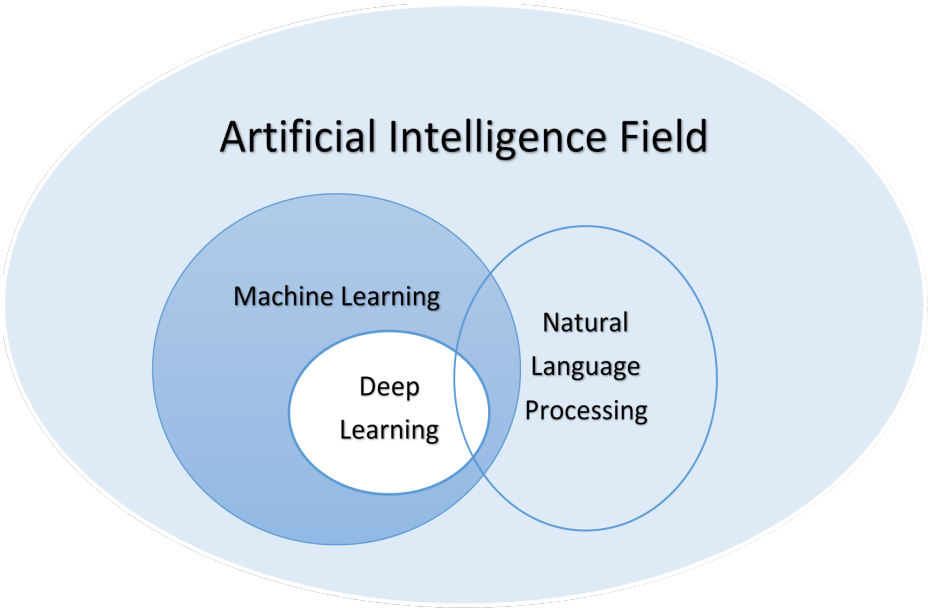


Figure 1.1 The relationship between AI, Machine Learning and NLP

Simplification (ATS) as a Natural Language Processing problem within the broader scope of Artificial Intelligence (AI) as shown in Figure 1.1.

In this thesis, research was conducted on the problem of Automatic English Text Simplification at the sentence level, encompassing both lexical and syntactic simplification aspects without focusing on a specific text transformation. Recent advancements in machine learning techniques have been explored, with a particular focus on Sequence-to-Sequence (Seq2Seq) sentence simplification models. This research covers the development and evaluation of Seq2Seq models across both general and specific domains, which aim to achieve the following objectives:

1. Simplify text,
2. Preserve the original meaning, and
3. Maintain correct grammatical structures.

Simplified text can help individuals with disabilities, second language learners, and those seeking comprehension in specialized domains beyond their expertise, such as law, medical, or scientific fields. Additionally, simplification models can aid authors in adhering to a predefined text complexity level suited for specific reader groups. Authors can specify the desired complexity level, and an intelligent model can suggest alternative text when the original text surpasses or falls short of this predefined threshold. Moreover, text simplification can serve as a pre-processing step for other NLP tasks, enhancing their overall performance.

This research aims to improve both human comprehension and machine performance, thereby advancing text understanding.

A significant contribution to this field is made through the development of models aimed at simplifying English sentences in both general and specific domains. This research not only involves the creation and refinement of these models but also includes their rigorous evaluation, thereby establishing a benchmark for future studies in the area of text simplification.

1.2 Research Questions and Objectives

This thesis studies the problem of automatically simplifying English sentences using Sequence-to-Sequence models encompassing both evaluation and development. The approaches proposed in this thesis aim to address the following research questions:

1. **Do existing text simplification evaluation metrics correlate with human judgement?** Identify the set of evaluation metrics that demonstrate a strong correlation with human judgment, which should be considered for future research.

2. **How does the performance of Gated Recurrent Units (GRUs) compare to Long Short-Term Memory (LSTM) units in Sequence-to-Sequence (Seq2Seq) sentence simplification models?** Determine which type of recurrent unit, GRU or LSTM, demonstrates superior performance for the sentence simplification task within an encoder-decoder architecture incorporating an attention-based mechanism.

3. **Does the choice of text embedding technique affect the performance of Sequence-to-Sequence (Seq2Seq) sentence simplification models, and if so, how?** Explain the significance of embedding techniques in determining the quality of outcomes produced by Seq2Seq simplification models.

4. **Can pre-trained Large Language Models (LLMs) be fine-tuned to simplify complex sentences?** Assess whether the capabilities of LLMs can be effectively integrated into Seq2Seq sentence simplification models and fine-tuned specifically for the simplification task to leverage their benefits.

5. **Does fine-tuning a domain-specific Large Language Model (LLM) on a domain-specific dataset lead to enhanced quality in simplified sentences?** Explore whether domain-specific LLMs exhibit superior capability in generating simplified sentences tailored to specific domains compared to general LLMs when fine-tuned on domain-specific dataset.

1.3 Thesis Contributions

The main contributions of this thesis are outlined as follows:

- In Chapter 4,
  1. A study of existing evaluation metrics and their correlation to human judgement was conducted. This exploration led to the identification of a set of recommended metrics for the evaluation of subsequent Seq2Seq Text Simplification research.
  2. Thresholds were established for each metric based on benchmarks set by state-of-the-art models, thereby aiding the evaluation of forthcoming simplification models.
  3. An analysis was conducted to explore the impact of the increasing number of references on the reference-based metrics scores.
- The list of contributions in Chapter 5 is as follows:
  1. A study on the impact of various embedding techniques within an encoder-decoder architecture incorporating an attention mechanism is conducted, resulting in the identification and approval of the most effective technique for subsequent experiments.
  2. An evaluation of the performance of GRU units compared to LSTM units within an encoder-decoder architecture with an attention mechanism is conducted. This comparison spans multiple datasets with varying vocabulary sizes, providing recommendations for future research concerning the choice of units and datasets.
  3. An exploration of leveraging various pre-trained LLMs within a Transformer encoder-decoder architecture to fine-tune them on multiple datasets for the English sentence simplification task. This includes the potential for sharing parameters between the encoder and decoder to optimize memory usage.
  4. An error analysis was conducted on the simplified sentences generated by our models utilizing encoder-decoder architectures with attention-based mechanism. Two primary issues were identified, and strategies were developed to address these errors.
- In Chapter 6,
  1. An extensive evaluation of the performance of various general LLM checkpoints in comparison to domain-specific LLMs was conducted, focusing on their application to domain-specific sentence simplification task. This study aimed to

determine the effectiveness of general versus domain-specific LLMs in producing high-quality simplified sentences within specialized domains. The evaluation included differentiating between cased and uncased checkpoints, as well as comparing LLMs based on mixed-domain pre-training to those with domain-specific pre-training from scratch, in the context of simplifying domain-specific sentences.

1.4 Thesis Structure

The remainder of the thesis is organised as follows:

- Chapter 2 Background and Related Work: This chapter demonstrates the theoretical background of the thesis. Additionally, it covers recent advancements related to the thesis, offering a detailed exploration of state-of-the-art simplification models.
- Chapter 3 Experimental Setting: It provides the preliminary information needed for the experiments in the chapters of contributions.
- Chapter 4 Evaluating Text Simplification Models: In this chapter a set of reference-based metrics that correlate with human judgement is identified through a comprehensive analysis and discussion. Also, it defines a standard threshold across recent state-of-the-art simplification models.
- Chapter 5 Sequence-to-Sequence models for Sentence Simplification: This chapter explores two Seq2Seq distinct model architectures and analyzes various factors that influence their performance, including architecture components, embedding techniques, and training datasets.
- Chapter 6 Sentence Simplification for a Special Domain: In this chapter, we concentrate on the simplification of medical sentences by utilizing the optimal model identified in Chapter 5 and examining various Large Language Models (LLMs), encompassing both general and domain-specific variants.
- Chapter 7 Conclusions and Future Work: The work is summarized, and potential future directions are discussed, drawing on the issues and points of interest identified during this research.

Chapter 2

Background and Related Work

The early research in Automatic Text Simplification (ATS) began with work by Chandrasekar and Srinivas [16]. They were improving parsers' performance by applying the simplification as a pre-processing step to reduce sentence length. Researchers in the field of Text Simplification are making efforts to assist various readers or improve the performance of diverse Natural Language Processing tasks. Multiple approaches have been explored to attain these objectives. However, the problem is not solved yet and the field is open for innovative ideas to evolve. Although the majority of Text Simplification research is conducted on English texts, there is an active research for several other languages, including Dutch, Portuguese, Spanish, Italian, French, and German as shown in Table 2.1.

Researchers typically aim to leverage existing knowledge and methodologies rather than reinvent the wheel. They often apply established machine learning approaches or combine two or more methods to achieve optimal outcomes. The challenge here is facing language specific characteristics that make applying existing approaches a nontrivial task. This challenge is further intensified by the limited availability of essential resources required for developing and deploying effective models. One such resource is an appropriate corpus,

Language	**Reference**
English	Automatic Induction Rules for Text Simplification [17].
French	Acquisition of Syntactic Simplification Rules for French [18].
Arabic	Simplification of Arabic Masterpieces for Extensive Reading [19].
Spanish	Spanish Text Simplification: An Exploratory Study [20].
Portuguese	Towards Brazilian Portuguese Automatic Text Simplification Systems [21].
Russian	Text Simplification for Russian as a Foreign Language [22].
German	Building a German/Simple German Parallel Corpus for Automatic Text Simplification [23].
Italian	READ–IT: Assessing Readability of Italian Texts with a View to Text Simplification [24].
Dutch	Automatic Sentence Simplification for Subtitling in Dutch and English [25].

Table 2.1 The initiation of ATS research for the world's most widely spoken languages.

which serves as the foundation for training, validating, and evaluating machine learning models. In many cases, domain-specific corpora, particularly for low-resource languages or specialized fields like medicine or law, are either unavailable or insufficient in size and diversity to capture the complexities of real-world language use. Additionally, the computational power needed to fine-tune large-scale models or run experiments efficiently poses another significant constraint. High-performance hardware, such as GPUs or TPUs, is often necessary to manage the computational demands of training advanced models, but access to such infrastructure may be limited, especially for smaller research groups or those in resource-constrained environments. This scarcity of both data and computational capacity presents a major hurdle in developing and optimizing high-quality language models.

The following sections discuss the employed methodologies highlighting their limitations, the evaluation methods and metrics that are being used to measure the performance of an Automatic Text Simplification system. These sections also discuss the existing resources for the task of English Text Simplification.

2.1 Applied Approaches

The literature reveals a variety of approaches to address the problem, starting with rule-based methods, either hand-crafted or data-driven extracted through machine learning. After the emergence of neural networks, the trend changed towards building deep learning Text Simplification models capable of achieving higher-quality simplification. Some Text Simplification models are designed to focus on individual sentences, whereas others operate on a broader scale, encompassing paragraphs or even entire documents. Text Simplification models differ also by the simplification operations performed. Some perform all simplification operations together end-to-end including paraphrasing, lexical or phrase substitution, splitting, deletion, and addition. In contrast, others perform each operation separately, a concept known as controllable. Furthermore, some models focus on a single language, while others are generalized to function as multilingual models.

2.1.1 Hand-Crafted Rule-Based Systems

Early Text Simplification systems started by extracting simplification rules manually from a dataset. The dataset consists of complex text aligned to its simplified version. Researchers preferred to have multiple formulations of the simplified text and more representative datasets in order to extract more accurate rules. They compared the complex sentences with their equivalents, focused on the differences and ending up with syntactic or lexical rules. The hand-

B-CAUSEBY-A–>A-CAUSE-B
BEC-OF-A-B –>A-CAUSE-B
BEC-A-B –>B-BEC-A
CAUSEOF–>A-CAUSE-B

Table 2.2 Hand-crafted rules [8]

The original: Almost certainly, however, the underlying cause of the war was the problem of Aquitaine.
The reformulation: Almost certainly, however, the underlying problem of Aquitaine caused the war.

Table 2.3 Example of automatic reformulation using the extracted rules [8]

crafted rules encompass various linguistic transformations, including lexical substitution, converting passive voice to active, handling apposition, deletion, and sentence splitting (involving lengthy sentences, relative clauses, coordinate, or subordinate clauses) [14]. A relative clause functions as a dependent clause that adds descriptive information about a noun in the main clause and typically begins with a relative pronoun (e.g., who, whom, whose, which, that) or a relative adverb (e.g., where, when, why). A coordinate clause represents an independent clause capable of standing alone as a complete sentence; when multiple coordinate clauses are connected by coordinating conjunctions (e.g., and, but, or, so), they form a compound sentence, each clause bearing equal grammatical weight. Lastly, a subordinate clause (or dependent clause) is unable to stand alone and instead provides additional information to the main (independent) clause, beginning with a subordinating conjunction (e.g., because, although, if, while).

One of the frameworks that applied this approach was developed by Siddharthan in 2010 [8] to perform lexical and syntactic simplification. He aligned typed dependency structures of eight different formulations for 144 sentences in a corpus. The framework was able to reformulate sentences expressing the discourse relation of causation using four lexico-syntactic discourse markers- “cause” as a verb and as a noun, “because” as a conjunction and “because of” as a preposition as shown in Table 2.2. Sample reformulation using the extracted rules is presented in Table 2.3.

This approach is constrained by the number of extracted rules, which depends on the size and the quality of the utilized dataset. Moreover, it is time consuming and very expensive to hire professionals to build such a dataset with high specifications and for researchers to spend their time extracting the rules by manually comparing between complex and simplified text.

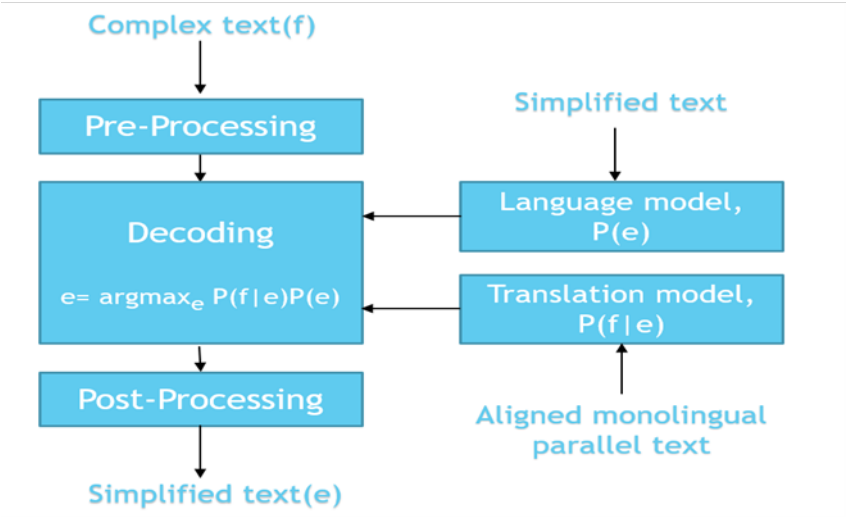


Figure 2.1 Statistical Machine Translation system [1]

2.1.2 Data-Driven Systems

Researchers found that rule-based systems suffer from various limitations mentioned earlier, which guided them to look for alternative solutions. A data-driven system was a better choice for them supported by the availability of English Wikipedia and its simplified version, Simple Wikipedia. These systems rely on the existence of extensive parallel corpora that consist of accurately aligned complex and simplified text pairs. The application of this approach took two major directions. Some researchers took advantage of such corpora to build Text Simplification models, simplifying complex text to a simpler version. Others applied machine learning techniques to learn simplification rules from the parallel corpus.

At the early stages, researchers approached the Automatic Text Simplification problem as a monolingual Statistical Machine Translation problem. They utilized existing Machine Translation systems and trained them on the complex text and its aligned simplified counterpart. As shown in Figure 2.1, the Machine Translation system comprises three main components: the translation model, which requires training on the aligned parallel text to learn a probabilistic dictionary of phrases and their corresponding simplified versions; the language model, trained over a large simplified text to indicate the likelihood of a translation being a good simplification; and the decoding component, responsible for finding the best translation with the highest probability [1].

Scientists who worked on Text Simplification used two different versions of Machine Translation systems: Phrase-based translation and Syntax-based translation. In the Phrase-based translation system, the input sentence is segmented into phrases, where a phrase is a

sequence of words not syntactically motivated. On the other side, Syntax-based translation uses information about the syntax of the sentence, when translating to other languages [1].

Phrase-Based Machine Translation (PBMT) model was mainly applied to perform lexical substitution and deletion. It poorly handled simplifications that need morphological changes, syntactic reordering, sentence splitting or insertion. PBMT does not cover syntactic simplification because it uses the least level of linguistic knowledge. In early studies, a PBMT system was employed as a black box to translate from a complex language to a simpler one. This framework primarily covered lexical substitution and simple reordering. The simplifications produced by the system were closer to the source in terms of length and order and it was overly cautious in performing other simplification operations [26]. This gradually evolved by extending the PBMT approach to allow phrasal deletion. In 2011, Coster and Kauchak modified the word alignment output by aligning phrases to NULL [27]. The deletion was constrained by the training data and the possible alignments, independent of any linguistic knowledge. As a result, they reached better performance compared to the original PBMT system without deletion. Other researchers augmented the PBMT system with a post-hoc re-ranking procedure designed to rank the output based on their dissimilarity from the source, where the most different ranked higher. For any input sentence the decoder produced a list of the ten best simplifications according to a complex scoring function including likelihood scores from the translation model and the target language model. After that, the system selected the output that was as different as possible from the source sentences based on the Levenshtein distance that counts the number of edits needed to transform the source string into the target string to ensure simplification. This system performed a small number of changes to the input covering only rephrasing phrases [28].

Drawbacks of PBMT motivated researchers to explore the Syntax-Based Machine Translation (SBMT) model, enabling them to incorporate some form of syntactic simplification. The first system proposed under this approach was by Zhu, Bernhard, and Gurevych in 2010, which covered splitting (segmentation and completion), deletion, reordering and word/phrase substitution [29]. At the training stage, they encoded probabilities for the four rewriting operations from the aligned complex-simple parallel corpus then maximized them with the Expectation Maximization (EM) algorithm. Then, the decoder generated the simplified sentences by greedily selecting the output sentence with highest probability. This model performed well for word substitution and segmentation, but other operations were problematic because they broke the coherence between the sentences. The problem with those systems is that they used a Machine Translation model as a black box and did not try to modify it for the Text Simplification task to get better results.

Rule involving lexical substitution: (VP, VP)->([ADVP [RB afterwards] VBD3 NP4], [VBD3 NP4 ADVP [RB later]])
Rule for splitting into main constituent and auxiliary sentence: (VP, VP, ST)->([VP1 and VP2], [VP1], [NP [PRP He] VP2 .])

Table 2.4 QG rules learned from the Woodsend and Lapata system [9]

In 2016, Xu, Napoles, Pavlick, Chen, and Callison-Burch were first to modify the Syntax-Based Machine Translation and were able to improve the results [30]. They focused on lexical simplification and only to a lesser extent on syntactic rewriting (reordering). Text Simplification was treated as a paraphrasing problem constrained by the fact that output should be simpler, preserving the meaning and maintaining a well formed text. Paraphrase rules from the Paraphrase Database (PPDB) were incorporated within the SBMT model as a rich source of lexical and syntactic simplification operations. PPDB was distributed with thirty-three features, and nine new features were added specifically for simplification purposes. This was the first study which showed promising correlations of automatic evaluation metrics with human evaluation.

Other researchers were also trying to learn simplification rules from aligned parallel text. In 2011, Woodsend and Lapata learned Quasi synchronous grammar (QG), a formalism that can capture structural mismatches and complex rewrite operations. It describes the alignment between Phrase Structure Trees of complex sentences and their simplified versions, covering both lexical and syntactic simplification [9]. Each QG rule described the transformations required from complex to simplified phrase subtrees as shown in Table 2.4. Then, an Integer Linear Programming (ILP) model efficiently searched through the space of QG rules, selected the most appropriate to apply, and incorporated grammatical constraints to enforce grammatical correctness. Given an input sentence, the model generated all possible simplifications starting at the root node of the parse tree, applying QG rules to subtrees until leaf nodes were reached. They were able to yield simplified output that guaranteed both a correct grammar and meaning preservation.

Another state-of-the-art lexical simplification system learned simplification rules by identifying aligned words in a sentence aligned corpus. To extract the rules, Horne et al. used a sentence aligned corpus, and induced a word alignment through GIZA++, a statistical alignment model. Then, word substitution candidates were extracted by identifying a word in the complex sentence that was aligned to a different word in the simple sentence. Three filters were applied to mitigate sentence and word alignment errors: pairs were removed if the complex word appeared in a stop list, it was constrained that both words had the

same part of speech (POS), and candidates were removed where the POS tag indicated a proper noun. After extracting aligned word pairs, they generated the simplification rules by collecting all simplification candidates that were aligned to the same complex word and added the complex word itself as a simplification candidate. To generalize the extracted rules, they added morphological variants of the words in the rules. To apply those rules, they trained a feature-based ranker using SVM_{rank} on a set of labeled simplifications and the most applicable candidate in the context was selected. The goal of the features was to capture the applicability of the word in the context of the sentence along with the simplicity of the word. The features were as follow: Candidate Probability, which is the proportion of the time was the simplification candidate aligned to a complex word in the aligned sentences, Word Frequency, Language Models and Context Frequency. Unfortunately, the extracted rules were constrained to the specific cases present in the utilized corpus, and they presupposed that the complex word had already been identified [31].

Neural Sequence-to-Sequence Systems

Neural Sequence-to-Sequence (Seq2Seq) models are a recent proposed approach and have achieved impressive results on multiple tasks. A Sequence-to-Sequence model is a deep learning model that takes a sequence of (letters, words, features of images...etc) and outputs a sequence of the same type. It attempts to build a single large Neural Network (NN), where each component is tuned based on training dataset. It consists of a large group of encoder-decoders, where the encoder transforms the source input in to a context vector and the decoder generates the output from that vector. A Recurrent Neural Network (RNN) is commonly used with Neural Seq2Seq models, achieving exceptional levels of performance in text generation tasks such as Machine Translation [32].

RNN is a type of NN where the hidden states can form a directed cycle to communicate the state history of previous input. This structure is suitable for variable-length input like sentences. For a sequence input X= (x_1, x_2, ..., x_t), at each time step t, the hidden state of the RNN $h_{(t)}$ is updated by:

$$h_{(t)} = f(h_{(t-1)}, x_t) \tag{2.1}$$

where f is a non-linear activation function. The complexity of f ranges from as simple as a sigmoid function to a Long Short-Term Memory (LSTM) unit. LSTM is a special kind of RNN and has the ability to learn long-term dependencies [33]. LSTM has a forget gate layer that decides how much information to throw away from previous cell state and an input gate layer to decide which values will be updated. Then, update the cell state and give the output

Machine Translation	Text Simplification
Most frequent words (e.g., top 15,000) are used for source and target languages. Every out-of-vocabulary word is replaced with UNK token.	Infrequent words cannot be simply ignored in the TS task. It is important to simplify them.
$\|V_s\| \approx \|V_t\|$ and $V_s \subset V_t = \phi$ Vs: Vocabulary of the source sentence Vt: Vocabulary of the target sentence	$\|V_t\| \ll \|V_s\|$ and $V_t \subset V_s$
Nearly no sharing words in source sentence X and target sentence Y.	Some or even all words in Y may remain the same after simplifying X.
The relation between source sentence and target sentence is usually one to one.	The relation could be one to many (splitting) or many to one (merging).

Table 2.5 Differences between Machine Translation and Text Simplification [10]

based on it. LSTMs have several variants, such as the Gated Recurrent Unit (GRU), where it combines the forget and input gates into a single update gate that controls how much past information is passed to the future, and has a reset gate to decide what information to forget from the past [34].

Subsequently, the introduction of the attention mechanism significantly enhanced the performance of neural Seq2Seq models. This mechanism enables the model to concentrate on relevant parts of input sentences, which improved the overall quality [32]. In 2017, the Transformer architecture was introduced, representing a novel model architecture that exhibited superior quality and increased parallelizability. This architecture required less training time and achieved new state-of-the-art performance across various tasks. It replaced recurrent layers with a multi-headed self-attention mechanism, enabling the model to capture global dependencies between input and output more effectively [7].

Indeed, numerous research studies have been conducted in this area utilizing the Neural Seq2Seq approach. In 2016, Wang, Chen, Rochford, and Qiang proposed applying a Recurrent Neural Network (RNN) encoder-decoder framework to address the Text Simplification problem. They suggested that integrating this framework could enhance a Lexical Simplification system by incorporating it as an additional linguistic feature during the ranking of candidates for complex words. Another approach involved directly applying this model to sentence simplification. However, they analysed the differences between Machine Translation and Text Simplification as listed in Table 2.5 and found that a single RNN encoder-decoder model is not able to handle all Text Simplification operations including splitting, deletion, word/phrase substitution and reordering. They proposed designing a neural network or any other classifier to determine the simplification operation that should be executed first. Then, for each operation a separate Neural Network (NN) model could be designed [10].

In 2017, Zhang and Lapata applied this approach to address the Text Simplification problem. They agreed with Wang et al. (2016) [10] that a pure Neural Machine Translation (NMT) approach is not suitable for Text Simplification, because copying words from complex

sentence to the simplified one is the most common operation compared to others like splitting, deletion, etc. In this case, the model will learn copying instead of other operations. Therefore, the researchers designed an encoder-decoder model implemented by Recurrent Neural Networks (RNN) merged with a deep reinforcement learning framework. Their model explored all possible simplifications while learning to optimize a reward function that encouraged simplicity, meaning preservation and and maintained correct grammar. The encoder read the source sentence into continuous space representations and the decoder generated the target sentence from this representation. Their encoder-decoder model used RNNs with Long Short-Term Memory (LSTM) hidden units, where the generation of a simple target word is dependent on all previous simplified words and a dynamic context vector of the source sentence. Their reward function is a weighted sum of three measures capturing simplicity, meaning preservation (relevance) and correct grammar (fluency). They used SARI [30] to reward simplicity, cosine similarity between complex sentence and system output vectors to reward relevance, and the normalized output sentence probability assigned by an LSTM language model trained on simplified sentences was applied to reward fluency. In addition, to encourage lexical simplification with simpler synonyms that match the context, a lexical simplification model was integrated with the reinforcement learning model using linear interpolation. The lexical simplification probabilities were learned from parallel corpus of complex and simplified sentences using LSTM encoder-decoder model. They proved that their model reached good simplification results compared to previous ATS systems.

There was another attempt to apply Sequence-to-Sequence neural networks on Text Simplification by Nisioi, Štajner, Ponzetto, and Dinu in 2017. The model was able to perform lexical simplification and content reduction achieving perfect grammaticality and meaning preservation with a high level of simplification. The architecture consisted of two LSTMs layers and a global attention layer on the decoder side [35].

Another Text Simplification study adapted the Sequence-to-Sequence models with memory augmented architecture called Semantic Neural Encoders (SNE). Vu, Hu, Munkhdalai, and Yu, used attention-based Sequence-to-Sequence model consisted of a Semantic Neural Encoder and an LSTM decoder. SNE is an RNN that allows each hidden state to summarize the preceding words in the sentence, but not considering the following words. The model was able to reduce the reading difficulty of the text with a correct grammar and preserve the original meaning [36].

A recent approach used Transformer architecture [7] to simplify text, which based on multi-layer and multi-headed attention with the integration of the Paraphrase Database for simplification (Simple PPDB). According to the researchers, the integrated model outperforms multiple baseline models [37].

The success of data-driven systems relies heavily on the availability of parallel corpora. However, these systems often struggle due to the lack of high-quality corpora that accurately align complex texts with their simplified versions in most languages.

2.1.3 Hybrid Systems

In 2014, the focus of research has shifted towards combining different approaches to handle both lexical and syntactic simplification more efficiently, leading to the development of hybrid systems. Narayan and Gardent designed a hybrid system, which combined semantics–based approach to handle syntactic simplification with a Phrase-based Machine Translation (PBMT) model for lexical simplifications [2]. For syntactic simplification, they encoded probabilities for splitting and deletion, which took an input as a deep semantic representation, namely, Discourse Representation Structure (DRS). Sentence splitting occurred when the same semantic entity participated in two distinct eventualities and the reconstruction of the shared element in the second simpler clause co-referred with its matching phrase in the first simpler clause. Sentence deletion was given by three different models: one for relations determining the deletion of prepositional phrases; another for modifiers (adjectives and adverbs); and a third for orphan words, which had no corresponding material in the DRS (e.g., "which"). On the other side, for lexical substitution including phrase substitution and reordering, they followed the previously mentioned Phrase-Based Machine Translation (PBMT) approach. Their simplification framework consisted of the DRS Simplification Model (DRS-SM) a probabilistic model for splitting and deletion; a PBMT model for phrase substitution and reordering; and a language model (LM) for grammaticality and fluency as shown in Figure-2.2. Their model produced simpler output that was both grammatically correct and preserved the original meaning.

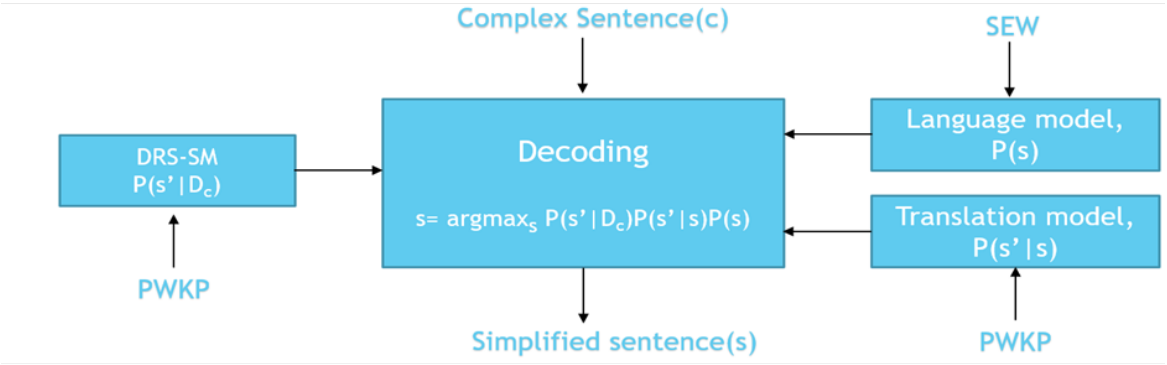


Figure 2.2 Narayan and Gardent Simplification Framework [2]

Another Hybrid Text Simplification System was designed by Siddharthan and Mandya in 2014. It was based on synchronous dependency grammars defined over typed dependencies. Their system combined manually written synchronous grammars for common syntactic simplifications with a much larger automatically acquired synchronous grammar for lexical simplification. To automatically acquire a synchronous grammar from dependency parses of aligned sentences, only the differences should be identified. Then, a generalization step is carried on to create a single rule from multiple instances and a filtering to remove rules that involved syntactic constructs. A total of 3180 lexical simplification rules were extracted from the aligned sentences. Furthermore, 136 Hand-crafted rules were manually encoded to cover syntactic simplification. Their simplification process is shown in Figure 2.3. Initially for any input sentence a dependency parse tree was obtained. After that, elementary trees (ET) were defined to be sub-sentential dependency structures containing one or more lexical items. This idea was based on Synchronous Dependency Insertion Grammar (SDIG), a tree substitution grammar defined on dependency trees. Then, ET Transfer component took place, where all the simplification rules that matched a source ET were applied iteratively. At the end, the system generated the simplified sentence from typed dependency representation by reusing the word order from the input sentence as a default, and the synchronous grammar encoded any changes in ordering [3].

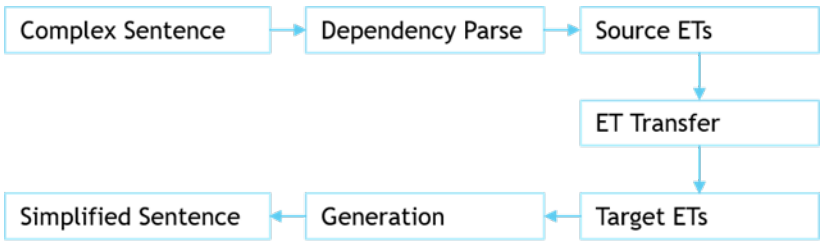


Figure 2.3 Siddharthan and Mandya Simplification Framework [3]

Hybrid systems were able to better achieve both lexical and syntactic simplification compared to earlier systems, but they are still limited by the available data resources.

2.1.4 Unsupervised Systems

Simplified corpora, which are at the heart of the previous approaches, are expensive and time-consuming to build. The absence of such corpora in most languages led researchers to explore unsupervised systems.

In 2012, Bott, Rello, Drndarevic, and Saggion followed the lexical simplification pipeline illustrated in Figure 2.4 to simplify lexicons only [38]. They trained a word vector model on a corpus consisting of eight million words. Their proposed system looks for candidates for

every single word in the text that has an entry in the thesaurus. They extracted a context vector for each substitutable word from the surrounding 9-word window. Then, they constructed a common vector for each of the word senses listed in the thesaurus by adding all the vectors of the words listed in each word sense including the substitutable word because it might be the simplest among alternatives. The most appropriate substitution set was defined as the one with the lowest cosine distance to the context vector. Then the best candidate was selected within the identified word sense based on other features like word length and frequency.

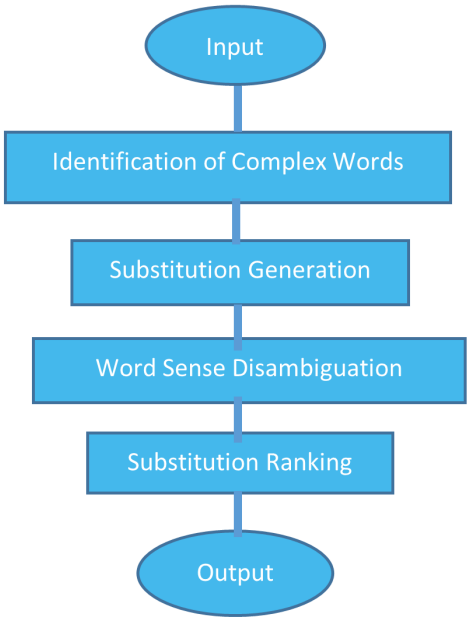


Figure 2.4 Lexical Simplification Pipeline [4]

Also, Glavaš and Štajner in 2015 [39] followed the same pipeline and were able to build unsupervised lexical simplification system (LIGHT-LS) that used word vector representations and required only regular corpus not a simplified one. Initially, they obtained vector representations for all words in a large regular text corpus. Their system considered all content words as replaceable and obtained substitution candidates by comparing semantic vectors of all corpus words with the semantic vector of the target word from the complex text. Then, the ten most similar candidates with largest cosine similarity were selected, and filtered to remove any candidates that were morphological derivations of the target word. After that, LIGHT-LS ranked the remaining candidates based on the average rank over a features set (semantic similarity, context similarity, difference of information content, and language model features). The substitution was done only if the selected candidate had

lower information content compared to the original word based on the hypothesis that word's informativeness correlates with its complexity.

These research studies are recognized because the required resources, including a huge corpus and a thesaurus, are available for most languages and if not, they are not expensive to build. On the other side, these systems are limited to consider single lexical unit simplification. They do not handle phrasal or syntactic simplification. Researchers continue to work on optimizing the lexical simplification pipeline as demonstrated by Shardlow in 2015 [4].

In 2017, Štajner and Glavaš presented a semantically-motivated Automatic Text Simplification system, which was the first system applied on the document level. They took advantage of a state-of-the-art event extraction system (EvGRAPH) and were able to eliminate large portions of irrelevant information from text, by keeping only parts of original text that belonged to factual event mentions, which are events that actually happened in real world. For syntactic simplification, the system placed each event mention in a separate sentence along with its arguments (agent, target, time and location), while the previous mentioned lexical simplification system (LIGHT-LS) was applied to replace complex and infrequent words with their simpler substitutions. Researchers proofed that their system produced more readable and simpler text. However, it was designed for text describing real-world events and cannot operate on descriptive text with few event mentions [40].

2.1.5 Controllable Systems

Text Simplification typically works end-to-end by simplifying syntactic structures and lexicon without targeting a specific goal or user group. However, controllable systems offer additional control over simplification processes to specify various aspects of the simplification. This makes simplified text more suitable for specific targets and better meet specific needs.

In 2020, Martin, de La Clergerie, Sagot, and Bordes trained a Transformer model with additional parameters under discrete parametrization mechanism, that allow users to set specific simplification controls. This allowed the model's outputs to be conditioned on attributes such as the amount of compression, as well as the degree of paraphrasing, lexical complexity, and syntactic complexity. The amount of compression is directly controlled by (#Chars), which is the length of the simplified sentence in terms of characters, which also manages the amount of content preserved between the complex and simplified sentences. On the other hand, paraphrasing is controlled by calculating the Levenshtein similarity [41] between complex and simplified sentences, which quantifies the amount of modification applied to the complex sentence through paraphrasing, addition, and deletion. For lexical complexity, a measure called WordRank was computed at the sentence level. This involved taking the third quartile of the log-ranks (inverse frequency order) of all words in a sentence.

The WordRank of the simplified sentence was then divided by that of the complex sentence to obtain a ratio. Lastly, syntactic complexity was controlled by approximating the ratio of the maximum depth of the dependency tree of the complex sentence to that of the simplified sentence [42].

In 2021, Maddela, Alva-Manchego, and Xu proposed a controllable approach that regulates various attributes of the simplified text, such as number of sentence splits, length, and number of words copied from the complex text. Their model combines linguistically-motivated syntactic rules with data-driven neural models to improve the diversity and controllability of the simplifications. Given a complex sentence, the model generates a set of intermediate simplifications incorporating both splitting and deletion. These intermediate sentences are then used as follows: one of them will be selected by a neural ranking model and then rewritten by the paraphrasing component to produce the final output; and a subset of these sentences will be used for data augmentation to train the paraphrasing model [43].

Also, in 2021 Sheang and Saggion fine-tuned T5 (a Unified Text-to-Text Transfer Transformer) [44] on the Text Simplification task, incorporating a controllable mechanism to regulate system outputs and generate adapted text for different target audiences. T5 is pre-trained on a number of supervised and unsupervised tasks such as machine translation, document summarization, question answering, classification tasks, and reading comprehension. They used the same control tokens as Martin et al. model [42] including the number of characters (#Chars) to control compression ratio, Levenshtein similarity for paraphrasing, WordRank for lexical complexity, and the depth of dependency tree for syntactic complexity. Additionally, they introduced word ratio (#Words) as a new control token to manage word length, since longer words tend to be more difficult to read. #Words represents the ratio of the number of words between the complex sentence and its simplified version [45].

2.2 English Data Resources

2.2.1 General Domain

At Automatic Text Simplification (ATS) research early stages, researchers were depending on manually created datasets to help establishing ATS systems on top of them. The quality of these systems was limited to the cases covered by the datasets, which was very limited. For example, Siddharthan (2010) used a corpus that contains 144 sentences each with seven lexico-syntactic reformulations and manually extracted the rules from this corpus [8].

Subsequently, multiple corpora were developed from various sources to support researchers' objectives, focusing either on sentence-level simplification, document-level simpli-

fication, or specific simplification tasks such as paraphrasing, lexical simplification, splitting, or deletion. In the following section, we discuss the high-quality sentence-level English corpora that encompass both lexical and syntactic simplification.

Wikipedia

The occurrence of the Simple English Wikipedia played a principal role in applying Machine Translation (MT) approaches to solve the Text Simplification problem. Simple English Wikipedia is using less complex words or phrases and easier grammatical structures compared to original Wikipedia. It has dominated ATS research since 2010, and was used together with the original Wikipedia to create parallel text and train MT based simplification systems or to automatically learn simplification rules. However, one of the leading reasons that ATS systems failed to generate simplified text is that parallel English Wikipedia corpus (PWKP). PWKP corpus contains around 108K sentence pairs, which are automatically aligned sentences between original Wikipedia and Simple Wikipedia created by Zhu, Bernhard, and Gurevych (2010) [29] and it became popular because of its size and availability. A study in 2016 proved that parallel Wikipedia simplification corpus contains a large portion that is not much simpler, or not accurately aligned. Researchers manually inspected the PWKP corpus, randomly chose 200 one-to-one sentence pairs because one-to-many sentence splitting cases consists only 6.1% of PWKP. The inspection results are clarified in Table 2.6 [5]. Moreover, models trained on Wikipedia generalize poorly to other unobserved text. This analysis results emphasize that researchers should be careful while using PWKP corpus [5].

<table>
<tr><td rowspan="2">Not Aligned
17%</td><td rowspan="2">Not Simpler
33%</td><td colspan="3">Real Simplification
50%</td></tr>
<tr><td>Deletion Only
21%</td><td>Paraphrase Only
17%</td><td>Deletion & Paraphrasing
12%</td></tr>
</table>

Table 2.6 Percentage breakdown obtained from manual examination of PWKP [5]

In 2011, Coster and Kauchak generated a parallel simplification corpus by aligning sentences between English Wikipedia and Simple English Wikipedia. After the alignment process, they extracted a final set of 137K aligned sentence pairs [46]. In 2013, they updated the dataset with recent Wikipedia data and improved text processing techniques, resulting in a dataset containing 167K aligned sentence pairs [47].

In 2017, Zhang and Lapata introduced WikiSmall and WikiLarge. WikiSmall is the parallel corpus- PWKP, which has been created by Zhu et al. in 2010 [29]. They split the dataset into training, development and testing sets. The test set consisted of 100 complex-simple sentence pairs. The training set contains 89,042 sentence pairs after they removed

duplicates and test sentences. A random sample of 205 sentence pairs was selected for development, with the remaining sentences allocated for training. Also, they constructed WikiLarge, a larger Wikipedia corpus by combining previously created simplification corpora. Specifically, they aggregated the aligned sentence pairs in Kauchak (2013) [47] and the aligned and revised sentence pairs in Woodsend and Lapata (2011) [9], where Wikipedia was utilized to create a parallel simplification corpus in two ways: by aligning sentences from MainEW with their corresponding SimpleEW counterparts, and by extracting training instances from revision histories in SimpleEW, leveraging Wikipedia's collaborative editing process. Additionally, they incorporated Zhu's (2010) WikiSmall dataset [29], and the development and test sets of the TurkCorpus[30]. After removing duplicates and sentences in development and test sets, the resulting training set contains 296,402 sentence pairs [48].

In 2020, Jiang, Maddela, Lan, Zhong, and Xu created a new version of Wikipedia corpus by aligning sentences between English Wikipedia and Simple English Wikipedia. First, they extracted article pairs from English and Simple English Wikipedia by leveraging Wikidata, a well-maintained database that indexes named entities and events, etc., and their Wikipedia pages in different languages. They found this method to be more reliable than using page titles [46] or cross-lingual links [29] [9], as titles can be ambiguous and cross-lingual links may direct to a disambiguation or mismatched page. A total of 138,095 article pairs were extracted. Then, they crowdsourced the sentence alignment annotations for 500 randomly sampled document pairs. This manually labeled dataset named WIKI-MANUAL with a train/dev/test split of 350/50/100 article pairs. They designed a novel neural CRF alignment model, which utilizes fine-tuned BERT [49] to measure semantic similarity and leverages the similar order of content between parallel documents, combined with an effective paragraph alignment algorithm. Finally, they trained their alignment model on this annotated dataset to automatically align sentences for all the 138,095 document pairs in Wikipedia. By applying their alignment model to all the 138,095 article pairs, new simplification dataset was constructed, WIKI-AUTO. WIKI-AUTO has 604K non-identical aligned and partially aligned sentence pairs. Non-identical sentences differ in structure or wording while preserving the original meaning. Partially aligned sentence pairs, where only a portion of the information overlaps between the two sentences, allowing for partial preservation of meaning and structure [50].

On the other side, some researchers worked hard to create a subset of Wikipedia aligned corpus for assessing sentence simplification. In 2016, Xu et al. created their own evaluation dataset, TurkCorpus, by selecting 2,350 sentence pairs from the PWKP dataset that were similar in length. They collected eight reference simplifications for each sentence pair, which were manually created by human annotators. These reference simplifications focused

more on paraphrasing (word substitution and reordering) rather than deletion or splitting, aligning with the specific goals of their research. Sentences were partitioned into 2,000 for development and 359 for testing. This approach enabled them to improve the performance of their system significantly [30].

In 2020, Alva-Manchego, Martin, Bordes, Scarton, Sagot, and Specia introduced ASSET, a new dataset for assessing sentence simplification in English. ASSET is an extended version of the TurkCorpus [30] utilizing the same original sentences but incorporating crowdsourced multi-reference manual simplifications. These simplifications encompass a richer set of rewriting transformations, including splitting, paraphrasing, and deleting. ASSET stands for **A**bstractive **S**entence **S**implification **E**valuation and **T**uning, a new dataset for tuning and evaluation of automatic Sentence Simplification models. ASSET consists of 23,590 human simplifications associated with the 2,359 original sentences from TurkCorpus, where each original sentence rewritten ten times. Through quantitative and qualitative experiments, they showed that simplifications in ASSET are better at capturing characteristics of simplicity when compared to other standard evaluation datasets for the task [51].

Newsela

In 2015, Xu et al. [5] introduced Newsela [52], a new simplification dataset consists of 1,130 news articles. Each article was re-written four to five times targeting children at different grade levels (1 represents the least simplified level and 5 the most simplified one) by professional editors at Newsela, a company that produces reading materials for pre-college classroom use. Researchers commented that Newsela unlike Simple Wikipedia has a well-defined objective, which is supporting teachers to prepare curricula that match the English language skills required at each grade level. Those articles are classified in to four or five levels based on Lexile readability measure [53], which is commonly used to measure text complexity level and users' readability scores. Newsela is pleased to make resources available for academic research in text difficulty, text simplification, and other disciplines that support their mission of unlocking the written word for everyone. To request access to the corpus, researchers are supposed to fill an electronic form and sign an agreement to protect the data privacy. Xu et al. designed an alignment algorithm for the Newsela corpus based on Jaccard similarity and aligned sentences in the simpler form to the sentences in the immediate complex level. The similarity between aligned pair was measured by the overlap between word lemmas based on the following equation:

$$Sim(s_1, s_2) = \frac{|Lemmas(s_1) \cap Lemmas(s_2)|}{|Lemmas(s_1) \cup Lemmas(s_2)|} \tag{2.2}$$

Then, sentences were aligned into groups across the original and the four simplification levels for each article. They conducted another manual examination on a random sample of sentence pairs from Newsela like the one they did for PWKP [5]. The manual examination was based on 50 random sentences from the original Newsela aligned to level 2 sentences and 50 sentences from the original aligned to level 4 and the results are shown in Figure 2.5.

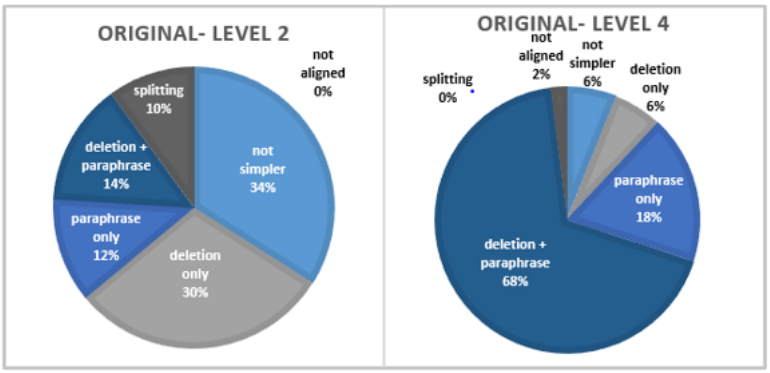


Figure 2.5 Manual Examination of Newsela Sentences [5]

Automatically aligning Newsela is more straightforward and reliable compared to Original-Simple Wikipedia because Newsela editors were carefully producing simplified text with high quality. Moreover, Newsela will make it easier for researchers to define Reading Grade Level for readers in different language understanding levels.

The shortage of resources remains a significant challenge for ATS research, particularly for most languages. The required resources vary depending on the approach applied, but in all cases, they must be comprehensive and representative. For instance, if a corpus of simplified text is needed to train an ATS system, it must encompass a sufficient variety of samples covering the different types of simplification the system aims to achieve. Additionally, if an aligned parallel corpus is required, this corpus must be accurately aligned to ensure quality training data. High-quality input samples are critical for achieving optimal output since ATS systems are highly dependent on machine learning techniques. While Simple Wikipedia has historically been used, it has shown limitations that minimize the effectiveness of ATS systems due to its restricted quality. Conversely, the Newsela dataset offers a more robust resource, providing incremental levels of simplified text aligned with different school grades, making it a valuable tool for improving the quality of ATS models.

Along with the development of WIKI-AUTO, Jiang et al. introduced Newsela-AUTO, which is much larger and of better quality compared to the previous aligned version of Newsela. First, they created manually annotated sentence-aligned datasets: NEWSELA-MANUAL with 50 article sets with a 35/5/10 split for train/dev/test, respectively. They annotated sentence alignments for article pairs at adjacent readability levels (e.g., 0-1, 1-2)

as the alignments between non-adjacent levels (e.g., 0-2) can be then derived automatically. After that, their aligner was trained on this dataset and automatically aligned sentences in the remaining article groups in Newsela to create a new sentence-aligned dataset, NEWSELA-AUTO, which consists of 666K sentence pairs predicted as aligned and partially-aligned [50].

2.2.2 Medical Domain

There has been a recent interest in simplifying medical text to to make it accessible to a broader range of readers, aiming to help them maintain good health. In this section, we discuss publicly available datasets designed for sentence simplification task in medical domain, originating from sources other than Newsela or Wikipedia.

CELLS

Guo et al. found that biomedical literature accessibility was limited to experts due to complex linguistic structures and specialized terminology of expert-authored content. Researchers stated that COVID-19 pandemic highlighted the challenges the public encounters when trying to understand scientific information for health-related decisions. Despite being widely available in scientific papers, the necessary information is often inaccessible due to medical jargon, scientific writing styles, and a lack of sufficient scientific background. As a result, there is a necessity to convey scientific knowledge in lay language, which has motivated research on the automated generation of lay language summaries (LLSs). This includes the task of background explanation: adding external content in the form of definitions, motivation, or examples to enhance comprehensibility. As a contribution of this research CELLS was introduced as the largest dataset (63K pairs) and broadest-ranging (12 journals) parallel scientific abstracts and expert-authored lay language summaries. To develop CELLS, researchers manually reviewed biomedical journals and identified 19 with an LLS section. Scientific abstracts (source) and their aligned LLSs (target) were collected from these journals. Abstracts where LLSs are not associated with a full-length paper or non-biomedical topics were excluded. This left a set of 62,886 source–target pairs from 12 journals. To facilitate research on key lay language summaries generation subtasks, they derived subsets for simplification (SELLS), which includes sentence-level paired data, and background explanation (BELLS), which contains paragraph segment-paired data [54].

MedLane

Luo et al. recognized that patients often encounter difficulties in understanding medical issues and health diagnoses. They emphasized the necessity of simplifying medical texts to make them more readable and understandable. To benchmark this task, they constructed a new dataset named MedLane to support the development and evaluation of automated clinical language simplification approaches. The dataset consists of 12,801 training samples, 1,015 validation samples, and 1,016 testing samples. Researchers selected clinical notes from the MIMIC-III database [55], which contains de-identified data from 58,976 Intensive Care Unit patient admissions. After that, they invited six researchers familiar with medical data to annotate them. For each sentence, there were two extra senior researchers holding the Doctor of Medicine (M.D.) degree to check the annotation quality. If the translated sentences deemed to be of low quality, the senior researchers re-translated them. The annotation process maintained readability, accuracy and understandability. This dataset has some ethical consideration because it is directly extracted from the MIMIC-III database. The MIMIC-III database was performed under Health Insurance Portability and Accountability Act (HIPAA) standards, which require the removal of all the identifying data elements in the list of HIPAA (e.g., name, phone number, address, and so on). Therefore, there is no privacy issue for the data. For sharing their dataset to be publicly available, they follow the same requirement of the MIMIC-III data, where the requester must complete a recognized training for protecting human research participants and sign an agreement to protect the data privacy [56].

MultiCochrane

In 2023, Joseph et al. sourced MultiCochrane from publicly available technical abstracts and plain-language summaries (PLS) of Cochrane systematic reviews; these are comparable texts, though not parallel. These abstracts and PLS exist in several languages, aligned almost one-to-one with their English counterparts. This allows annotations and alignments to be easily adapted to create a multilingual dataset. In total, researchers used 7,755 abstract-PLS pairs from the Cochrane Library. They first created sentence-aligned MC-CLEAN of 101 abstract-PLS pairs using a mixture of manual alignments for English and semi-automatic alignments with partial verification for other languages. The rest of the abstracts were used for the creation of the large-scale noisy MC-NOISY by automatic alignments (60,058 English sentences) and filtering (29,703 English sentences) to further filter misalignments [57].

2.3 Evaluation Methods and Metrics

One major success factor for any system is the ability to evaluate its performance effectively and representatively to facilitate improvement. Automatic Text Simplification systems should also be evaluated through sophisticated methods that accurately measure output simplicity, indicated by factors such as users' reading speed and comprehension, or enhancement of NLP task performance. Additionally, maintaining correct grammar and preserving the original meaning of the input text are essential goals that must be measured.

2.3.1 Human Evaluation

The most reliable evaluation method previously employed involves assessment by human language experts. Researchers depend on human evaluators to determine the quality of outputs from simplification models. These experts assess ATS systems based on three primary factors: fluency (grammatical correctness), adequacy (preservation of meaning), and simplicity, using either a Likert scale (1-5) or a continuous scale (0-100). Lower scores indicate poor performance in maintaining the respective factor, while higher scores suggest that the simplified sentence achieves high quality. Additionally, the time required for human editors to post-edit the simplified text to ensure grammatical correctness before presenting it to the end user is sometimes measured as an indicator of a high-quality system. However, this evaluation method is both expensive and time-consuming. Furthermore, comparing the output quality of different ATS systems is challenging when evaluated by multiple experts, as human judgments are inherently subjective and vary among individuals. Therefore, alternative automatic evaluation methods are necessary to make the evaluation process faster and more reliable [58].

2.3.2 Text Readability Metrics

Early researchers did not focus on designing specific evaluation methods and metrics for the Text Simplification task; instead, they relied on existing NLP evaluation methods. Automated readability metrics, including the Flesch-Kincaid Grade Level and the FOG Index, were employed to assess the readability and ease of comprehension of a given text. However, these metrics are limited in scope, as they fail to adequately capture critical aspects such as the degree of simplicity achieved, the grammatical accuracy of the simplified sentences, and the extent to which the original meaning of the text is preserved.

Flesch-Kincaid Grade Level

The Flesch-Kincaid Grade Level (FKGL) is among the most prominent readability metrics, first introduced in the 1940s. It was designed to align text complexity with grade levels in the United States education system, providing a practical and intuitive tool for evaluating the suitability of texts for various age groups and educational stages [59]. Both the FKGL and the Flesch Reading Ease score were developed by Rudolf Flesch, with the FKGL formula later refined to more accurately correspond to U.S. school grade levels. These metrics are extensively used for evaluating the readability of simplified sentences, providing critical insights into how accessible a text is for readers with varying levels of language proficiency. By quantifying text complexity, these measures help ensure that simplified sentences are appropriately tailored to meet the needs of diverse audiences.

The Flesch Reading Ease score evaluates text readability on a scale ranging from 0 to 100, with higher scores indicating greater ease of reading. This metric assesses the complexity of sentences and words, serving as a reliable benchmark for determining text accessibility for a general audience. The formula is:

$$\text{Flesch Reading Ease} = 206.835 - 1.015\left(\frac{\text{number of words}}{\text{number of sentences}}\right) - 84.6\left(\frac{\text{number of syllables}}{\text{number of words}}\right) \tag{2.3}$$

In contrast, the Flesch-Kincaid Grade Level (FKGL) calculates readability by combining two key text statistics: the average number of syllables per word and the average number of words per sentence. Unlike the Flesch Reading Ease score, FKGL produces values correlated with U.S. school grade levels, with the lowest possible value being -3.40 and no defined upper limit. Lower FKGL scores correspond to texts that are simpler and easier to read, making this measure particularly useful for evaluating the complexity of simplified sentences. It is calculated using the following equation:

$$FKGL = 0.39\left(\frac{\text{number of words}}{\text{number of sentences}}\right) + 11.8\left(\frac{\text{number of syllables}}{\text{number of words}}\right) - 15.59 \tag{2.4}$$

However, both only depends on shallow features of the simplified sentence, such as word length and sentence length, ignoring the grammaticality or semantic adequacy of the content. Consequently, short sentences could receive good scores even if they are ungrammatical or fail to preserve the original meaning. Tanprasert and Kauchak argued that FKGL is not a proper evaluation metric for text simplification and should not be used to evaluate text simplification systems [60].

Fog Index

The Fog Index (also known as the Gunning Fog Index) is a readability metric widely used in text simplification to evaluate the complexity of sentences [61]. Developed by Robert Gunning, the index estimates the years of formal education needed to understand a text upon first reading. It is especially useful in assessing the effectiveness of simplified sentences, as it provides a quantitative measure of the clarity and readability of a text. The formula for the Fog Index is:

$$\text{Fog Index} = 0.4\left(\frac{\text{number of words}}{\text{number of sentences}}\right) + 100\left(\frac{\text{number of complex words}}{\text{number of words}}\right) \tag{2.5}$$

where complex words are words with three or more syllables that are not proper nouns, familiar jargon, or compound words.

In the context of evaluating simplified sentences, a lower Fog Index score indicates that the sentence is more accessible and easier to understand, which is typically the goal of text simplification. While useful, the Fog Index is sometimes limited in assessing meaning preservation and nuance in sentence simplification. However, it remains valuable in determining overall readability improvements and comparing text before and after simplification.

2.3.3 Text Generation Metrics

Researchers used text generation evaluation metrics such as Machine Translation, compression, and paraphrase generation metrics depending on the applied methodology. When the Machine Translation approach was applied to simplify text, Automatic Text Simplification systems were evaluated using Machine Translation evaluation metrics, such as BLEU [62], NIST [63], and TERp [64].

BLEU, NIST and TERp

BLEU is the most commonly employed MT metric for measuring the quality of simplified text. It stands for **Bi**-**L**ingual **E**valuation **U**nderstudy, and proposed by Papineni et al. in 2002 [62]. BLEU compares the n-grams overlap via precision of a system simplification with a human reference simplification, independent of position. References are sentences simplified multiple times by human editors. However, BLEU is not entirely suitable because it penalizes for operations that are common in Text Simplification such as word deletion, insertion and reordering. Nevertheless, researchers continue using BLEU because they found that it sometimes correlates with human judgment scores of adequacy and fluency, even if not with simplicity [30].

In 2016, Text Simplification Quality Assessment workshop was conducted to bring researchers working on ATS, MT quality estimation and MT automatic evaluation to adopt their metrics to the simplification task. The findings indicated that MT evaluation metrics have a high possibility to replace human evaluation of grammaticality and meaning preservation, but not suitable for simplicity assessment [65].

Alva-Mancheg, Scarton, and Specia highlighted that BLEU, while a popular metric for machine translation, shows limitations when used for sentence simplification. They found that BLEU has a relatively weak correlation with human judgments of simplicity, as it primarily measures n-gram overlap without accounting for the nuanced changes required for simplification, such as reducing complexity and maintaining meaning [66].

NIST, similar to BLEU, measures n-gram precision but adds more weight to less frequent, informative words. This helps prioritize critical words that contribute more to translation adequacy [63].

TERp (Translation Edit Rate plus) is designed to account for various edit operations beyond simple word matches. It calculates the number of edits needed to change the candidate translation into the reference translation, normalizing by the total number of words in the reference [64].

BERTScore

In 2020, Zhang, Kishore, Wu, Weinberger, and Artzi proposed a new metric, BERTScore, to evaluate Text Generation tasks. It computes a similarity score for each token in the output sentence with each token in the reference sentence. Instead of exact matches, token similarity is computed using contextual embeddings. Given a reference sentence x = $(x_1, \ldots, x_k)$ and an output sentence $\hat{x} = (\hat{x}_1, ..., \hat{x}_l)$, contextual embeddings are used to represent the tokens, and compute matching using cosine similarity. BERTScore computes precision, recall and F1 measure, which can be useful for evaluating different language generation tasks. The complete score matches each token in x to a token in $\hat{x}$ to compute recall, and each token in $\hat{x}$ to a token in x to compute precision. Then, precision and recall are combined to compute an F1 measure. Researchers recommended microsoft/deberta-xlarge-mnli contextual embeddings [67] as the best performance model, in order to have the best correlation with human evaluation [68].

2.3.4 Text Simplification Metrics

FKBLEU and SARI

In 2016, Xu et al. were the first to design metrics specially for evaluating simplification systems [30]. The proposed metrics were: FKBLEU that explicitly measures readability and SARI that implicitly measures it. FKBLEU combines previously proposed metric for paraphrase generation, iBLEU [69] which is an extension of the BLEU metric and the readability metric Flesch- Kincaid Index. Given a candidate sentence O, human references R and input text I, FKBLEU is defined as:

$$\begin{aligned} FKBLEU &= iBLEU(I,R,O) \times FKdiff(I,O) \\ FKdiff &= sigmoid(FK(O) - FK(I)) \end{aligned} \tag{2.6}$$

Sentences with higher FKBLEU values are better simplifications with higher readability [30]. Because of the computation method of Flesch-based metrics, short sentences can achieve high scores even if they are ungrammatical or do not preserve meaning. Therefore, while these metrics can measure simplicity in a shallow way, they are inadequate for comprehensive evaluation or comparison of Sentence Simplification models [28].

On the other hand, SARI compares **S**ystem **O**utput **A**gainst **R**eferences and **I**nput sentence. SARI is the arithmetic average of n-gram precision and recall of copying and insertion with the precision of deletion because overdeleting hurts readability much more significant than conservative systems:

$$\begin{aligned} SARI &= d_1 F_{add} + d_2 F_{keep} + d_3 P_{del} \\ &where\ d_1 = d_2 = d_3 = \frac{1}{3} \\ P_{operation} &= \frac{1}{k} \sum_{n=[1,\ldots,k]} p_{operation}(n) \\ R_{operation} &= \frac{1}{k} \sum_{n=[1,\ldots,k]} R_{operation}(n) \\ F_{operation} &= \frac{2 \times P_{operation} \times R_{operation}}{P_{operation} + R_{operation}} \end{aligned} \tag{2.7}$$

where k is the highest n-gram order and was set to 4 in their experiments and the operation could be add, keep or delete. SARI rewards insertion operations when system output was not in the input but occurred in the references and rewards words copied or deleted in both system output and the references. They showed that SARI correlates with human judgments

of simplicity. Consequently, this metric has become the standard measure for evaluating and comparing ATS models [30].

In 2021, Alva-Manchego et al. emphasized that SARI is more suited for sentence simplification tasks compared to BLEU. SARI focuses on measuring the quality of additions, deletions, and edits made to simplify text, making it better aligned with human judgments of simplicity. They showed that SARI has stronger correlations with human evaluations, as it directly assesses the modifications that simplification systems make, capturing essential elements like reducing complexity while maintaining meaning [66].

SAMSA

In 2018, Sulem, Abend, and Rappoport proposed a new metric called SAMSA (**S**implification **A**utomatic evaluation **M**easure through **S**emantic **A**nnotation). SAMSA was the first metric to address the structural aspects of text simplification, unlike previous metrics which focused solely on lexical aspects. It decomposes the input based on its semantic structure and compares it to the output. The advantage of SAMSA is that it does not require multiple reference simplifications for comparison and it correlates well with human judgments. However, SAMSA's main premise is that each output sentence should contain a single event from the input, which is not ideal for all simplification cases [70].

LENS

In 2023, Maddela, Dou, Heineman, and Xu introduced LENS, a Learnable Evaluation Metric for Text Simplification, trained on the SimpEval corpus. This corpus is composed of two parts: $SimpEval_{past}$, which includes 12K human ratings on 2.4K simplifications from 24 past systems, and $SimpEval_{2022}$, which contains over 1K human ratings of 360 simplifications, including text generated by GPT-3.5 [71]. Given an input text c, the corresponding system output s, and a set of n references $R = \{r_1, r_2, ..., r_n\}$, LENS produces a real-valued score $z_{max} = max_{1 \leq i \leq n}(z_i)$ that maximizes over the quality scores z_i of s in regards to each reference r_i. The model encodes all texts into vectors (c, s, r_i) using Transformer-based encoders such as RoBERTa [72], then combines them into an intermediate representation, which is then fed to a feedforward network to predict z_i. Researchers found that LENS correlates much better with human judgment than existing metrics [73].

2.4 Related Work

In the previous sections, we provided a comprehensive review of the literature on Text Simplification, covering foundational concepts, key methodologies, and relevant background information to establish a broad understanding of the field. In this section, however, we shift our focus to an in-depth discussion of specific research studies and advancements that are most directly related to the objectives of our study. Here, we explore research efforts that closely align with our work in terms of approaches, methodologies, and the challenges addressed.

2.4.1 Correlation of Automatic Metrics with Human Judgments

Since the identification of the Text Simplification problem, researchers have explored and developed various approaches, evaluation metrics, and datasets to address it and assess its outcomes. An important aspect of this development has been investigating the correlation between evaluation metrics and human judgment scores of the simplified text to demonstrate the robustness of proposed models, verify the quality of datasets, and show the effectiveness of the metrics to measure fluency, adequacy, and simplicity of the outcomes.

In 2016, Xu et al. introduced the TurkCorpus dataset, designed specifically for evaluating sentence simplification models, where each sentence is paired with eight references. They also proposed two new evaluation metrics, FKBLEU and SARI, tailored to assess simplification models. Additionally, they presented a syntax-based machine translation framework (SBMT) that utilizes the large-scale Paraphrase Database (PPDB) as a source for paraphrase rules, and tuning towards the SARI metric that handle text simplification as a paraphrasing problem. To demonstrate the benefits of their contributions, they studied the Spearman's correlation on their model between various evaluation metrics (FK, BLEU, iBLEU, FKBLEU, and SARI) and human ratings at the sentence level, utilizing multiple references instead of a single reference as in previous studies. They found that SARI achieves a much better correlation with human judgments in terms of simplicity while still capturing the notions of grammaticality and meaning preservation. In contrast, BLEU shows higher correlations with grammaticality and meaning preservation when using multiple references but fails to effectively measure the most important aspect of simplification-simplicity [30].

In 2020, with the introduction of ASSET, a dataset featuring multiple types of transformations, researchers demonstrated that simplifications in ASSET more effectively capture the characteristics of simplicity compared to other standard evaluation datasets for the task. A study questioning the suitability of popular metrics for evaluating automatic simplifications in a multiple-transformation scenario was conducted. They used publicly-available

simplifications produced by Automatic Sentence Simplification Systems: PBMT-R [28], which is a phrase-based MT model; Hybrid [2], which uses phrase-based MT coupled with semantic analysis; SBMT-SARI [30], which relies on syntax-based MT; NTS-SARI [35], a neural Sequence-to-Sequence model with a standard encoder-decoder architecture; and ACCESS [42], an encoder-decoder architecture conditioned on explicit attributes of sentence simplification. They randomly selected 100 original sentences from ASSET, and for each sentence, one system simplification was sampled. This approach was employed to ensure variability in the types of rewritings performed. The Pearson correlation was computed between the human ratings and the evaluation metrics (BLEU and SARI) using ASSET or TurkCorpus as the reference sets. BLEU shows a strong positive correlation with Meaning Preservation when using simplifications from either ASSET or TurkCorpus as references. There is also some positive correlation with Fluency judgments, but this is not consistently observed for Simplicity: no correlation is found when using TurkCorpus and only a moderate correlation when using ASSET. This aligns with previous studies indicating that BLEU is not a reliable metric for estimating simplicity [28][30][70]. In the case of SARI, correlations are positive but generally low across all criteria, and significant only for simplicity when using ASSET's references. These results show that SARI might not be suitable for the evaluation of automatic simplifications with multiple rewrite operations [51].

In 2023, another correlation study between automatic metrics and human judgments was introduced with the development of LENS, the first supervised metric for Text Simplification. Maddela et al. compared LENS to existing metrics (SARI, $BERTScore_{precision}$, BLEU, and Flesch-Kincaid Grade Level readability (FKGL)) on three datasets, which will be discussed later: $SIMPEVAL_{2022}$, WIKI-DA, and NEWSELA-LIKERT. They reported Kendall's Tau correlation for $SIMPEVAL_{2022}$, and Pearson correlation between the metric scores and the human ratings for WIKI-DA and NEWSELA-LIKERT. Empirical experiments demonstrated that LENS achieved a significantly higher correlation with human ratings on $SIMPEVAL_{2022}$, more than twice as high as the correlation scores of $BERTScore_{precision}$ and SARI, respectively [73].

Meta-Evaluation of Automatic Metrics

Meta-evaluation of automatic metrics in Text Simplification is conducted to analyze the variation of the correlation between metrics' scores and human judgments. This analysis helps determine the reliability of these metrics for evaluating simplified text.

The first meta-evaluation study was presented by Alva-Manchego et al. in 2021 across three dimensions: the level of simplicity of the system outputs, the approaches used by the simplification systems, and the set of manual references used to compute the metrics.

They investigated how well existing metrics can assess sentence-level simplifications where multiple operations may have been applied. The study was conducted on their collected dataset (WIKI-DA), together with the Simplicity Gain and Structural Simplicity datasets. Their study focused on metrics developed to estimate the simplicity of system outputs, or that have been traditionally used for this task (BLEU, SARI, SAMSA, FKGL, FKBLEU, iBLEU, and BERTScore). Then, they computed the correlations between automatic scores and human judgments via Pearson for each metric. Furthermore, Williams significance tests [74] were performed to determine if the increase in correlation between two metrics is statistically significant or not. For the first dimension, they found that metrics can more reliably score low-quality simplifications in terms of WIKI-DA. For the second dimension, correlations change based on the system type. In the WIKI-DA dataset, most metrics are better at scoring system outputs from neural Sequence-to-Sequence models. For the third dimension, combining all multi-reference datasets does not significantly improve metrics' correlations over using only ASSET in the WIKI-DA dataset. To conclude, they found that existing metrics such as SARI and BERTScore struggle to capture all the aspects and achieve a high correlation with human evaluation. These metrics fail even more when evaluating high-quality systems that have close performance, calling for a more robust and accurate metric for text simplification [66].

Datasets with human judgments

- **Simplicity Gain Dataset**

  Xu et al. (2016) created this dataset to study the suitability of metrics for measuring the Simplicity Gain of automatic simplifications. The authors simplified 93 original sentences using four Sentence Simplification systems: PBMT-R, SBMT-BLEU, SBMT-FKBLEU, and SBMT-SARI. For Simplicity Gain judgments, workers on Amazon Mechanical Turk were asked to count the number of "successful lexical or syntactic paraphrases occurred in the simplification." The judgments from five different workers were averaged to get the final score for each instance [30]. This data set has limitations that may prevent the generalization of findings based on its data. The number of evaluated instances (372) is relatively small, and these were generated by only four automatic systems, three of which have very similar characteristics. Additionally, the evaluated systems did not perform significant simplification changes, as judged by humans [66].

- **Structural Simplicity Dataset**

  In 2018, Sulem et al. created Structural Simplicity dataset to evaluate the performance of Sentence Simplification models that combine hand-crafted rules (based on a semantic parsing) for sentence splitting, with standard MT-based architectures for lexical paraphrasing. Native English speakers were asked to use a 5-point Likert scale (-2 to +2 scores) to measure Structural Simplicity, defined as whether the output is simpler than the input, ignoring the complexity of the words. The authors simplified 70 sentences using different configurations of six systems. The judgments from three different annotators were averaged to obtain the final score for each instance [75]. They further exploited this data to examine the suitability of BLEU for assessing Structural Simplicity [76].

- **WIKI-DA Dataset**

  Alva-Manchego et al. (2021) collected a more reliable dataset for evaluating the correlation between metrics and human judgments of overall simplicity. Leveraging publicly available system outputs on the test set of TurkCorpus, they collected simplifications from six systems. For each system, 100 automatic simplifications were randomly sampled, not necessarily all from the same set of original sentences, but ensuring that the system output was not identical to the original sentence. Crowdsourced ratings on a 0-100 continuous scale were collected for fluency, meaning preservation, and simplicity. For each simplification instance, 15 ratings per quality aspect were collected. This dataset is more reliable for analyzing automatic metrics in a multi-operation simplification scenario since the judgments are not tied to the correctness of a specific rewriting operation [66].

- **NEWSELA-LIKERT Dataset**

  In 2021, Maddela et al. built a new dataset with human judgements, NEWSELA-LIKERT to evaluate their Controllable Text Simplification model with Explicit Paraphrasing. Their model implemented a hybrid approach that leverages linguistically-motivated rules for splitting and deletion, and couples them with a neural paraphrasing model to produce varied rewriting styles. They asked five Amazon Mechanical Turk workers to rate the fluency, adequacy and simplicity of 100 random sentences from NEWSELA-AUTO test set generated by their model and four prior work, all of which were trained on NEWSELA-AUTO. The dataset includes simplifications from Maddela et al.'s controllable model, BERT-Initialized Transformer [50], EditNTS [77], LSTM baseline [48], and Hybrid-NG [2]. Each sentence was rated on a 5-point Likert scale [43].

- **SIMPEVAL Dataset**

  In 2023, Maddela et al. introduced SIMPEVAL, which comprises $SIMPEVAL_{PAST}$ and $SIMPEVAL_{2022}$. This corpus includes over 13K human judgments on 2.8K simplifications from 26 systems. Maddela et al. collected $SIMPEVAL_{PAST}$, which contains 12K human ratings on 2.4K simplifications from 24 systems on 100 sentences from TurkCorpus [30] and ASSET [51], to train LENS automatic metric. To evaluate LENS and other simplification metrics, they created $SIMPEVAL_{2022}$ that consists of 1,080 human ratings on 360 simplifications (6 systems on 60 original sentences) from both human and state-of-the-art models, including GPT-3.5. The authors asked in-house annotators to rate each simplification on a single 0-100 overall quality scale [73].

2.4.2 Seq2Seq Sentence Simplification Models

In this section, we introduce the models relevant to our research, providing a comparative framework to evaluate our own model's performance against established benchmarks.

DRESS

DRESS stands for a **D**eep **RE**inforcement **S**entence **S**implification model, which is an LSTM-based system with reinforcement learning. The reinforcement learning framework was employed to encourage a wider variety of rewrite operations while remaining fluent and preserving the meaning of the source [48]. This model is described in Section 2.1.2

BERT-Initialized Transformer

Jiang et al. introduced a new state-of-the-art for text simplification, where the encoder and decoder follow the BERT-base architecture. The encoder is initialized with the bert-base-uncased checkpoint and the decoder is randomly initialized [50].

Controllable T5-base model

Sheang and Saggion explored the use of Unified Text-to-Text Transfer Transformer (T5) [44] fine-tuned on Text Simplification combined with a controllable mechanism to regulate the system outputs that can help generate adapted text for different target audiences. Such a model can be adjusted to fit the need of different users without having to build everything from the ground up [45]. This model is detailed in Section 2.1.5.

2.4.3 Seq2Seq for a Special Domain

In this section, we present the models relevant to our research on simplifying English sentences in the medical domain, providing a comparative framework to assess the performance of our model against established benchmarks.

The study conducted by Joseph et al. in 2023 [57] closely aligns with our research objectives. They addressed the task of simplifying complex medical texts into multiple languages using their proprietary dataset, MultiCochrane (Section 2.2.2), which is the first sentence-aligned multilingual text simplification dataset tailored for the medical domain across four languages: English, Spanish, French, and Farsi. Their approach involved fine-tuning two models:

mT5

Joseph et al. fine-tuned separate mT5-base models [78] on different language pairs: (English, English), (English, Spanish), (English,French), and (English, Farsi).

Flan-T5 fine-tuned

Researchers also evaluated fine-tuned versions of Flan-T5-base (T5 with instruction fine-tuning) [79] for each language pair. They utilized a simple prompt, specifically prepending "Simplify this sentence:" to the input (complex) sentence. When simplifying to languages other than English, the prompt is changed to "Simplify this sentence in [LANG]:".

Chapter 3

Experimental Setting

The preceding chapters encapsulate crucial concepts within the area of Sentence Simplification research from multiple dimensions and delineate its evolution since its inception. The objective of the research outlined in the thesis is to simplify English sentences to enhance accessibility and comprehensibility for diverse readers. The primary emphasis centered on addressing the issue as a monolingual machine translation problem, translating complex language into a corresponding simpler version. This approach is known as Sequence-to-Sequence (Seq2Seq) approach,where the model processes an input sequence and generates a corresponding output sequence. The research tackles this problem from two principal perspectives: the development of neural models for simplifying English sentences and the evaluation of the simplified sentences produced by these models.

Overall, the following chapters present a thorough investigation into the development and evaluation of neural models for sentence simplification, contributing significantly to the advancement of this domain. The insights gained from this research are poised to enhance the accessibility of information, making complex texts comprehensible to a broader audience.

3.1 Evaluation Problem Set

The predominant method initially employed for evaluating simplified sentences relied heavily on assessments conducted by human language experts. These experts provided detailed qualitative judgments on various aspects of the text, such as grammaticality, simplicity, and the preservation of meaning. While this method ensured high accuracy and reliability, it was also notably time-consuming and resource-intensive, limiting its scalability.

In addition to human assessments, researchers have utilized a range of automated metrics to evaluate simplified text. These metrics measure various dimensions, including readability, complexity, compression ratio, and machine translation quality. More recently, specialized

metrics designed explicitly for text simplification have been developed. Some of these metrics focus on specific editing operations commonly employed in simplification, such as splitting, deletion, addition, or substitution of words and phrases.

Before the development and implementation of Seq2Seq models for sentence simplification, a critical question emerged: how to effectively evaluate these models to ensure they produce high-quality simplifications. This question is central to understanding the efficacy of different simplification approaches and guiding future research in this field.

Chapter 4 addresses this crucial issue in depth. It provides a detailed analysis of both traditional and novel metrics, examining them appropriately in the context of sentence simplification. Chapter 4 aims to ensure that future advancements in text simplification are guided by reliable and meaningful assessment criteria, ultimately enhancing the quality of simplified texts.

3.2 Seq2Seq Models Problem Set

With the introduction of Seq2Seq models to tackle the Sentence Simplification problem, most researchers have incorporated LSTM (Long Short-Term Memory) [80] units into their architectures. This widespread adoption of LSTM raises a pertinent question: why choose LSTM over GRU (Gated Recurrent Unit) [81]? To address this, an in-depth comparative analysis was performed to evaluate these two neural network components across several critical dimensions: the quality of generated simplified sentences, the time required for training, and the overall model efficiency, as measured by model loss. This comprehensive study aims to provide insights into how each component contributes to achieving optimal simplification outcomes while balancing computational cost and performance. A diverse range of training and development datasets was employed to ensure a well-rounded evaluation.

The comparative analysis involved rigorous testing of both LSTM and GRU models on several benchmark datasets commonly used in the field of Sentence Simplification. These tests aimed to assess not only the performance in terms of sentence simplification quality but also the computational efficiency and robustness of the models.

Furthermore, the study delved deeper into the GRU architecture by examining its performance with two distinct embedding techniques. The first technique involved self-trained embeddings, which were generated from the training and validation datasets. The second technique utilized pre-trained embeddings, specifically the widely recognized GloVe (Global Vectors for Word Representation) embeddings [82]. By incorporating these different embedding strategies, the study sought to determine how variations in input representation could influence the effectiveness of the GRU architecture in the context of Sentence Simplification.

This comprehensive comparison aimed to provide a clearer understanding of the relative strengths and weaknesses of LSTM and GRU architectures. By doing so, it aspired to guide future research and development efforts towards selecting the most effective neural network architecture and embedding techniques for Sentence Simplification task, ultimately enhancing the quality and efficiency of automated simplification systems.

The emergence of Transformer architecture [7] and the remarkable advancements in pre-trained Large Language Model (LLM) frameworks such as BERT (Bidirectional Encoder Representations from Transformers) [49] have revolutionized numerous Natural Language Processing (NLP) tasks, including Machine Translation (MT), Question Answering (QA), and Text Summarization. Leveraging the power of pre-trained LLMs has become a common strategy to enhance the performance of various NLP systems. In this study, we explore the potential of integrating pre-trained LLMs into Seq2Seq models for the task of sentence simplification.

The Transformer architecture, introduced by Vaswani et al., has shown unparalleled effectiveness in capturing long-range dependencies in sequential data, making it particularly well-suited for tasks involving text generation and understanding. The self-attention mechanism employed in Transformers allows for efficient processing of input sequences, enabling the model to attend to relevant parts of the input during both encoding and decoding stages [7].

In our study, we focus on leveraging pre-trained LLMs, such as BERT, within the Seq2Seq framework for the task of sentence simplification. By fine-tuning pre-trained LLMs on a large corpus of simplified sentences, we seek to strengthen Seq2Seq models with a deeper understanding of linguistic structures and simplification patterns. This transfer learning approach allows the model to leverage the knowledge encoded in the pre-trained LLMs, enabling it to generate high-quality simplified sentences that preserve meaning, fluency, and grammaticality.

3.3 Sentence Simplification for a Special Domain

Another primary focus of this thesis is the simplification of sentences within a specialized domain, particularly the medical field. Given the critical importance of health in human lives and the availability of high-quality aligned parallel datasets in this domain, we sought to explore the potential of sentence simplification to enhance the comprehension of health-related information and promote self-care practices among individuals.

The utilization of pre-trained Large Language Models (LLMs), particularly within the context of Sequence-to-Sequence (Seq2Seq) models, represents a cutting-edge approach that

has demonstrated significant efficacy across various Natural Language Processing (NLP) tasks. This study aims to investigate the impact of leveraging pre-trained LLMs, ranging from general LLMs like BERT [49] to specialized ones like BioBERT [83], on the quality of model outcomes in the context of medical sentence simplification. Given the specialized nature of medical text and the unique terminologies and conventions employed within this domain, the efficacy of general-purpose LLMs like BERT may be limited in capturing domain-specific aspects and producing optimal simplification outcomes.

Our investigation compares the performance of Seq2Seq models equipped with general LLMs like BERT against those utilizing specialized variants such as BioBERT. By evaluating the quality of model outcomes, we seek to study the differential impact of leveraging general versus specialized LLMs in the context of medical sentence simplification.

Chapter 4

Evaluating Text Simplification Models

In recent years, Natural Language Processing (NLP) tasks have evolved, taking on substantial responsibilities and, in certain circumstances, surpassing the capabilities of humans. Several factors contribute to this evolution, including improvements in model architectures transitioning from rule-based to Deep Neural Networks (DNN) and Large Language Models (LLMs). Moreover, the foundation and utilization of high-quality data for training, coupled with high-performance computing power equipped with high-speed GPUs and substantial memory, plays a crucial role. On the other hand, one of the most impactful factors is the ability to accurately assess the performance of the model and the quality of its outcomes.

Given the thesis's focus on simplifying English sentences using Sequence-to-Sequence (Seq2Seq) models, and before the exploration of various model architectures, a fundamental question arises: how does one effectively evaluate and compare the performance of those architectures and their respective outputs?

As mentioned in Section 2.3 the evaluation of text generated by Automatic Text Simplification (ATS) models typically involves either assessments by human language experts or the calculation of specific, well-defined metrics.

In this chapter, we had an in-depth analysis of the relationship between human evaluation and reference-based automatic evaluation metrics across different state-of-the-art sentence simplification models to define which metrics we can rely on when evaluating newly developed simplification models. Based on our findings, we recommend using LENS, BERTScore$_{Precision}$, BERTScore$_{Recall}$ and BERTScore$_{F1}$ when evaluating sentence simplification models. Although BLEU and SARI are not correlated with human judgement, we recommend to continue reporting them for comparison with previously published state-of-the-art approaches. Furthermore, we established a threshold for each of the recommended reference-based metrics based on the evaluation of state-of-the-art sentence simplification

models and gold standard dataset. This threshold aims to facilitate future evaluations and comparisons of sentence simplification models.

4.1 Meta-Evaluation of Automatic Evaluation Metrics

This study analyzes the relationship between the automatic evaluation metrics and human judgement for Sentence Simplification task across multiple simplification models. The goal is to find whether we can rely on those automatic evaluation metrics when evaluating sentence simplification models rather than depending on humans in the future. Moreover, this meta-evaluation inspects if metrics' correlations are affected by the type of the model that generated the simplifications considering that all of them trained on the same high quality parallel dataset (NEWSELA-AUTO).

The focus was on recently published reference-based metrics, specifically BERTScore and LENS along with traditional metrics commonly used for Sentence Simplification task: BLEU and SARI. SAMSA was not included in our study because its main premise is that a structurally correct simplification consists of each sentence containing a single event from the input, which does not align with the typical primary goal of general simplification [70]. General simplification involves various operations such as lexical or phrase substitution, splitting, paraphrasing, addition, and deletion, while SAMSA specifically address correct splitting. To evaluate the automatic metrics, we computed the correlations between metrics' scores and human judgements via Pearson, Spearman and Kendall's Tau. Since Pearson only detects linear relations and is sensitive to outliers, rank correlations: Spearman and Kendall's Tau helped us overcome this and can detect monotonic non-linear relationship. The difference between linear and non-linear relationships in correlation analysis lies in the type of association they describe between two variables, and this impacts the choice of correlation metric [84]. A linear relationship exists when changes in one variable are proportional to changes in another. For example, if one variable doubles, the other may double (positive correlation) or halve (negative correlation). While non-linear relationships involve variables that don't change proportionally but may still exhibit a monotonic association (when one variable consistently increases or decreases with the other, but not necessarily at a constant rate). To our knowledge, the first meta-evaluation study of Sentence Simplification automatic evaluation metrics was conducted by Alva-Manchego et al. in 2021 [66]. However, our study includes a recently published metric for text simplification, LENS [73], and the dataset used, along with Human Judgements on Simplicity [43], differs from the one used in the 2021 study. Moreover, we studied the variation of both linear and non-linear measures of correlation, namely Pearson, Spearman [84] and Kendall's Tau [85], with respect to four

different simplification models. All models were trained on the same high-quality dataset, NEWSELA-AUTO [50]. On the other hand, our study aligns with the findings of Maddela et al. [73]. However, a distinction arises regarding the NEWSELA-LIKERT dataset. While Maddela et al. exclusively reported the linear correlation, we provided both linear and non-linear correlations along with the significance levels. This choice was made because it is well-known that for ordinal data, the two non-linear correlations, Spearman and Kendall's Tau, are more appropriate measures of correlation. Furthermore, our analysis of BERTScore included examination of precision, recall, and F1 values, utilizing the best-performing model as reported by Zhang et al. [68]. In contrast, the Maddela et al. study only reported precision values [73]. We followed the approach of Alva-Manchego et al. [66] in conducting Williams significance tests [74] to assess whether the correlation between two variables is statistically significant.

4.1.1 Data

The meta-evaluation study was conducted on a dataset with human judgements - NEWSELA-LIKERT [43]. This dataset was created to evaluate the performance of Controllable Text Simplification model with Explicit Paraphrasing. It has human evaluation of the overall simplification quality of 100 random sentences from the NEWSELA-AUTO test set. The simplified sentences were generated by Maddela et al. model [43] and three other state-of-the-art previous models: HYBRID [2], EDITNTS [77] and TRANSFORMER [50], where all the models were trained on the NEWSELA-AUTO dataset. Each simplified sentence fluency, adequacy and simplicity was rated on a 5-point Likert scale by five Amazon Mechanical Turk workers, where 5 is the best and 1 is the worst. The ratings were averaged as the human ratings are fairly consistent, with very few outliers. As previously noted, $fluency$ measures whether the generated output adheres to correct grammatical standards, *adequacy* determines if the output retains the original meaning of the input sentence, and *simplicity* assesses whether the output is simplified compared to the input, considering both lexical choice and syntactic structure.

There were three additional evaluation datasets with human judgements: SIMPEVAL$_{PAST}$, SIMPEVAL$_{2022}$, and WIKI-DA. **SIMPEVAL**$_{PAST}$ contains 12K human ratings on 2.4K simplifications from 24 systems on sentences from TurkCorpus [30]. This dataset was employed to train LENS. **SIMPEVAL**$_{2022}$ consists of 1,080 human ratings on 360 simplifications from both humans and state-of-the-art models, including GPT-3.5. This dataset was utilized to evaluate LENS and other simplification metrics [73]. **WIKI-DA** released by Alva Manchego et al. in 2021 [66] with 0-100 continuous scale ratings on fluency, adequacy, and simplicity for 600 simplifications across six systems. While SIMPEVAL$_{PAST}$, SIMPEVAL$_{2022}$, and

WIKI-DA are derived from Wikipedia, NEWSELA-LIKERT is derived from news articles in NEWSELA [52]. A comparative summary of these four datasets is provided in Table 4.1.

These three datasets were excluded from the current study due to their involvement in prior metric development and evaluation. Specifically, SIMPEVAL_{PAST} and SIMPEVAL_{2022} played a key role in the development of the LENS metric [73], making them unsuitable for independent validation here. Additionally, WIKI-DA formed the basis of the initial meta-evaluation of automatic evaluation metrics led by Alva Manchego et al. in 2021 [66], serving as a foundational dataset in that study.

Dataset	Source	Size	Number of Systems
NEWSELA-LIKERT	NEWSELA-AUTO	100 sentences	4
WIKI-DA	TurkCorpus	100 sentences	6
SIMPEVAL_{PAST}	TurkCorpus	100 sentences	24
SIMPEVAL_{2022}	Wikipedia	60 sentences	6

Table 4.1 Comparison of Evaluation Datasets

4.1.2 Methodology

Initially, non-referenced basic readability metrics were applied to the NEWSELA-LIKERT dataset to gain an understanding of how sentence simplicity is assessed across various sources. The results summarized in Table 4.2, where the columns correspond to different sentence sources: *Complex* denotes the original, complex input sentences, *Reference* represents their simplified counterparts created by language experts, and the remaining columns represent the simplified outputs generated by various state-of-the-art simplification models. Meanwhile, the rows represent various non-referenced readability metrics used to evaluate these sentences. Most readability metrics fail to show significant differences in values across multiple sources, highlighting their inadequacy for effectively assessing sentence simplicity. An exception is the FOG Index, which demonstrates a stronger ability to detect sentence complexity. According to the results presented in Table 4.2, the FOG Index assigns a score of 7.5 to the complex input sentences, indicating that they are comprehensible to a seventh-grade-level reader. In contrast, the simplified output sentences generated by Maddela et al. [43] achieve a significantly lower score of 4.78, suggesting that the content is suitable for a fourth-grade-level reader. The primary limitation of most readability metrics lies in their reliance on superficial textual features, such as sentence length or word complexity, which do not adequately reflect the nuanced aspects of simplification, such as grammatical correctness (fluency), retention of the original meaning (adequacy), and true simplicity of lexicon

and syntax. This underscores the need for more advanced evaluation metrics that can comprehensively measure these critical dimensions of sentence simplification.

	Complex	Reference	HYBRID [2]	Maddela et al. (2021) [43]	TRANSFORMER [50]	EDITNTS [77]
flesch reading ease	89.89	89.28	92.93	95.67	**86.91**	87.82
flesch kincaid grade	4.5	**4.7**	3.3	2.3	3.6	3.2
fog index	**7.5**	6.64	5.64	4.78	5.18	5.28
difficult words	**24**	16	16	16	23	18
syllable count	240	**291**	184	147	159	146
lexeme count	197	**253**	158	123	127	116

Table 4.2 Basic Readability Statistics - Highest in difficulty are marked in **bold**, while the least are underlined.

Consequently, this study focuses on reference-based metrics BLEU, SARI, BERTScore and LENS to measure simplified sentences and investigate their correlations with simplicity, adequacy and fluency scores from human judgements. All the metrics were calculated at the sentence-level. For BLEU and SARI, the implementations provided by EASSE [86] were utilized in this study. On the other side, for BERTScore both the implementation with RoBERTa as the default model and also microsoft/deberta-xlarge-mnli model [67] based on the authors recommendation [68] were used. RoBERTa works well for many natural language processing (NLP) tasks, including text generation evaluation, while DeBERTa provides superior results when used with BERTScore for text evaluation. Authors recommend DeBERTa because it provides better alignment with human judgments for text quality evaluation, and the fine-tuning on MNLI makes it particularly good at capturing nuanced semantic relationships, making it ideal for scoring tasks where semantic fidelity is crucial. While RoBERTa remains the default due to its balance of speed and performance, DeBERTa-xlarge-mnli is often recommended by researchers for tasks requiring higher semantic fidelity. Using the latter can improve alignment with human evaluation benchmarks. The later was reported here because it had the best correlation with human evaluation. For LENS we used the implementation provided by Maddela et al. [73] using RoBERTa language model and $k = 3$, where k represents the number of reference sentences.

After that, the three different correlation types: Pearson, Spearman and Kendall's Tau were calculated for each metric with human judgement different aspects' scores across the four simplification models: HYBRID [2], EDITNTS [77], TRANSFORMER [50] and

Maddela et al. [43] as shown in Tables 4.3, 4.4, 4.5 and 4.6 respectively. Multiple correlations were applied to detect any kind of relationship whether it was linear or non-linear.

4.1.3 Results

The resulted correlation between the automatic evaluation metrics (BLEU, SARI, LENS, BERTScore$_{Precision}$, BERTScore$_{Recall}$, and BERTScore$_{F1}$) and the human judgement evaluation based on simplicity, adequacy and fluency are presented in Table 4.3, Table 4.4, Table 4.5 and Table 4.6 for the state-of-the-art simplification models: HYBRID [2], EDITNTS [77], TRANSFORMER [50] and Maddela et al. [43] respectively. Each table shows the three different correlations Pearson, Spearman and Kendall's Tau between each pair of automatic metrics and human judgements' aspects. All significant results with $p < .05$ are boldfaced.

	Simplicity			Adequacy			Fluency		
	Pearson	**Spearman**	**Kendall's Tau**	**Pearson**	**Spearman**	**Kendall's Tau**	**Pearson**	**Spearman**	**Kendall's Tau**
BLEU	-0.042704	-0.080227	-0.061295	**0.305895**	**0.247185**	**0.174170**	**0.221159**	0.151132	0.100834
SARI	0.098624	0.028061	0.021687	**0.281641**	0.171048	0.113217	**0.324004**	**0.220126**	**0.152129**
BERTScore$_P$	0.143842	0.068493	0.050753	0.140552	0.076047	0.049008	**0.276201**	**0.199609**	**0.143602**
BERTScore$_R$	0.085582	0.013768	0.016634	**0.364709**	**0.263457**	**0.186610**	**0.412355**	**0.323510**	**0.230593**
BERTScore$_{F1}$	0.123921	0.047653	0.035162	**0.274748**	0.181527	0.12322	**0.371118**	**0.286157**	**0.204717**
LENS	**0.284418**	**0.236678**	**0.176231**	**0.308810**	**0.265191**	**0.191197**	**0.553059**	**0.498933**	**0.365820**

Table 4.3 Correlation-HYBRID simplification model (significant correlations with $p < .05$ are **boldfaced**).

	Simplicity			Adequacy			Fluency		
	Pearson	**Spearman**	**Kendall's Tau**	**Pearson**	**Spearman**	**Kendall's Tau**	**Pearson**	**Spearman**	**Kendall's Tau**
BLEU	-0.177190	-0.140434	-0.103769	0.042482	-0.019924	-0.025588	0.056722	-0.016503	-0.010145
SARI	0.184546	0.156650	0.116005	0.174147	0.130178	0.080087	**0.208836**	0.117007	0.078169
BERTScore$_P$	0.138181	0.083496	0.055118	0.142600	0.123742	0.087933	0.118100	**0.07011**	0.081614
BERTScore$_R$	-0.011886	-0.011435	-0.005126	**0.234410**	**0.226968**	**0.159175**	0.109790	**0.323510**	0.101147
BERTScore$_{F1}$	0.077024	0.052734	0.035033	0.197942	0.193782	0.134521	0.125228	**0.286157**	0.105195
LENS	**0.496585**	**0.505682**	**0.364454**	**0.452231**	**0.344530**	**0.239375**	**0.601925**	**0.453529**	**0.326975**

Table 4.4 Correlation-EDITNTS simplification model (significant correlations with $p < .05$ are **boldfaced**).

4.1.4 Discussion

From the previous section and based on the interpretation of correlation coefficient values by Corder and Foreman [87], we can analyze the correlations across the multiple approaches. **LENS** shows moderate significant correlation with simplicity and medium to strong with

	Simplicity			Adequacy			Fluency		
	Pearson	**Spearman**	**Kendall's Tau**	**Pearson**	**Spearman**	**Kendall's Tau**	**Pearson**	**Spearman**	**Kendall's Tau**
BLEU	-0.067261	-0.075353	-0.051788	**0.378386**	**0.370779**	**0.262711**	0.177556	**0.246839**	**0.175007**
SARI	**-0.203313**	**-0.216973**	**-0.151497**	**0.275280**	**0.274311**	**0.191213**	0.139153	0.121658	0.087574
BERTScore$_P$	0.139345	0.131615	0.090240	**0.314171**	**0.307780**	**0.207120**	**0.228873**	**0.308736**	**0.187060**
BERTScore$_R$	**-0.231895**	**-0.231675**	**-0.162777**	**0.391481**	**0.411641**	**0.288427**	0.172463	**0.323510**	0.134079
BERTScore$_{F1}$	-0.056910	-0.059074	-0.046282	**0.380872**	**0.392788**	**0.265581**	**0.214277**	**0.286157**	**0.176668**
LENS	**0.288571**	**0.339714**	**0.23874**	0.156929	0.109114	0.074516	**0.429411**	**0.423822**	**0.306160**

Table 4.5 Correlation-TRANSFORMER simplification model (significant correlations with $p < .05$ are **boldfaced**).

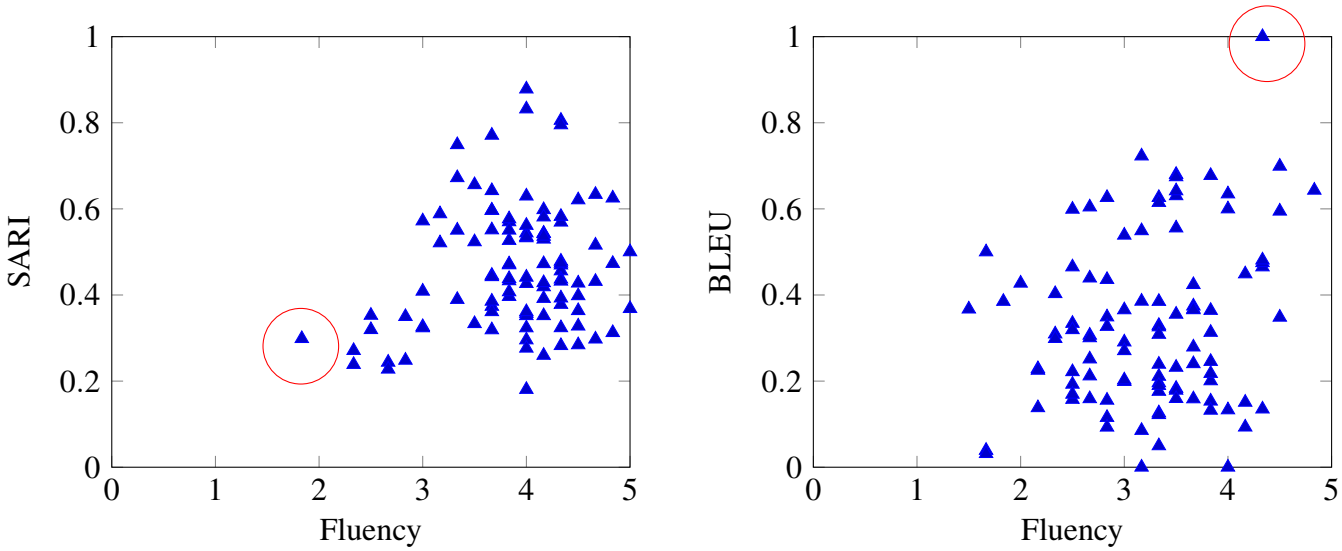
	Simplicity			Adequacy			Fluency		
	Pearson	**Spearman**	**Kendall's Tau**	**Pearson**	**Spearman**	**Kendall's Tau**	**Pearson**	**Spearman**	**Kendall's Tau**
BLEU	-0.116446	-0.112289	-0.079251	-0.119405	-0.079144	-0.061568	0.130129	0.194735	0.135584
SARI	-0.007717	0.019843	0.008117	**0.259365**	**0.304577**	**0.219893**	0.102994	0.170013	0.111771
BERTScore$_P$	0.138181	0.083496	0.055118	0.142600	0.123742	0.087933	0.117997	**0.169106**	0.081614
BERTScore$_R$	-0.011885	-0.011435	-0.005126	**0.234410**	**0.226968**	**0.159175**	0.109790	**0.323510**	0.101147
BERTScore$_{F1}$	0.077024	0.052734	0.035033	0.197942	0.193782	0.134521	0.125228	**0.286157**	0.105195
LENS	**0.284043**	**0.255757**	**0.176624**	**0.347847**	**0.367430**	**0.265194**	**0.405085**	**0.432066**	**0.310846**

Table 4.6 Correlation-Maddela et al. simplification model (significant correlations with $p < .05$ are **boldfaced**).

Fluency for all the four different approaches. Also, it has significant moderate correlation with Adequacy among all the models except TRANSFORMER.

Across the four different approaches three of them have additional control layer over the simplification of the sentence either by splitting or deletion with the paraphrasing, while TRANSFORMER is the only vanilla simplification model. TRANSFORMER shows that most of the automatic metrics correlated with some aspects of human judgement. **BLEU**, **BERTScore**$_{precision}$, **BERTScore**$_{Recall}$ and **BERTScore**$_{F1}$ have a medium significant correlation with Adequacy and Fluency. While **SARI** has absolute moderate significant correlation with Simplicity and Adequacy and **BERTScore**$_{Recall}$ has absolute moderate significant correlation with Simplicity. On the other side in Maddela et al. [43], **BERTScore**$_{precision}$, **BERTScore**$_{Recall}$ and **BERTScore**$_{F1}$ have medium significant correlation with Fluency. While **BERTScore**$_{precision}$ and **SARI** have moderate significant correlation with Adequacy. In EDITNTS and HYBRID models, **BERTScore**$_{precision}$, **BERTScore**$_{Recall}$ and **BERTScore**$_{F1}$ have moderate significant correlation with Fluency. Where **BERTScore**$_{Recall}$ correlates also with Adequacy. On the other side, **SARI** has significant medium correlation with Fluency and **BLEU** with Adequacy in the HYBRID model.

When analyzing the results and by looking at Tables 4.3, 4.4, 4.5 and 4.6, we can realize that some of Pearson correlations' values are significant, while Spearman and Kendall's Tau are not. Figures 4.1a and 4.1b show examples of this special case and how the data are distributed. Figure 4.1a, shows a scatter plot of SARI scores VS Fluency for EDITNTS



(a) EDITNTS Fluency vs SARI.

(b) HYBRID Fluency vs BLEU.

Figure 4.1 Effect of outlier sentences on Pearson correlation.

model. Where Pearson= **0.208836**, Spearman= 0.117007 and Kendall's Tau= 0.0781688, there is an outlier that affected the value of Pearson correlation. Another example is shown in Figure 4.1b for HYBRID model between BLEU and Fluency. Pearson = **0.221159**, Spearman = 0.151132 and Kendall's Tau = 0.100834. Again, the outlier affected Pearson correlation, while Spearman and Kendall's Tau show no significant correlation.

Table 4.7 highlights outlier sentences that influence the Pearson correlation by demonstrating instances of unsuccessful simplification. In example (a), the EDITNTS [76] model generates a simplified sentence that fails on multiple fronts: it contains grammatical errors and does not preserve the original meaning of the input sentence. This lack of adequacy and fluency reflects a significant failure in the model's simplification process.

In example (b), the HYBRID [33] model does not simplify the sentence meaningfully; instead, it merely truncates the original sentence. While this produces a shorter sentence, it does not improve readability or reduce complexity, leaving the sentence essentially as difficult to understand as the input.

These outlier cases illustrate how errors in simplification, whether due to flawed grammar, inadequate meaning preservation, or ineffective strategies such as mere truncation, can skew evaluation metrics. This highlights the importance of identifying and addressing such failures to ensure reliable performance assessments of simplification models.

Overall, we can conclude that LENS has medium to strong correlation with all the human evaluation aspects: Simplicity, Adequacy and Fluency. While BERTScore$_{precision}$, BERTScore$_{Recall}$ and BERTScore$_{F1}$ have medium correlation with Fluency and BERTScore$_{Recall}$

a	**original**	Gray matter is a kind of brain tissue .
	EDITNTS	gray is a kind of brain brain.
b	**original**	Nagle says the trash is a gold mine for garbage pickers .
	HYBRID	nagle says the trash is a gold mine.

Table 4.7 Outlier sentences

has moderate correlation with Adequacy. Transformer is the only model that has correlation of both BLEU with FLUENCY and SARI with Simplicity.

Following this analysis and considering our findings, LENS is recommended at the first place to measure the three distinct aspects of simplified sentences: Simplicity, Adequacy and Fluency. Additionally, for a precise evaluation of Fluency in simplified sentences, BERTScore$_{precision}$, BERTScore$_{Recall}$ and BERTScore$_{F1}$ are recommended metrics. Furthermore, BERTScore$_{Recall}$ can be used to support the assessment of meaning preservation. Nevertheless, we suggest that reporting BLEU and SARI metrics be continued to facilitate comparisons between the performance of newly developed models and previously published results by researchers.

4.2 Establishing Evaluation Metric Thresholds

In this section, thresholds are established for each of the recommended metrics based on state-of-the-art Text Simplification models and a gold standard dataset. The purpose of establishing such thresholds is to define the achievable score standards for the metrics when computed on high-quality simplified sentences. This approach will assist in measuring and comparing the performance of newly developed simplification models. An in-depth study of the selected metrics was conducted on ASSET parallel dataset [51], ChatGPT 3.5 as a sentence simplification model [71] besides the following state-of-the-art simplification models: HYBRID [2], EDITNTS [77], TRANSFORMER [50] and Maddela et al. [43].

4.2.1 Methodology

Initially as mentioned in Section 4.1.2 the metrics for the four state-of-the-art simplification models were computed on 100 random sentences from the NEWSELA-AUTO test set, where each complex sentence is aligned with a single reference sentence. On the other side for the gold standard dataset, ASSET was selected because it incorporates ten simplification references for each complex sentence crafted by qualified workers. Each simplification in ASSET is generated through the execution of multiple rewriting transformations, including

lexical paraphrasing (lexical simplification and reordering), sentence splitting, and compression (deleting unimportant information). This makes it better in capturing the characteristics of simplicity compared to other standard evaluation datasets for the task [51].

Besides the four simplification models, 100 complex sentences were randomly selected from ASSET test dataset. Then, one of the ten aligned references served as the gold standard simplification and compared to a single reference along with the complex input sentence to compute the reference-based metrics. Moreover, the power of the Large Language Model, ChatGPT 3.5 [71], was taking place. ChatGPT was provided by the same 100 complex sentences from ASSET test dataset and instructed to simplify it one by one through zero-shot prompting, where the model was not provided with any examples of simplified sentences. An illustration is presented in Table 4.8. Again, the metrics were computed by comparing the 100 simplified sentences with the complex input sentence and the same single reference to ensure fairness across the study. This is explained in Figure 4.2. After that,the threshold was established for each of the recommended reference-based metrics (BLEU, SARI, LENS and BERTScore precision, recall and F1) considering the six simplifications different sources as presented in Table 4.9.

user	Simplify the following sentence:
ChatGPT	Please provide the sentence you would like me to simplify.
user	One side of the armed conflicts is composed mainly of the Sudanese military and the Janjaweed, a Sudanese militia group recruited mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan.
ChatGPT	The Sudanese military and the Janjaweed, made up of Afro-Arab Abbala tribes, are mostly on one side of the conflict.
user	Jeddah is the principal gateway to Mecca, Islam's holiest city, which able-bodied Muslims are required to visit at least once in their lifetime.
ChatGPT	Jeddah is the main entrance to Mecca, the most sacred city in Islam, which Muslims must visit at least once in their lifetime if they are able.

Table 4.8 Sentence simplification example using ChatGPT 3.5.

From another side, the simplified sentences from ASSET gold standard and ChatGPT outcomes were compared to an increased numbers of references rather than only a single reference. This approach was taken to observe the effect of multiple references on the reference-based metrics' values.

4.2.2 Results

The values of recommended reference-based metrics computed using **single reference** along with their respective thresholds are presented in Table 4.9 and visualized in Figure 4.3.

After that, ASSET-related simplified sentences (Gold Standard and ChatGPT) were compared to **four references** randomly selected from ASSET dataset, and then this number was further increased to **nine references** to study their effect on metrics' values, as shown in

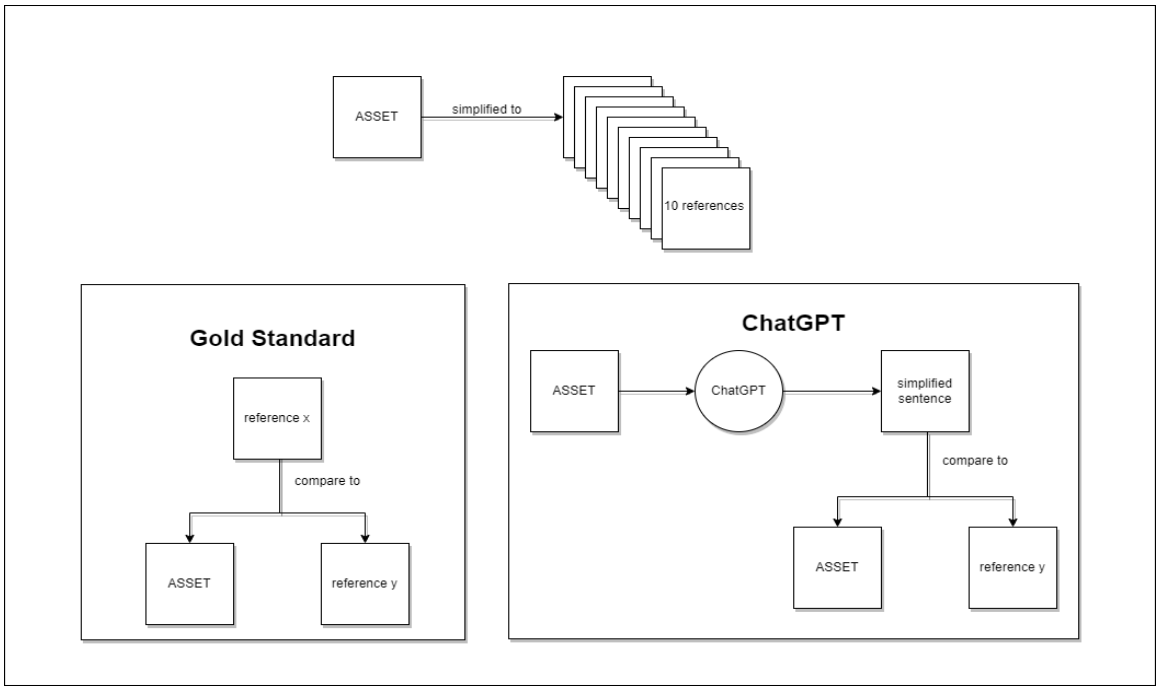


Figure 4.2 ASSET-gold standard and ChatGPT as a sentence simplification model.

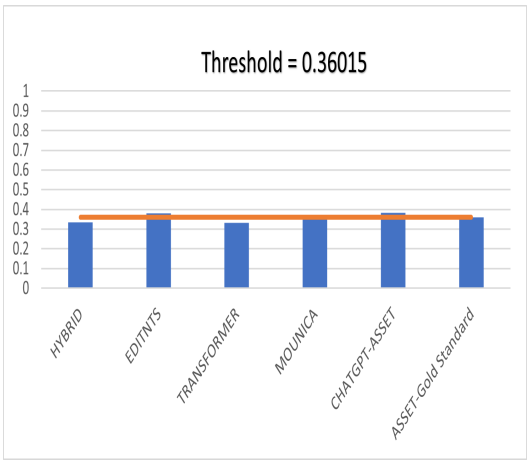
Simplified sentence source	**BLEU**	**SARI**	**LENS**	**BERTScore$_P$**	**BERTScore$_R$**	**BERTScore$_{F1}$**
HYBRID	0.333989	0.387251	20.03622	0.800073	0.783445	0.790585
EDITNTS	0.380854	0.461236	44.98601	0.835597	0.828575	0.830761
TRANSFORMER	0.331628	0.437544	48.04623	0.833351	0.823536	0.82751
Maddela et al.	0.372552	0.439279	46.93207	0.827739	0.8298	0.827992
CHATGPT-ASSET	0.38174	0.40743	46.63138	0.906786	0.902764	0.904569
ASSET-Gold Standard	0.36014	0.43942	43.29921	0.866065	0.873306	0.868820
Threshold	0.360151	0.428693	41.65519	0.844935	0.840238	0.841706

Table 4.9 Sentence Simplification metrics' values and thresholds across multiple sources-Single Reference.

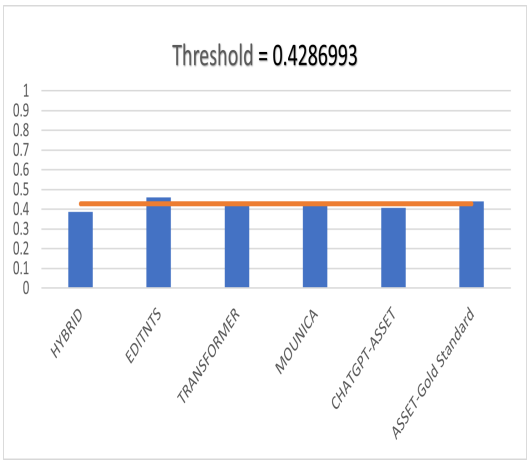
Tables 4.10 and 4.11 respectively. The impact of the increased number of references on each reference-based metric is shown in Figure 4.4.

Simplified sentence source	**BLEU**	**SARI**	**LENS**	**BERTScore$_P$**	**BERTScore$_R$**	**BERTScore$_{F1}$**
CHATGPT-ASSET	0.5772	0.50092	68.33244	0.919394	0.928035	0.918515
ASSET-Gold Standard	0.61164	0.45394	62.44666	0.915615	0.916439	0.912684

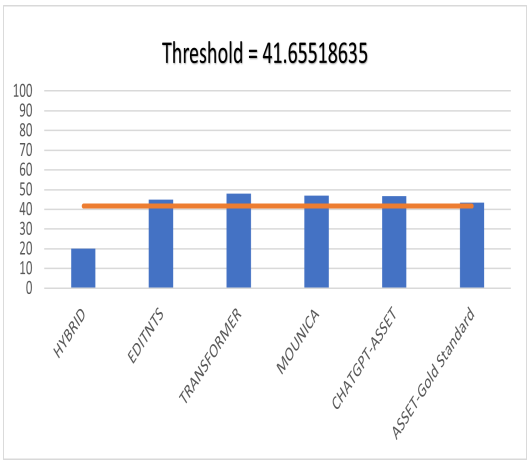
Table 4.10 Sentence Simplification metrics' values- Four References.



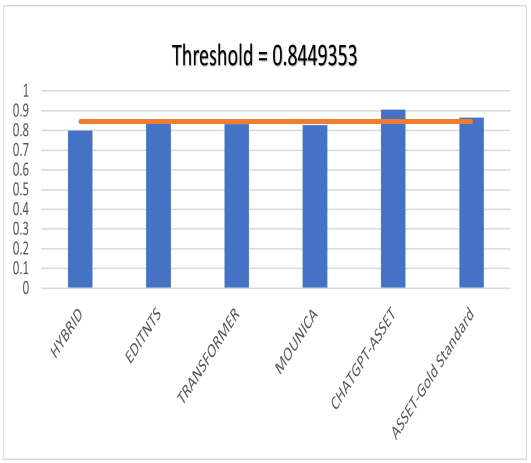
(a) BLEU



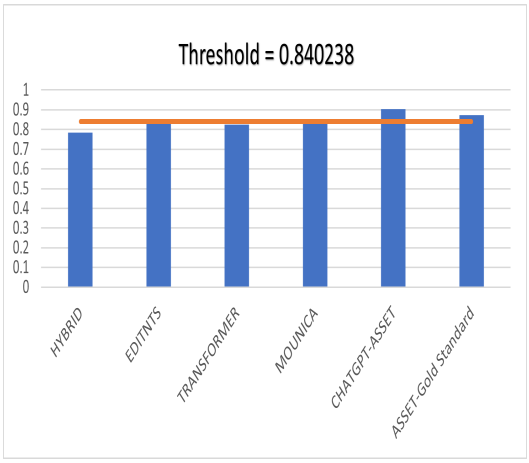
(b) SARI



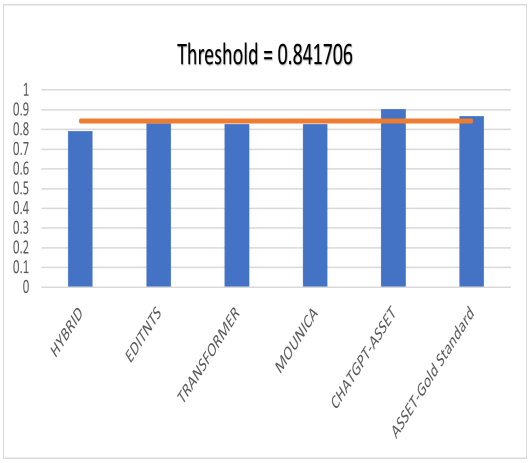
(c) LENS



(d) BERTScore$_{Precision}$



(e) BERTScore$_{Recall}$



(f) BERTScore$_{F1}$

Figure 4.3 Metrics' Thresholds across sentence simplification models and ASSET-gold standard.

Simplified sentence source	BLEU	SARI	LENS	BERTScore$_P$	BERTScore$_R$	BERTScore$_{F1}$
CHATGPT-ASSET	0.65673	0.49071	69.13186	0.935653	0.942347	0.936411
ASSET-Gold Standard	0.70128	0.44556	63.53623	0.929472	0.929862	0.925449

Table 4.11 Sentence Simplification metrics' values- Nine References.

4.2.3 Discussion

First of all, Table 4.9 presents reference-based metrics using a single reference. Upon close examination of the values of each metric across various simplification models, it is obvious that they fall within the same range. Computing the average of these values can construct a robust trusted threshold. This threshold will be defined as a standard in future comparisons with other simplification models.

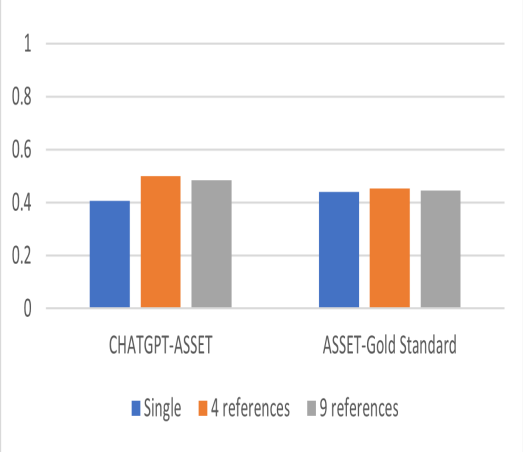
Furthermore, based on the information provided in Tables 4.10 and 4.11, it can be inferred that the increase in the number of references has a positive impact on the metrics' values. A positive correlation defined between the total number of references and the reference-based metrics. Except for SARI, the values demonstrated an increase with the four references, and subsequently, they remained stable or slightly decreased even with a further increase in the number of references.

Further analysis was conducted to understand the stabilization of the SARI values. SARI is the arithmetic average of three distinct formulas: that calculate the addition operations F score (SARI-add), keep operations F score (SARI-keep) and deletion operations Precision (SARI-del) [30]. Figure 4.5 shows the three components' values for SARI related outputs with respect to different number of values. It is notable, that there is an increase in the values with the four references compared to single reference. However, a slight decrease is clear in SARI-add and SARI-keep values with the nine references compared to the four references and this goes back to the recall term in the F score equation for each operation as noted in Section 2.3.4.

To understand this, we need to break down the key factors influencing the metrics: SARI-add measures how well the simplified sentence adds new content that aligns with the references. SARI-keep evaluates how effectively the simplified sentence retains relevant content from the original sentence. Both metrics are computed using the F-score, which combines precision (how much of the predicted content matches the references) and recall (how much of the reference content is captured by the predicted content).

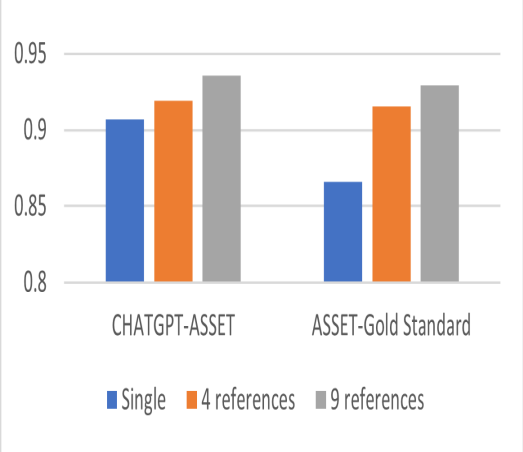
When the number of references increases, the model has a broader set of reference sentences to match against. This tends to improve recall, as there are more opportunities to overlap with any of the references, leading to higher metric values initially. However, with a very large number of references (e.g., nine), the diversity among references also increases.

(a) BLEU

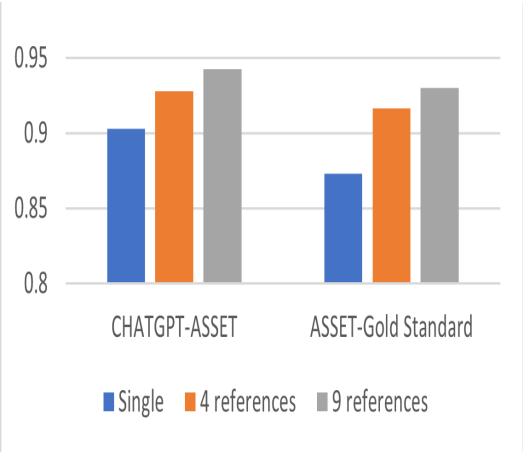


(b) SARI

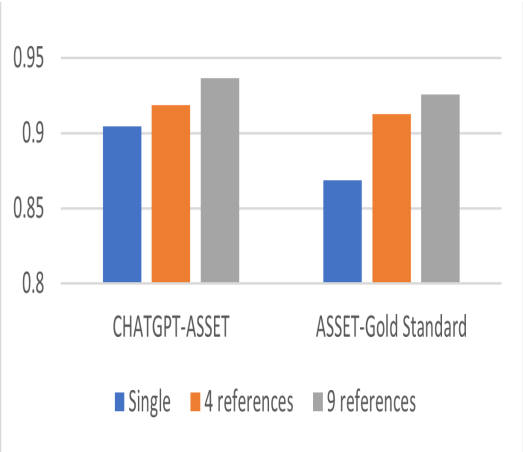
(c) LENS



(d) BERTScore$_{Precision}$



(e) BERTScore$_{Recall}$



(f) BERTScore$_{F1}$

Figure 4.4 The behavior of metrics with respect to the increasing number of references.

Not all references may fully align in terms of content additions or retained phrases. This can dilute the match between the predicted sentence and the references, potentially lowering recall. An example is shown in Table 4.12.

In the F-score formula, recall decreases when the predicted sentence fails to match the expanded diversity of reference sentences. Even if the precision (alignment with the most accurate references) remains high, the drop in recall due to mismatches with less aligned references will result in a slightly lower F-score for SARI-add and SARI-keep.

original	One side of the armed conflicts is composed mainly of the Sudanese military and the Janjaweed, a Sudanese militia group recruited mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan.
ChatGPT	The Sudanese military and the Janjaweed, made up of Afro-Arab Abbala tribes, are mostly on one side of the conflict.
References	On one side of the conflicts are the Sudanese military and the Janjaweed, a Sudanese militia group. They are mostly recruited from the Afro-Arab Abbala tribes.
	One side of the armed conflicts is composed mainly of the Sudanese military and the Janjaweed. The Janjaweed are a Sudanese militia group recruited mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan.
	One side of the armed conflicts is mainly the Sudanese military and the Janjaweed militia group.
	One side of the war is made up of the Sudanese military and the Janjaweed, a Sudanese militia group from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan.
	One side of the war is mainly made up of the Sudanese military and the Janjaweed. The Janjaweed is a Sudanese militia group who come mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan.
	One side of the fighters is made up of the Sudanese military and the Janjaweed. The Janjaweed is a Sudanese militia group. It is recruited mostly from the Afro-Arab Abbala tribes located in the northern Rizeigat region of Sudan.
	One side of the armed conflict includes the Sudanese military and the Janjaweed, a Sudanese militia group made up of Afro-Arab Abbala tribesmen.
	One side of the armed conflicts is composed mainly of the Sudanese military and the Janjaweed. The Janjaweed are a Sudanese militia group, with recruits from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan.
	One side of the war is mainly the Sudanese military and a Sudanese milita group called the Janjaweed. The Janjaweed is mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan.

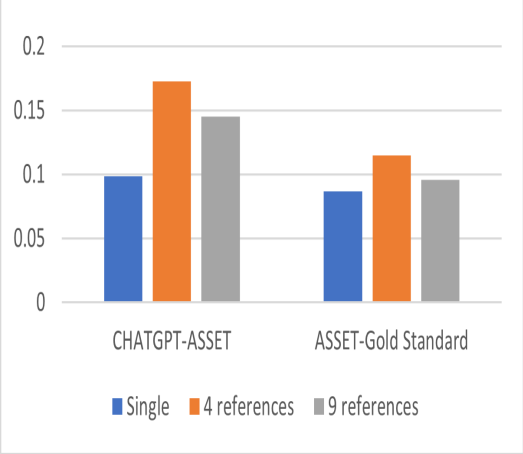
Table 4.12 Example of sentence simplification generated by ChatGPT, the original ASSET sentence and nine reference simplifications.

Unlike SARI-add and SARI-keep, SARI-del relies on precision rather than recall. Precision is less sensitive to the increasing number of references, so its values remain stable or are less affected by additional references. In summary, the decrease in SARI-add and SARI-keep with nine references compared to four is primarily due to the recall term in the F-score. This decline reflects the challenge of aligning predictions with a larger and more diverse set of references, where mismatches become more probable despite the model's overall quality.

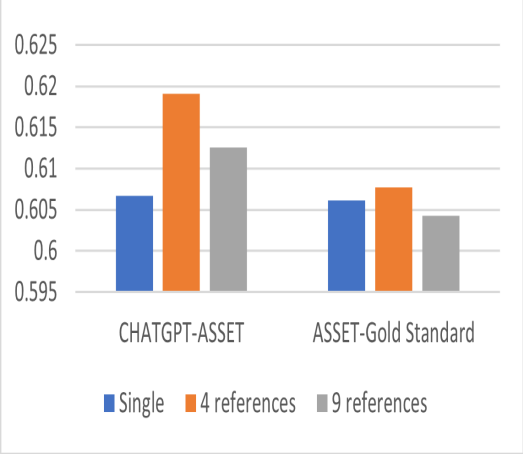
4.3 Summary

In this chapter, the degree of which reference-based automatic evaluation metric can measure the quality of a sentence simplification model was studied across multiple simplification models.

Our meta-evaluation study concludes that LENS is one of the best reference-based metrics to use with sentence simplification evaluation. It was able to measure the three different aspects: Adequacy, Fluency and Simplicity. The study also recommends using BERTScore$_{precision}$, BERTScore$_{Recall}$ and BERTScore$_{F1}$ to measure simplified sentence Fluency. Furthermore, BERTScore$_{Recall}$ can be applied to measure the meaning preservation



(a) SARI-Add



(b) SARI-Keep

(c) SARI-Delete

Figure 4.5 SARI further analysis with respect to increasing number of references.

of the simplified sentence compared to the complex input sentence. Finally, although SARI and BLEU did not show strong correlation with any of the aspects, we still recommend reporting them in future research to help comparing them with previous published state-of-the-art models.

Moreover, a threshold was established for each of the recommended metrics, simplifying the evaluation process for future developments in simplification models by benchmarking against state-of-the-art models.

At the end, the impact of employing multiple references instead of a single reference in computing the values of reference-based metric was studied. The study demonstrated that higher number of references will increase the metric values, except for SARI.

Chapter 5

Sequence-to-Sequence Models for Sentence Simplification

Until recently, prior to the revolutionary impact of Transformer architectures [7] and pre-trained Large Language Models (LLMs) such as BERT [49], the predominant approach to addressing the Text Simplification problem involved the use of Sequence-to-Sequence (Seq2Seq) models with encoder and decoder components. Most researchers utilized Long Short-Term Memory (LSTM) units, which are sophisticated recurrent units that implement a gating mechanism [80]. This raises a pertinent question: why was there an emphasis on LSTM rather than on another recurrent unit variant, the Gated Recurrent Unit (GRU) [81]? To address this, we conducted a comparative study on the performance of these two different units in the task of Sentence Simplification.

The impact of these recurrent units were systematically assessed on key aspects of sentence simplification, including the quality of the generated simplified sentences, and computational efficiency. By incorporating multiple datasets for training, evaluation, and testing, our study offers a robust comparison that takes into account variations in dataset source, vocabulary size, and other relevant parameters.

Additionally, each Seq2Seq model includes an embedding layer as the first hidden layer of the network, which encodes the sequence into a vector representation. We explored the impact of different embedding layer approaches, whether learned as part of the model or through the utilization of a pre-trained embedding layer.

This comprehensive comparison aimed to provide a deeper understanding of the relative advantages and limitations of LSTM and GRU units in the context of Sentence Simplification, thereby informing future choices in model architecture and training strategies.

In 2017, the introduction of Transformers [7] revolutionized Seq2Seq models across a variety of Natural Language Processing (NLP) tasks, including Machine Translation,

Text Summarization, Sentence Splitting, and Sentence Fusion. The remarkable success of Transformers in these areas motivates us to explore the potential of leveraging pre-trained Large Language Model (LLM) checkpoints to enhance sentence simplification models through the Transformer architecture following the strategy proposed by Rothe et al. [88].

This exploration is driven by the superior capabilities of Transformers in capturing long-range dependencies and contextual information, which are critical for producing high-quality simplifications. By integrating pre-trained LLM checkpoints, such as BERT [49], into Transformer-based Seq2Seq models, we aim to investigate the extent to which these advanced architectures can improve the performance of simplification models. Specifically, we seek to determine whether the rich contextual embeddings provided by pre-trained LLMs can lead to more fluent, coherent, and semantically accurate simplified sentences.

Through this study, we aim to advance the state-of-the-art in sentence simplification by harnessing the strengths of Transformer architectures and pre-trained LLMs, ultimately contributing to more effective and accessible simplification models for diverse applications.

In this chapter, we evaluate and compare the performance of LSTM and GRU units in the context of the sentence simplification task, incorporating various considerations and modifications. Based on our experiments, we conclude that LSTM units outperform GRU units within an encoder-decoder architecture with an attention-based mechanism. LSTM units generate higher quality simplified sentences according to the evaluation metrics and demonstrate superior performance in terms of computational efficiency.

On the other hand, the effectiveness of leveraging pre-trained Large Language Model (LLM) checkpoints was demonstrated through fine-tuning on different dataset sources and exploring various pre-trained LLMs. An in-depth analysis revealed that the best performance, according to the evaluation metrics, was achieved by applying bert-base-cased checkpoints with training on the Wikipedia dataset. Additionally, minor issues in encoder-decoder models with attention mechanisms were successfully addressed by employing warm-started Transformer encoder-decoder models.

5.1 Motivation

The significance of simplified sentences in enhancing human comprehension across diverse domains, such as educational materials, news articles, instructional content, medical records, physician notes, scholarly articles, and books, has driven us to develop robust systems capable of producing simplified sentences that meet specific criteria. Simplified sentences must ensure simplicity in both lexical and syntactic structures, maintain grammatical accuracy, and preserve the original meaning. This chapter focuses on Sequence-to-Sequence (Seq2Seq)

models, which have demonstrated remarkable performance in a wide range of text generation tasks. The primary objective is to evaluate their effectiveness in addressing the general English Sentence Simplification task. This evaluation includes a comprehensive analysis of their potential, as well as their limitations, providing valuable insights into their applicability and performance in simplifying English sentences.

5.2 Data

In the context of working with Seq2Seq models, having a parallel aligned dataset is essential. Focusing on end-to-end simplification of general English sentences limited our data selection options to Newsela and Wikipedia. Based on previous research studies introduced in Chapter-2 (Section 2.2.1), the best-aligned versions of these two corpora at the sentence level are as follows:

- WIKILarge [48]
- WIKI-AUTO [50]
- ASSET [51]
- NEWSELA-AUTO [50]

Sentence-Aligned Dataset	**Original Sentences**			**No. References**
	Training	**Validation**	**Testing**	
WIKILarge	296402			1
WIKI-AUTO	488,332			1
ASSET		2000	359	10
NEWSELA-AUTO	394,300	43,317	44,067	1

Table 5.1 General statistics of the datasets

Experiments were conducted using four distinct simplification datasets to evaluate the performance of sentence simplification models. Table 4.2 summarizes key statistics for each dataset, including the number of training, validation, and testing sentences, as well as the number of references aligned with each set. The study explored various dataset combinations, examining factors such as dataset size, source diversity, vocabulary size, and the number of references, to assess their influence on model performance. Training on data from diverse sources introduces a broader range of variability in input-output patterns, equipping the model to better handle complex and unseen examples across different domains. When datasets

from various resources are combined, the model gains exposure to a wide array of sentence structures, vocabulary, and linguistic styles. This variability enhances the model's ability to generalize effectively, reducing the likelihood of overfitting to the specific lexical, syntactic, or stylistic biases of a single dataset. In terms of lexical simplification, diverse datasets can provide a rich vocabulary, exposing the model to different synonyms, word substitutions, and usage contexts. The model learns to adapt its output based on the context of the input sentence, making it better equipped to simplify sentences in a nuanced and contextually appropriate manner. For syntactic simplification, combining datasets allows the model to encounter various sentence structures, ranging from straightforward declarative forms to complex constructions with multiple clauses. Exposure to such a range helps the model learn how to break down and restructure complex sentences into more digestible forms, accommodating different levels of linguistic complexity. Additionally, integrating datasets from multiple domains increases the overall size of the training corpus, which is crucial for deep learning models. Larger datasets allow for more robust parameter optimization and improve the model's ability to capture subtle patterns in sentence simplification tasks. This diverse training also enhances adaptability, enabling the model to simplify sentences across various topics and styles, from scientific articles to casual conversation. By recombining data resources, sentence simplification models achieve greater coverage of linguistic diversity, improving their capability to handle a variety of lexical and syntactic simplification strategies. This holistic approach ensures that the model performs well, delivering outputs that are both accurate and contextually appropriate.

5.3 Evaluation Strategy

In all experiments, the models were evaluated using automatic evaluation metrics recommended in Chapter 4, with the aim of minimizing reliance on human judgment. BLEU and SARI scores were calculated using the EASSE library [86]. For BERTScore, the implementation employed the microsoft/deberta-xlarge-mnli model [68]. Lastly, the LENS metric was reported using RoBERTa with LENS (k=3), as suggested by the authors [73].

5.4 Encoder-Decoder Model with Attention-Based Mechanism

The encoder-decoder architecture proved to be powerful for Sequence-to-Sequence problems in the field of natural language processing including Text Simplification. The encoder

reads the complex sentence and summarizes the information in a context vector. This context vector aims to embed information for all input words to help the decoder makes accurate predictions. The decoder interprets the context vector obtained from the encoder and generates the simplified sentence, and each output is also taken into consideration for future predictions. The attention mechanism introduced a significant enhancement to the traditional encoder-decoder architecture, addressing its limitation in capturing contextual relationships within long and complex sentences. In conventional encoder-decoder models, the encoder compresses the entire input sequence into a single fixed-length vector, which the decoder uses to generate the output sequence. However, this fixed-length representation often struggles to retain all the necessary contextual information, especially for longer sentences [32].

The attention mechanism mitigates this issue by enabling the decoder to selectively focus on relevant parts of the input sentence during the generation process. At each time step, the decoder determines which segments of the input sequence require more emphasis, effectively distributing the contextual encoding responsibility across multiple encoder hidden states. This approach allows for more precise alignment between the input and output sequences [32].

Bahdanau et al. global attention mechanism was applied as a solution to enhance the traditional encoder-decoder framework. Rather than relying solely on the final hidden state of the encoder as a fixed representation of the input sequence, Bahdanau's attention dynamically computes a context vector c_i at each decoding step. This context vector is constructed as a weighted sum of all encoder hidden states h_j, where the weights—referred to as attention scores α_{ij}—indicate the relevance of each encoder hidden state h_j to the current decoder step as described in the following equations:

$$\begin{aligned} c_i &= \sum_{k=1}^{T_\infty} \alpha_{ij} h_j \\ \alpha_{ij} &= \frac{exp(e_{ij})}{\sum_{k=1}^{T_\infty} exp(e_{ik})} \\ e_{ij} &= a(s_{i-1}, h_j) \end{aligned} \tag{5.1}$$

The attention scores α_{ij} are computed based on the alignment between the decoder's current hidden state s_i and each encoder hidden state h_j, where a is feedforward neural network. This dynamic computation enables the model to adaptively extract and emphasize contextually important information, resulting in more accurate and coherent output sequences [32].

This section compares various types of recurrent units utilized in the attention based encoder-decoder architecture, with an emphasis on more sophisticated units that implement a gating mechanism, specifically the Long Short-Term Memory (LSTM) unit [80] and the Gated Recurrent Unit (GRU) [81]. It is well established in the field that the LSTM unit performs effectively in Text Simplification task, while the GRU unit has been comparatively underutilized in this context. Furthermore, a fair comparison of the capabilities of these two units is lacking in the existing literature. LSTMs and GRUs were created as the solution to short-term memory issues in long sequences. They incorporate internal mechanisms known as gates, which can regulate the flow of information. These gates can learn which data in a sequence is important to keep or forget. By doing that, it can pass relevant information down the long chain of sequences to make predictions.

The internal architecture of both units are different base on the operations within the cell, which allow them to keep or forget information as shown in Figure 5.1. LSTM has three different gates: Forget, Input, and Output. The Forget gate decides what is relevant information to keep from prior steps. The Input gate decides what information is relevant to add from the current step. The Output gate determines what the next hidden state should be. Also, it keeps track of two different states: the hidden state is used for predictions and the cell state acts as the memory of the network [80]. On the other side, GRU has only two gates: Update gate combines Forget and Input gate of an LSTM and Reset gate is used to decide how much past information to forget. It keeps track of only a single state to transfer information, which is the hidden state [81].

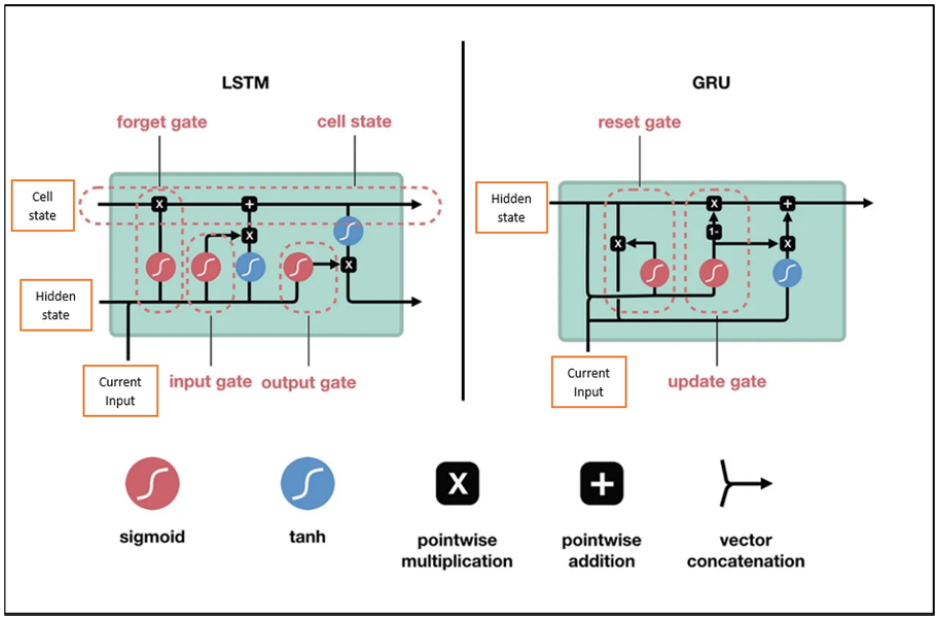


Figure 5.1 LSTM and GRU Architectures[6]

5.4.1 Model Architecture

As shown in Figure 5.2, the initial step in the Seq2Seq model involves transforming the input sentence into a suitable representation that can be fed into the model, which is achieved through an embedding layer. This layer converts each word in the input sentence into a dense vector of fixed size, capturing semantic properties of the words. Then, the encoder, a Recurrent Neural Network (RNN), takes the sequence of embedding vectors as input and processes them one at a time. On the other hand, the decoder, another RNN, generates the simplified sentence, initialized with the vector produced by the encoder. The decoder's objective is to generate the output sentence word by word. The attention layer derives a context vector that captures relevant source-side information to help predict the current target word. The decoder starts by feeding a simple input sentence into its embedding layer. The resulting vector, along with the context vector produce an attentional hidden state fed into the RNN, which is then processed by the RNN to generate the simplified output sentence [32].

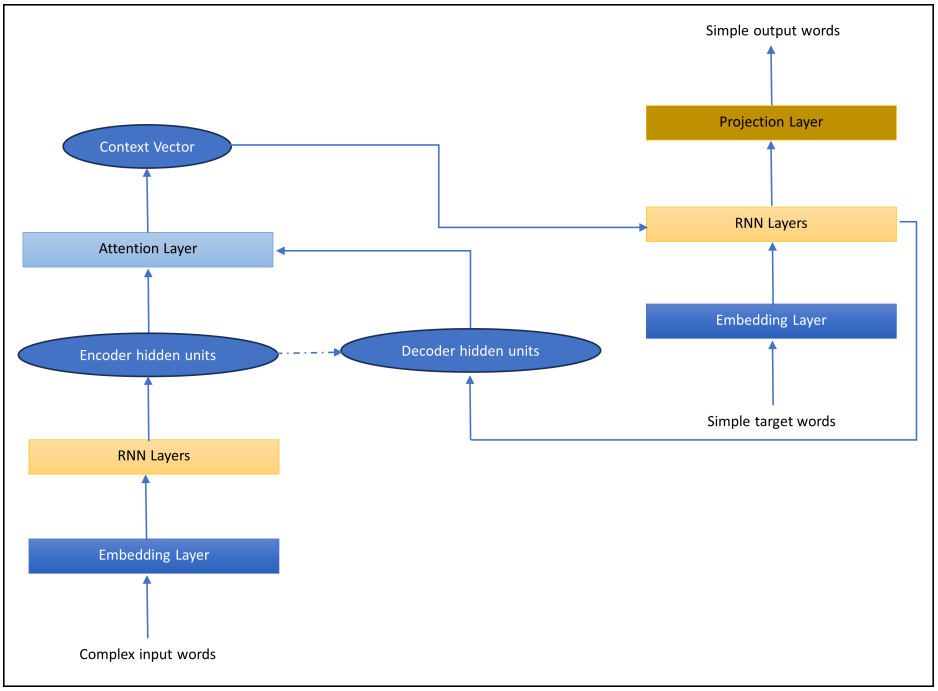


Figure 5.2 Encoder-Decoder with Attention-Based Architecture

5.4.2 Training and Testing Details

During training, the Seq2Seq model uses a technique called teacher forcing. Instead of feeding its own predictions back into the model, it feeds the actual target words from the training data. This approach enhances the model's learning effectiveness by preventing error

accumulation over time. However during testing, when the model is being used to simplify new sentences, the decoder feeds its own predictions back into the model, as the actual target words are not known. The model's hyperparameters were optimized through multiple runs, then fixed for the training phase, as outlined in Table 5.2. Except the number of epochs, various values were tested to assess their impact on the model's performance in terms of loss and outcomes.

Hyperparameter	**Value**
Optimization function	ADAM
Loss function	Sparse Categorical Cross Entropy
Number of RNN Nodes	512
Number of RNN Layers	2
Number of epochs	30
Batch Size	128
Seed	13

Table 5.2 Attention-based encoder-decoder hyperparameters

The number of RNN layers in Seq2Seq simplification models plays a critical role in determining the model's complexity, training dynamics, and output quality. Through extensive experimentation, we observed that models with single or double RNN layers often deliver comparable performance to those with deeper layer stacks. When combined with attention mechanisms, single or double-layer RNNs strike an effective balance between output quality, training stability, and computational efficiency. Increasing the number of layers introduces additional parameters, which, while potentially enhancing model capacity, can also increase the risk of overfitting—particularly when training on small or less diverse datasets. In such scenarios, shallower architectures, such as those with one or two layers, tend to generalize better, offering robust performance without the complexities and risks associated with deeper networks.

5.5 Gated Recurrent Unit (GRU)

The experiments were initiated by exploring Gated Recurrent Units (GRUs) within the attention-based encoder-decoder model for the task of Automatic English Sentence Simplification.

5.5.1 Focus on Embedding Layer

As previously discussed in the model architecture description (Section 5.4.1), both the encoder and decoder components of the model incorporate embedding layers. These embedding layers are crucial for transforming the input into dense vector representations, which can then be efficiently processed by the model. In our experiments, we explored two distinct embedding layer techniques, each evaluated under different data conditions. Specifically, we examined the performance of these techniques when the training, validation, and testing data were sourced either exclusively from a single source (Wikipedia or Newsela) or from multiple sources (both Wikipedia and Newsela). The embedding techniques investigated are outlined below.

Pre-Trained Embedding

For pre-trained embeddings, GloVe was applied [82], which generates vector representations for words through an unsupervised learning algorithm. GloVe was trained on extensive generic corpora, specifically Wikipedia 2014 and Gigaword 5, which together comprise six billion tokens and encompass a vocabulary of 400 thousand words.

An alternative to GloVe is Word2Vec [89], but GloVe offers multiple advantages. GloVe incorporates global statistics of the entire corpus during training whereas Word2Vec focuses on local context. This can lead to better overall word embeddings that capture global relationships between words. Additionally, GloVe tends to perform better with less frequent words since it aggregates statistics from the entire corpus, whereas Word2Vec sometimes struggles with infrequent words due to its reliance on local context.

Self-Trained Embedding

Here, a word embedding is learned as an integral part of the deep learning model based on the training dataset. Initially, the embedding layer is initialized with random weights and subsequently learns representations (embeddings) for all the words present in the training dataset.

5.5.2 Results

The results are divided into two sections. The first section presents the performance outcomes of the GRU encoder-decoder model with attention-based mechanism and pre-trained GloVe embeddings. The second section provides the results of the GRU encoder-decoder model with self-trained embeddings. Each section further distinguishes between two different data

set configurations: one where training, validation, and testing data originate from a single source (either Newsela or Wikipedia) and another where data are sourced from both datasets.

GRU Encoder-Decoder Model with Attention-Based and Pre-Trained GloVe Embeddings

- Single Data Source

  This subsection presents a sample of the simplification output in Table 5.3. The evaluation metrics presented in Table 5.4 when the model is trained, validated, and tested on data exclusively from either Newsela or Wikipedia. The results are further illustrated in Figure 5.3.

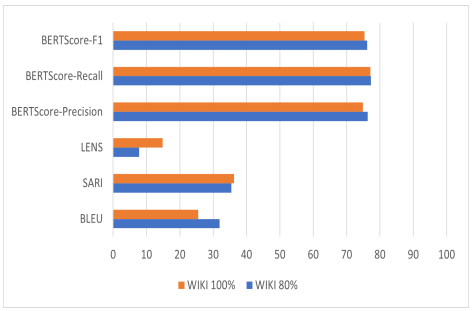
Original	One side of the armed conflicts is composed mainly of the Sudanese military and the Janjaweed, a Sudanese militia group recruited mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan. Jeddah is the principal gateway to Mecca, Islam's holiest city, which able-bodied Muslims are required to visit at least once in their lifetime. The Great Dark Spot is thought to represent a hole in the methane cloud deck of Neptune. His next work, Saturday, follows an especially eventful day in the life of a successful neurosurgeon. The tarantula, the trickster character, spun a black cord and, attaching it to the ball, crawled away fast to the east, pulling on the cord with all his strength.
GRU-GloVe	one side of the armed conflicts is composed mainly of the syrian army and the afro islands group of the northern balkans . the name is the main gateway to mecca which able relatively muslims visit . the great dark spot is thought to represent a hole in the methane cloud . his next year , he is also a good friends . the black mamba is a black hole with it fades into the back of the back with the back to stop the back to move the loss of the shoulder over the shoulder on the cord with all his tribemates .

Table 5.3 Sample outputs of GRU-GloVe automatic simplification

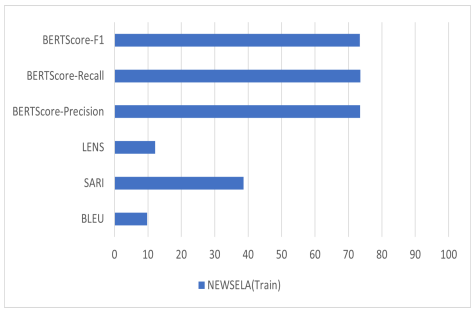
Training Set	**Validation Set**	**Testing Set**	**BLEU**	**SARI**	**LENS**	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
WIKILarge 80%	WIKILarge 20%	ASSET(test)	0.31894	0.35476	7.789961	0.763481	0.772514	0.761856
WIKILarge 100%	**ASSET(validate)**	**ASSET(test)**	**0.25486**	**0.36256**	**14.8572**	**0.748711**	**0.771618**	**0.753936**
NEWSELA-AUTO(train)	NEWSELA-AUTO(validate)	NEWSELA-AUTO(test)	0.09734	0.38657	12.1385	0.734593	0.735357	0.73377

Table 5.4 Results of attention-based GRU-GloVe model using a single data source

- Multiple Data Sources



(a) Testing set: ASSET



(b) Testing set: NEWSELA

Figure 5.3 Automatic evaluation results on the performance of GRU-GloVe across single data source

This subsection reports on the performance when the data for training, validation, and testing are drawn from a combination of both Newsela and Wikipedia. The metrics are presented in Table 5.5 and further illustrated in Figure 5.4.

Training Set	Validation Set	Testing Set	BLEU	SARI	LENS	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
WIKILarge 80%	WIKILarge 20%	NEWSELA-AUTO(test)	0.11638	0.35387	15.23949	0.72410649	0.75231707	0.7366057
WIKILarge 100%	NEWSELA-AUTO(validate)	ASSET(test)	0.26143	0.35448	7.262713	0.75250643	0.7652269	0.75249529
WIKILarge 100%	**NEWSELA-AUTO(validate)**	**NEWSELA-AUTO(test)**	**0.12714**	**0.3808**	**16.70841**	**0.74356693**	**0.75064921**	**0.74580282**
WIKILarge 100%	ASSET(validate)	NEWSELA-AUTO(test)	0.10962	0.35315	13.55825	0.72091508	0.74941158	0.73356771
NEWSELA-AUTO(train)	ASSET(validate)	ASSET(test)	0.08876	0.33406	2.980751	0.68186027	0.7205835	0.69320506
NEWSELA-AUTO(train)	ASSET(validate)	NEWSELA-AUTO(test)	0.10369	0.38648	12.1385	0.73849618	0.73353231	0.73490918
NEWSELA-AUTO(train)	NEWSELA-AUTO(validate)	ASSET(test)	0.15445	0.31529	4.30062	0.73486191	0.72599465	0.72532183

Table 5.5 Results of attention-based GRU-GloVe model using multiple data sources

GRU Encoder-Decoder Model with Attention-Based and Self-Trained Embeddings

- Single Data Source

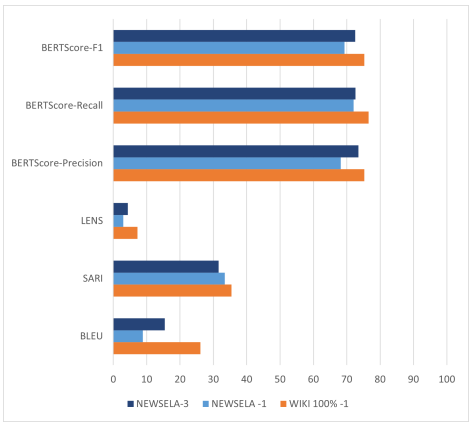
  This subsection outlines the performance metrics presented in Table 5.6 when the GRU model with self-trained embeddings is applied to data solely from Newsela or Wikipedia. The results are further illustrated in Figure 5.5.

Training Set	Validation Set	Testing Set	BLEU	SARI	LENS	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
WIKILarge 80%	WIKILarge 20%	ASSET(test)	0.15808	0.34177	3.907424	0.706851	0.730494	0.711719
WIKILarge 100%	ASSET(validate)	ASSET(test)	0.12354	0.34769	4.137886	0.686095	0.727079	0.698562
NEWSELA-AUTO(train)	**NEWSELA-AUTO(validate)**	**NEWSELA-AUTO(test)**	**0.07038**	**0.3764**	**13.54702**	**0.720085**	**0.719335**	**0.718535**

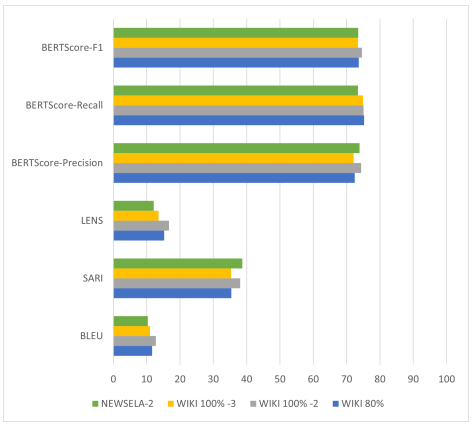
Table 5.6 Results of attention-based GRU self trained embedding model using a single data source

- Multiple Data Sources

  This subsection presents in Table 5.7 the outcomes evaluation results of the GRU model when trained, validated, and tested on a mixture of data from both Newsela and Wikipedia. The corresponding evaluation metrics are illustrated in Figure 5.6.

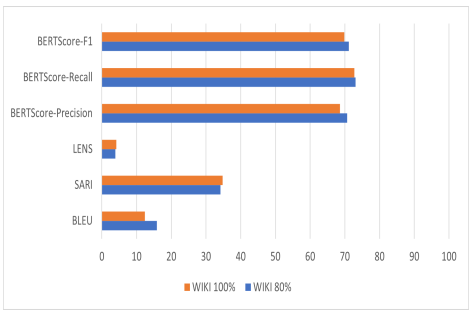


(a) Testing set: ASSET

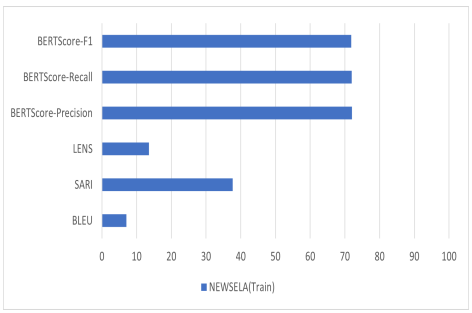


(b) Testing set: NEWSELA

Figure 5.4 Automatic evaluation results on the performance of GRU-GloVe across multiple data sources



(a) Testing set: ASSET



(b) Testing set: NEWSELA

Figure 5.5 Automatic evaluation results on the performance of GRU self trained embedding model across single data source

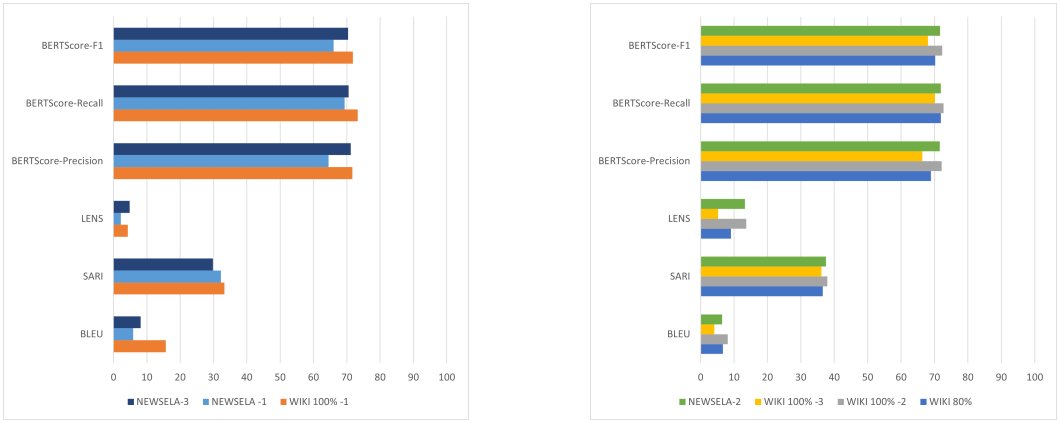
5.5.3 Discussion

The results from previous section provides a comparative analysis of how pre-trained versus self-trained embeddings influence the performance of the GRU model under different data sourcing conditions.

In terms of embedding techniques and based on the evaluation metrics presented in Tables 5.4, 5.5, 5.6 and 5.7, pre-trained embeddings consistently outperformed self-trained embeddings. Specifically, when examining Figures 5.3 and 5.5 for single data sources, and Figures 5.4 and 5.6 for multiple data sources, pre-trained embeddings demonstrated slightly superior performance across these datasets. This comparison ensures a fair assessment across different experimental conditions.

Training Set	Validation Set	Testing Set	BLEU	SARI	LENS	$BERT\,Score_{Precision}$	$BERT\,Score_{Recall}$	$BERT\,Score_{F1}$
WIKILarge 80%	WIKILarge 20%	NEWSELA-AUTO(test)	0.06683	0.36544	9.106654	0.68956888	0.71858799	0.70229125
WIKILarge 100% -1	NEWSELA-AUTO(validate)	ASSET(test)	0.15649	0.33296	4.294518	0.71626645	0.73239958	0.71784031
WIKILarge 100% -2	**NEWSELA-AUTO(validate)**	**NEWSELA-AUTO(test)**	**0.08127**	**0.37925**	**13.66435**	**0.72131795**	**0.72682267**	**0.72280794**
WIKILarge 100% -3	ASSET(validate)	NEWSELA-AUTO(test)	0.04093	0.3614	5.215497	0.66386652	0.70103198	0.68000609
NEWSELA-AUTO(train) -1	ASSET(validate)	ASSET(test)	0.05897	0.32205	2.193645	0.64498556	0.69316548	0.6606015
NEWSELA-AUTO(train) -2	ASSET(validate)	NEWSELA-AUTO(test)	0.06426	0.37544	13.28724	0.71581846	0.71922493	0.71621102
NEWSELA-AUTO(train) -3	NEWSELA-AUTO(validate)	ASSET(test)	0.08126	0.29781	4.863309	0.71199727	0.70569599	0.70413673

Table 5.7 Results of attention-based GRU self trained embedding model using multiple data sources



(a) Testing set: ASSET

(b) Testing set: NEWSELA

Figure 5.6 Automatic evaluation results on the performance of GRU self trained embedding model across multiple data sources

In terms of data source and based on the evaluation metrics, following the same comparison as before, our findings indicate that training on multiple data sources gives higher quality simplified sentences. Training on the Wikipedia dataset yields superior performance compared to Newsela. This observation is supported by the vocabulary sizes presented in Table 5.8 and illustrated in Figure 5.7, where WIKILarge exhibits a larger vocabulary compared to NEWSELA-AUTO, correlating with improved performance in the simplification model.

Training Set	Validation Set	Vocabulary Size
NEWSELA-AUTO(train)-1	NEWSELA-AUTO(validate)	35797
NEWSELA-AUTO(train)-2	ASSET(validate)	46812
WIKILarge 80%	WIKILarge 20%	131119
WIKILarge 100% -1	ASSET(validate)	139666
WIKILarge 100% -2	NEWSELA-AUTO(validate)	140575

Table 5.8 Vocabulary size based on training and validation datasets

In terms of computational efficiency, which encompasses the total time spent on training, validation, and testing cycles, as well as the model's loss, we observed that the two

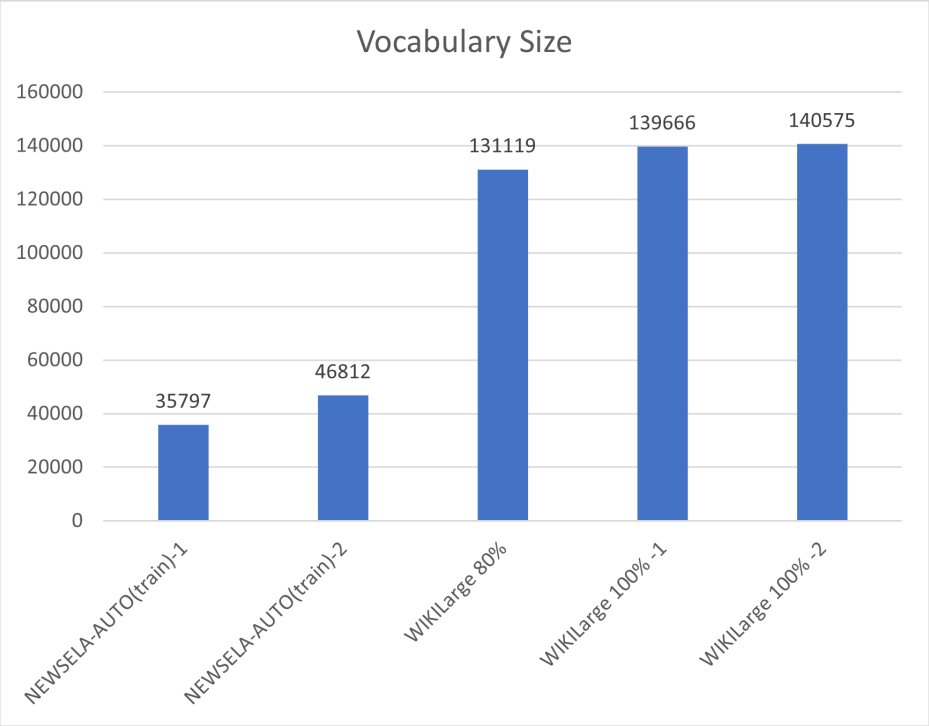


Figure 5.7 Distribution of vocabulary size across different training and validation datasets

embedding techniques demonstrated comparable performance in terms of time. However, a slight advantage in loss values was noted for self-trained embeddings. This result is anticipated, as self-trained embeddings are specifically optimized on the dataset used for the task, leading to better alignment with the model's objectives. These findings are illustrated in Figure 5.8.

5.6 Comparison of GRU and LSTM Architectures

It is challenging to definitively determine whether one type of gating units would consistently outperform the other. Although Bahdanau et al. (2015) reported that these two units performed comparably to each other in their initial experiments on machine translation, it remains uncertain whether this holds true for tasks beyond machine translation [32]. This motivates us to conduct more comprehensive empirical comparison between the LSTM unit and the GRU unit in encoder-decoder architecture with attention-based mechanism for the simplification task in this section.

Given the superior performance of pre-trained embeddings observed in previous GRU experiments, our subsequent investigations concentrated exclusively on the use of pre-trained embeddings for the LSTM models. This strategic decision was aimed at leveraging the

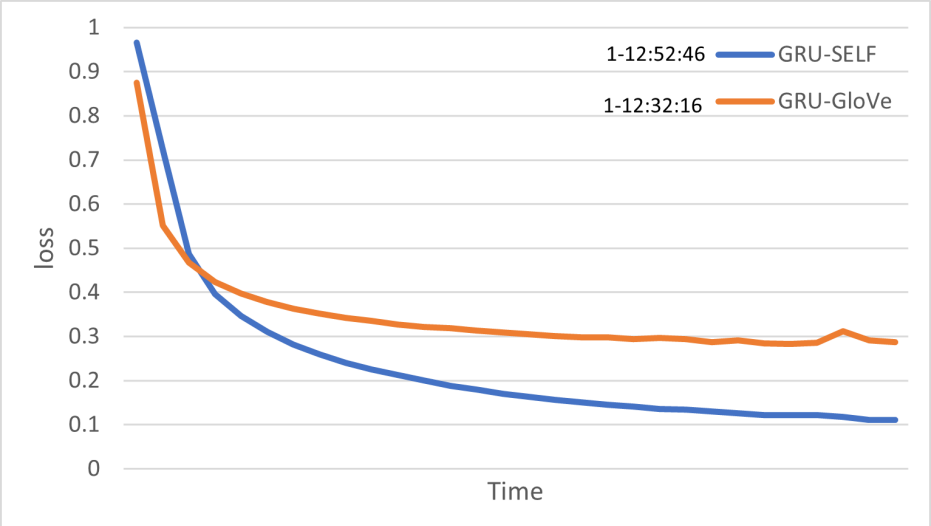


Figure 5.8 Efficiency computations of GRU pre-trained vs self-trained embeddings

demonstrated advantages of pre-trained embeddings, such as their ability to capture more comprehensive semantic relationships and enhance model performance. By utilizing pre-trained embeddings, we aimed to ensure the LSTM models benefit from the rich linguistic information embedded in these vectors, thereby potentially improving the accuracy and quality of sentence simplification outcomes.

5.6.1 Results

The results present the performance outcomes of the LSTM encoder-decoder model utilizing an attention-based mechanism and pre-trained GloVe embeddings.

Pre-Trained Embedding

The results of the LSTM encoder-decoder simplification model using the attention-based technique distinguish between two different data set configurations. The first configuration reports the outcomes when the model is applied to a single data source (either Newsela or Wikipedia), while the second details the performance metrics when the model is applied to a mixture of both data sources.

- Single Data Source

  This subsection lists sample outputs in Table 5.9 and discusses the performance metrics presented in Table 5.10 when the model is trained, validated, and tested on data exclusively from either Newsela or Wikipedia. The results are further illustrated in Figure 5.9.

Original	One side of the armed conflicts is composed mainly of the Sudanese military and the Janjaweed, a Sudanese militia group recruited mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan. Jeddah is the principal gateway to Mecca, Islam's holiest city, which able-bodied Muslims are required to visit at least once in their lifetime. The Great Dark Spot is thought to represent a hole in the methane cloud deck of Neptune. His next work, Saturday, follows an especially eventful day in the life of a successful neurosurgeon. The tarantula, the trickster character, spun a black cord and, attaching it to the ball, crawled away fast to the east, pulling on the cord with all his strength.
LSTM-GloVe	the side of the armed conflict is composed mainly of the sudanese military and the and the janjaweed . jeddah is the principal host to mecca , islam holiest city . the great dark spot is thought to represent a hole in the methane cloud deck of neptune . his next work , saturday , tells the most famous day . the tarantula , spun a black cord , attached it to the ball, and quickly crawled east, pulling hard .

Table 5.9 Sample outputs of LSTM-GloVe automatic simplification

Training Set	Validation Set	Testing Set	BLEU	SARI	LENS	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
WIKI 80%	**WIKI 20%**	**ASSET**	**0.4487**	**0.44043**	**13.8954**	**0.809911**	**0.810425**	**0.805666**
WIKI 100%	ASSET(validate)	ASSET	0.24664	0.36245	8.074024	0.746296	0.770372	0.751903
NEWSELA(Train)	NEWSELA(validate)	NEWSELA(test)	0.10515	0.3856	15.73424	0.735402	0.730998	0.732213

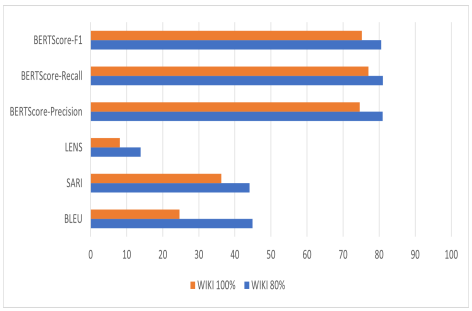
Table 5.10 Results of attention-based LSTM-GloVe embedding model using a single data source

- Multiple Data Sources

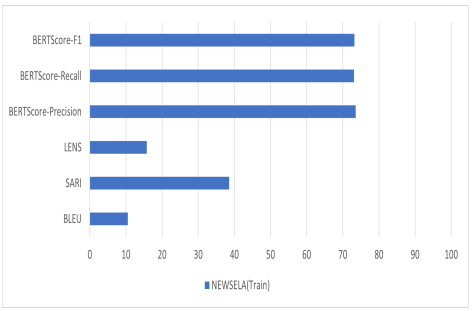
  This subsection presents in Table 5.11 the outputs' evaluation results of the LSTM model when trained, validated, and tested on a mixture of data from both Newsela and Wikipedia. The corresponding evaluation metrics are illustrated in Figure 5.10.

5.6.2 Discussion

This section explores the impact of using single versus multiple data sources on the performance of the LSTM encoder-decoder model with an attention mechanism. Additionally, it compares the performance of LSTM and GRU units in terms of the quality of simplified sentences, as assessed by evaluation metrics, and the computational efficiency of the models, measured by the time required for training, validation, and testing, as well as the improvement



(a) Testing set: ASSET



(b) Testing set: NEWSELA

Figure 5.9 Automatic evaluation results on the performance of LSTM-GloVe across single data source

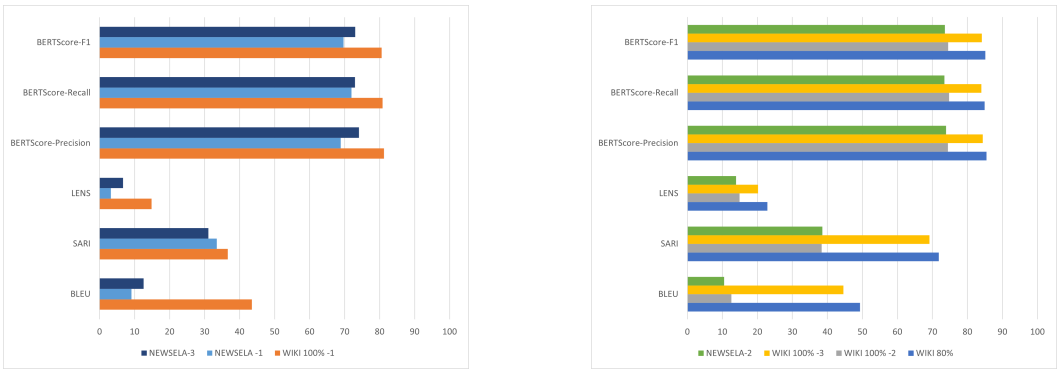
Training Set	Validation Set	Testing Set	BLEU	SARI	LENS	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
WIKI 80%	**WIKI 20%**	**NEWSELA(test)**	**0.49308**	**0.71812**	**22.89075**	**0.854407**	**0.848909**	**0.850965**
WIKI 100% -1	NEWSELA(validate)	ASSET(test)	0.43528	0.36656	14.8572	0.812787	0.808379	0.805653
WIKI 100% -2	NEWSELA(validate)	NEWSELA(test)	0.12.556	0.38383	14.89942	0.744123	0.747929	0.744883
WIKI 100% -3	ASSET(validate)	NEWSELA(test)	0.44552	0.69136	20.22036	0.843941	0.839569	0.841062
NEWSELA -1	ASSET(validate)	ASSET(test)	0.09089	0.33505	3.200119	0.688812	0.719869	0.696978
NEWSELA-2	ASSET(validate)	NEWSELA(test)	0.10515	0.3856	13.92677	0.738772	0.733958	0.735266
NEWSELA-3	NEWSELA(validate)	ASSET(test)	0.12562	0.31057	6.684691	0.741079	0.729767	0.730341

Table 5.11 Results of attention-based LSTM-GloVe embedding model using multiple data sources

in model loss.

In terms of data source and based on the evaluation metrics presented in Tables 5.10 and 5.11, our findings reveal that utilizing multiple data sources enhances the performance of our sentence simplification models compared to relying on a single data source. This suggests that a diverse dataset provides a richer variety of linguistic patterns and structures, which contributes to a more robust model training. Additionally, when comparing individual data sources, training on the Wikipedia dataset consistently yields superior outcomes compared to the Newsela dataset. This may be attributed to the larger vocabulary size and greater variety of sentence structures present in Wikipedia, as illustrated in Table 5.8 and Figure 5.7. Consequently, the Wikipedia dataset offers a more comprehensive and effective training corpus for the sentence simplification task.

In comparing the performance of GRU and LSTM units within encoder-decoder simplification models with attention mechanism, our findings indicate that LSTM units significantly outperformed GRU units based on the evaluation metrics. This provides a clear indication that LSTM-based models are better suited for achieving higher quality simplified sentences.



(a) Testing set: ASSET (b) Testing set: NEWSELA

Figure 5.10 Automatic evaluation results on the performance of LSTM-GloVe across multiple data sources

LSTM units outperform GRU units due to several key reasons rooted in their architecture and capability to handle sequential data effectively. LSTMs have a more complex gating mechanism, enables them to better manage and retain information over longer sequences compared to GRUs. Sentence simplification requires capturing context over long input sequences, especially when the input contains detailed or complex structures. LSTMs are better equipped to preserve relevant contextual information while simplifying sentences, leading to improved performance. Moreover, LSTMs allow for finer control over what information to keep, discard, or output at each time step. This granularity helps in simplification, where specific details must be retained while redundant or complex information is reduced. In contrast, GRUs combine some of these functions into fewer gates, which can make them less flexible in managing intricate dependencies. Sentence simplification involves understanding and restructuring syntax and semantics. LSTMs, with their ability to process and retain detailed information across layers, are better suited to maintain the balance between grammatical correctness, meaning preservation, and simplicity. In cases where the input sentences are long and intricate, GRUs may struggle to retain relevant information, leading to degraded performance. LSTMs are more robust in such scenarios due to their explicit memory cell mechanism that helps mitigate issues. The superior performance of LSTM units is evident from evaluation metrics, which reflect better fluency, adequacy, and simplicity in the output sentences. This aligns with their theoretical advantage in managing complex input-output mappings.

In terms of computational efficiency, LSTM units demonstrate slightly better performance in reducing the time required for training, validation, and testing cycles. Additionally, LSTM models achieve a lower loss value, indicating more effective optimization during the training process. Consequently, the slight edge in computational efficiency, combined with

superior model performance, suggests that LSTM units are more advantageous for tasks requiring high-quality simplified text. These findings are illustrated in Figure 5.11.

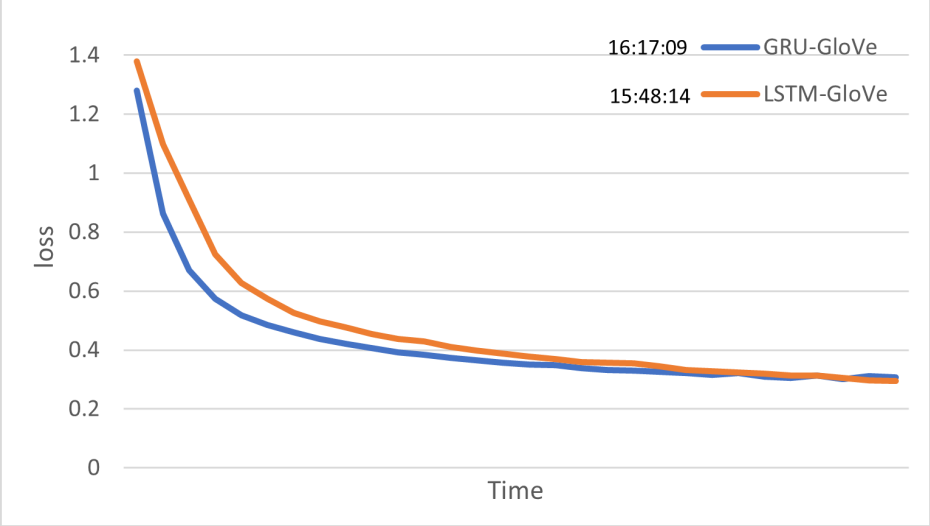


Figure 5.11 Efficiency computations of GRU vs LSTM

5.7 Leveraging Pre-trained Checkpoints for Seq2Seq Simplification Models

In 2017, Vaswani et al. revolutionized text generation tasks by shifting from encoder-decoder models based on complex recurrent neural networks with attention layers to the Transformer architecture. The Transformer relies entirely on attention mechanisms, replacing recurrent layers with multi-headed self-attention. In the Transformer model, the encoder maps an input sequence of symbol representations $(x_1, ..., x_n)$ to a sequence of continuous representations $z = (z_1, ..., z_n)$. Based on z, the decoder generates an output sequence $(y_1, ..., y_m)$ one element at a time, consuming previously generated symbols as additional input for generating the next one. The architecture of the Transformer model is shown in Figure 5.12 [7].

Recently, pre-trained Language Models (LMs) have had a positive impact on the field of natural language processing (NLP). These models are trained on large amounts of unlabelled data and subsequently fine-tuned on specific tasks, yielding superior results compared to training a randomly initialized model directly on those tasks. Following the introduction of the Transformer architecture, multiple transformer-based language models, known as pre-trained Large Language Models (LLMs), have been developed. These models are trained on massive datasets with substantial computational costs. Unsupervised and self-supervised

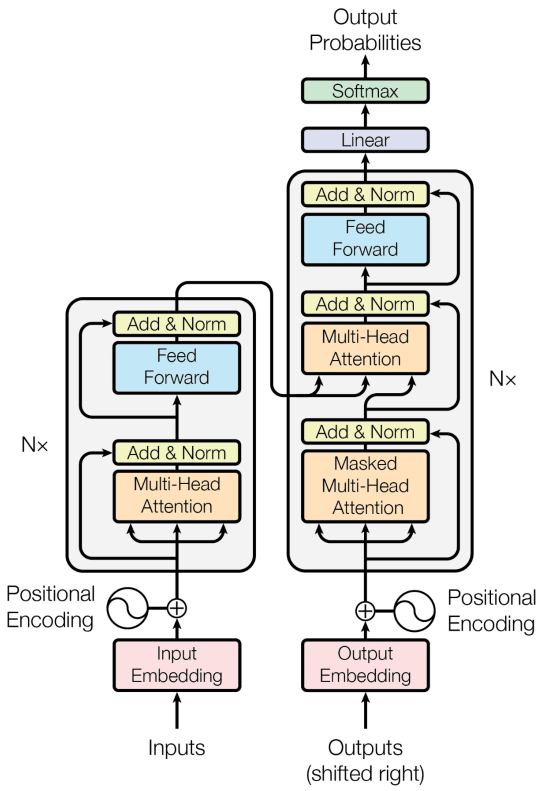


Figure 5.12 The Transformer model architecture [7]

pre-training methods have focused on either the Transformer's encoder part, as seen with BERT [49] and RoBERTa [72], or the decoder part, like GPT [90]. These approaches have established a qualitatively new level of baseline performance for many widely used Natural Language Understanding (NLU) benchmarks.

Similar to BERT and GPT, massive pre-trained encoder-decoder models on generic data have demonstrated significant performance improvements across a variety of Seq2Seq tasks following fine-tuning on task-specific datasets. One notable example is T5, developed by Raffel et al. in 2019 [44]. However, the substantial computational costs associated with pre-training these encoder-decoder models mean that their development is primarily undertaken by large companies and research institutions.

In 2020, Rothe et al. introduced a novel approach to leveraging pre-trained checkpoints for sequence generation tasks. They proposed initializing a Transformer encoder-decoder model with pre-trained encoder and/or decoder checkpoints, such as BERT and GPT. This methodology not only enhances performance but also significantly reduces training costs. The authors demonstrated that these warm-started Transformer encoder-decoder models achieve competitive results compared to large pre-trained encoder-decoder models like T5

across multiple Seq2Seq tasks. These tasks include Machine Translation, Text Summarization, Sentence Splitting, and Sentence Fusion. The models developed using this approach established new state-of-the-art results in these areas [88].

In our study, we adopt their approach to develop a high-quality Seq2Seq model for the task of sentence simplification. Specifically, we leverage the Transformer architecture, initializing it with pre-trained large language model (LLM) checkpoints. This initialization enables the model to benefit from the extensive linguistic knowledge and contextual understanding embedded within pre-trained LLMs, serving as a strong foundation for fine-tuning. Our primary goal is to achieve superior performance in sentence simplification by improving the quality of simplified outputs while optimizing the efficiency of the training process. This includes reducing training time and minimizing loss values, ensuring faster convergence and more accurate predictions. By combining the powerful capabilities of pre-trained models with fine-tuning on task-specific data, our approach seeks to balance computational efficiency with high-quality results, addressing both semantic preservation and simplicity in the output sentences.

5.7.1 Model Architecture

Our model architecture is a Transformer, illustrated in Figure 5.12, warm-started using pre-trained LLM checkpoints with minor modifications. Following the findings from Rothe et al. [88], we applied pre-trained encoder LLM checkpoints to both the encoder and decoder sides. Additionally, we experimented with sharing the weights between the encoder and the decoder.

For the encoder, we utilized the BERT Transformer layer implementations [49], which differ slightly from the regular Transformer layer [7]. Specifically, BERT employs a GELU activation function instead of the standard ReLU. The implementation of the decoder layers mirrors that of the BERT implementation but includes two key modifications. First the self-attention mechanism is masked to look only at the left context. Secondly an encoder-decoder attention mechanism was added to align with the canonical Transformer decoder [88]. These adjustments are crucial for adapting the pre-trained encoder LLM checkpoints to the Seq2Seq architecture.

5.7.2 Training and Testing Details

In our experiments, we investigated the performance of two distinct LLMs: BERT [49] and RoBERTa [72]. BERT and RoBERTa represent significant advancements in natural language

processing. These models are designed to handle a wide range of NLP tasks with high accuracy and efficiency, making them ideal candidates for our sentence simplification task.

BERT (Bidirectional Encoder Representations from Transformers) is an LLM recognized for its ability to pre-train deep bidirectional representations by jointly conditioning on both left and right context in all layers. This bidirectionality allows BERT to capture complex dependencies and relationships within the input text, making it particularly effective for various NLP tasks such as text classification, named entity recognition, question answering, and more. During pre-training, BERT is trained on large corpora of text from English Wikipedia and BookCorpus using two unsupervised tasks: masked language modeling (MLM) and next sentence prediction (NSP). The MLM task involves masking certain tokens in the input and predicting them based on the surrounding context, while the NSP task predicts whether two sentences follow each other in the original document [49].

RoBERTa (Robustly Optimized BERT Approach) is another prominent Large Language Model (LLM), which builds upon the BERT architecture with several enhancements aimed at improving pre-training and fine-tuning procedures. RoBERTa is trained on five English-language corpora of varying sizes and domains, totaling over 160GB of uncompressed text. The corpora are BookCorpus plus English WIKIPEDIA same as BERT, CC-NEWS, which is the English portion of the CommonCrawl News dataset, OPENWEBTEXT and STORIES. Unlike BERT, RoBERTa employs dynamic masking during pre-training, where tokens are randomly masked and replaced with new tokens each time they appear. This approach helps RoBERTa better leverage the training data and learn more robust representations. Additionally, RoBERTa benefits from larger batch sizes and training on more data compared to BERT, which further enhances its performance. The training corpus consists of several corpora gathered from web crawling and public corpora [72].

Each of these models is available in two versions: base and large. The base version contains fewer layers and parameters compared to the large version. Specifically, the base models have 12 layers, while the large versions have 24 layers, with corresponding increases in the number of attention heads and hidden units. This structural difference leads to a trade-off between computational efficiency and representational capacity. The base versions, while less complex, offer faster training and inference times, which is crucial when working with large datasets or when computational resources are limited.

We opted to use the base checkpoints for our experiments to strike a balance between leveraging the robust pre-trained representations of these LLMs and maintaining efficient computational resource usage. This choice was strategic, enabling us to conduct comprehensive experiments within a feasible timeframe and with available computational resources. By doing so, we were able to effectively harness the capabilities of BERT and RoBERTa in

simplifying sentences without the prohibitive costs associated with training and deploying the larger versions of these models. The specific hyperparameters employed in our experiments are detailed in Table 5.12.

Hyperparameter	Value
Optimization function	ADAM
Number of layers	12
Number of attention heads	12
Number of epochs	3
Batch Size	16
Seed	42

Table 5.12 Warm-started Transformer model hyperparameters

5.7.3 Investigated Model Variants and Pre-trained Checkpoints

Investigated Model Variants

In this section, we describe several combinations of model initialization.

- **BERT2BERT**

  A BERT-initialized encoder paired with a BERT-initialized decoder. All weights are initialized from public BERT-base [49] checkpoints. The only variable that is initialized randomly is the encoder-decoder attention [88].

- **RoBERTa2RoBERTa**

  Same as BERT2BERT, but the encoder and decoder are initialized with the public RoBERTa-base [72] checkpoint [88].

- **SharedRoBERTa**

  Like RoBERTa2RoBERTa, but the parameters between encoder and decoder are shared. This greatly reduces the memory of the model [88].

Pre-trained Checkpoints

Here we enumerate all the pre-trained LLM checkpoints that were explored in our experiments.

- **BERT-base Checkpoints**

We tokenize our text using the WordPiece [91] to match the BERT pre-trained vocabulary. Depending on the experiment, we use one of the following publicly available checkpoints: bert-base-cased, or bert-base-uncased [49]. Both checkpoints have a vocabulary size of approximately 30K wordpieces. Additionally, BERT trains positional embeddings for up to 512 positions, which is the maximum input and output length for our model. This approach ensures compatibility with the pre-trained BERT models and leverages their robust linguistic representations for our sentence simplification task [88].

- **RoBERTa-base Checkpoint**

  The vocabulary treatment in RoBERTa [72] is compatible with the SentencePiece tokenization [88]. We use the roberta-base model, which is case-sensitive and features a vocabulary size of 50K. The model inputs consist of sequences of up to 512 contiguous tokens.

5.7.4 Results

Impact of bert-base-uncased on Single and Multiple Data Sources

Here we outline the results of the experiments conducted utilizing the bert-base-uncased checkpoint. The results are categorized based on the origin of the training, validation, and testing datasets, whether sourced from a singular origin (Wikipedia or Newsela) or from multiple origins.

- Single Data Source

  This subsection presents examples in Table 5.13 and discusses the performance metrics outlined in Table 5.14, obtained by training, validating, and testing the model exclusively on data from either Newsela or Wikipedia using the bert-base-uncased LLM checkpoint. The results are further visualized in Figure 5.13 for a more comprehensive analysis.

- Multiple Data Sources

  This subsection presents in Table 5.15 the outcomes evaluation results of leveraging bert-base-uncased for Seq2Seq Simplification Models when trained, validated, and tested on a mixture of data from both Newsela and Wikipedia. The corresponding evaluation metrics are illustrated in Figure 5.14.

Original	One side of the armed conflicts is composed mainly of the Sudanese military and the Janjaweed, a Sudanese militia group recruited mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan. Jeddah is the principal gateway to Mecca, Islam's holiest city, which able-bodied Muslims are required to visit at least once in their lifetime. The Great Dark Spot is thought to represent a hole in the methane cloud deck of Neptune. His next work, Saturday, follows an especially eventful day in the life of a successful neurosurgeon. The tarantula, the trickster character, spun a black cord and, attaching it to the ball, crawled away fast to the east, pulling on the cord with all his strength.
BERT-base-uncased	one side of the armed conflicts is made up mainly of the sudanese military and the janjaweed, a militia group recruited mostly from the afro - arab abbala tribes of the northern rizeigat region in sudan . jeddah is the main gateway to mecca, islam's holiest city, which can be reached at least once in muslims lifetime . the great dark spot is thought to be a hole in the methane cloud deck of the planet neptune, and is believed to have been the source of a huge dark spot in the atmosphere of the earth's atmosphere, which is known as the " big dark spot ". his next work, saturday, is about an especially eventful day in the life of a successful neurosurgeon, and is considered to be one of the most important works of his time in the field . the tarantula, the trickster character, spun a black cord and, attaching it to the ball, pulled it away fast to the east, pulling on the cord with all his strength .

Table 5.13 Sample outputs of BERT-base-uncased automatic simplification

Impact of Various Pre-trained LLM Checkpoints

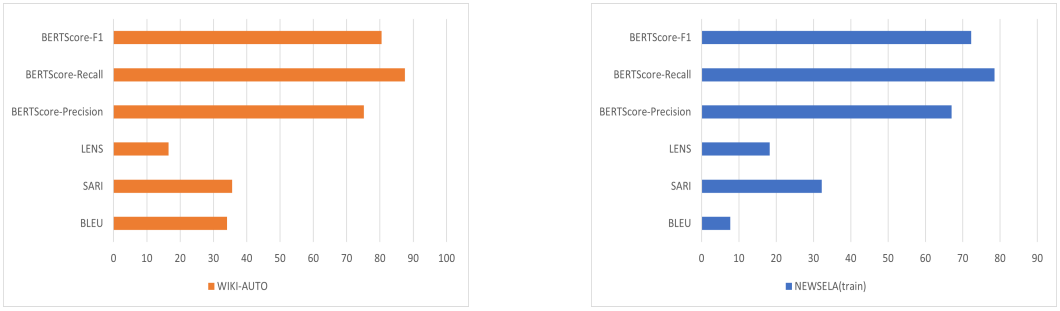
Table 5.16 shows a comprehensive list of all investigated model variants including google-bert/bert-base-uncased [49], google-bert/bert-base-cased [49], and FacebookAI/roberta-base [72], which is evaluated with both shared and unshared parameter configurations. The corresponding evaluation metrics are illustrated in Figure 5.15.

5.7.5 Discussion

This section explores the impact of single versus multiple data sources on the performance of the warm-started Transformer encoder-decoder models. Additionally, it examines the effects of various LLM checkpoints when leveraged in Transformer encoder-decoder models for the sentence simplification task.

Training Set	Validation Set	Testing Set	BLEU	SARI	LENS	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
NEWSELA(train)	NEWSELA(validate)	NEWSELA(test)	0.07677	0.32232	18.29021	0.66998136	0.78541464	0.7225166
WIKI-AUTO	**ASSET(validate)**	**ASSET(test)**	**0.34058**	**0.35646**	**16.52123**	**0.75159651**	**0.87499124**	**0.80471128**

Table 5.14 Results of BERT2BERT simplification model using single data source



(a) Testing set: ASSET (b) Testing set: NEWSELA

Figure 5.13 Automatic evaluation results on the performance of BERT2BERT simplification model across single data source

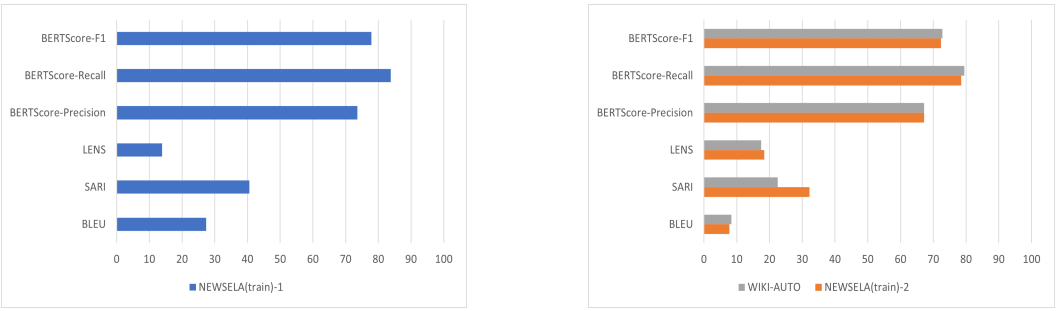
In terms of data source and based on the evaluation metrics presented in Tables 5.14 and 5.15, our analysis reveals that both single and multiple data sources yield comparable results. However, a single data source slightly enhances performance when the model is trained, validated, and tested on the same dataset. Additionally, when comparing individual data sources, training on the Wikipedia dataset improves performance compared to Newsela. This improvement can be attributed to Wikipedia's larger size and greater variety in syntactic and lexical simplifications, consistent with the findings from previous encoder-decoder models with attention (Section 5.4). However, with the current approach of leveraging pre-trained LLM checkpoints, the model already benefits from a robust vocabulary inherited from the initialization of these checkpoints. Consequently, fine-tuning the model on the simplification task using the Wikipedia dataset yields superior performance compared to the encoder-decoder model with attention mechanism.

In comparing the performance of various LLM checkpoints, it is clear from Figure 5.15 that the bert-base-cased checkpoint yielded the best performance among the different LLM checkpoints evaluated. However, RoBERTa-base and SharedRoBERTa-base also demonstrated comparable performance, with a slight improvement observed in SharedRoBERTa-base.

In terms of computational efficiency, the time required to complete the training, validation, and testing cycle correlates with the size of the dataset, specifically the number of sentences and vocabulary size. Regarding model loss, the best results were achieved by using

Training Set	Validation Set	Testing Set	BLEU	SARI	LENS	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
NEWSELA(train)-1	**ASSET(validate)**	**ASSET(test)**	**0.27378**	**0.40552**	**13.92087201**	**0.73565501**	**0.83835703**	**0.77910262**
NEWSELA(train)-2	ASSET(validate)	NEWSELA(test)	0.07779	0.3222	18.3874113	0.67161232	0.78563917	0.72357243
WIKI-AUTO	ASSET(validate)	NEWSELA(test)	0.08408	0.22478	17.47792744	0.67231917	0.79461068	0.72777599

Table 5.15 Results of BERT2BERT simplification model with multiple data sources



(a) Testing set: ASSET

(b) Testing set: NEWSELA

Figure 5.14 Automatic evaluation results on the performance of BERT2BERT simplification model across multiple data sources

multiple data sources with the bert-base checkpoints. These findings are illustrated in Figure 5.16.

5.8 Error Analysis

Through conducting multiple experiments and performing a detailed manual inspection of the results, several issues were identified in an effort to enhance the outcomes. These issues are discussed below:

5.8.1 Out-of-Vocabulary Words

In encoder-decoder models with attention mechanism, a phenomenon was observed where Out-of-Vocabulary (OOV) tokens were omitted during sentence simplification due to the presence of words that were unseen during training. This issue degrades the quality of the simplified sentences and requires resolution. These systems typically use a limited vocabulary, ranging from approximately 30K to 100K target words as shown in Table 5.8, which leads to the generation of OOV tokens. Despite the size of the vocabulary, there will almost always be words, such as proper names or numbers, that appear only in the development or test set and not during training.

Training Set	Validation Set	Testing Set	Pre-trained LLM	BLEU	SARI	LENS	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
WIKI-AUTO	ASSET(validate)	ASSET(test)	bert-base-uncased	0.34058	0.35646	16.52123	0.751597	0.874991	0.804711
WIKI-AUTO	**ASSET(validate)**	**ASSET(test)**	**bert-base-cased**	**0.34848**	**0.35673**	**17.58866**	**0.789508**	**0.920929**	**0.846388**
WIKI-AUTO	ASSET(validate)	ASSET(test)	RoBERTa-base	0.39523	0.33844	14.8572	0.786919	0.918846	0.844051
WIKI-AUTO	ASSET(validate)	ASSET(test)	SharedRoBERTa-base	0.41057	0.33188	14.8572	0.791969	0.9201	0.847757

Table 5.16 Results of leveraging multiple pre-trained LLM checkpoints for Seq2Seq simplification model

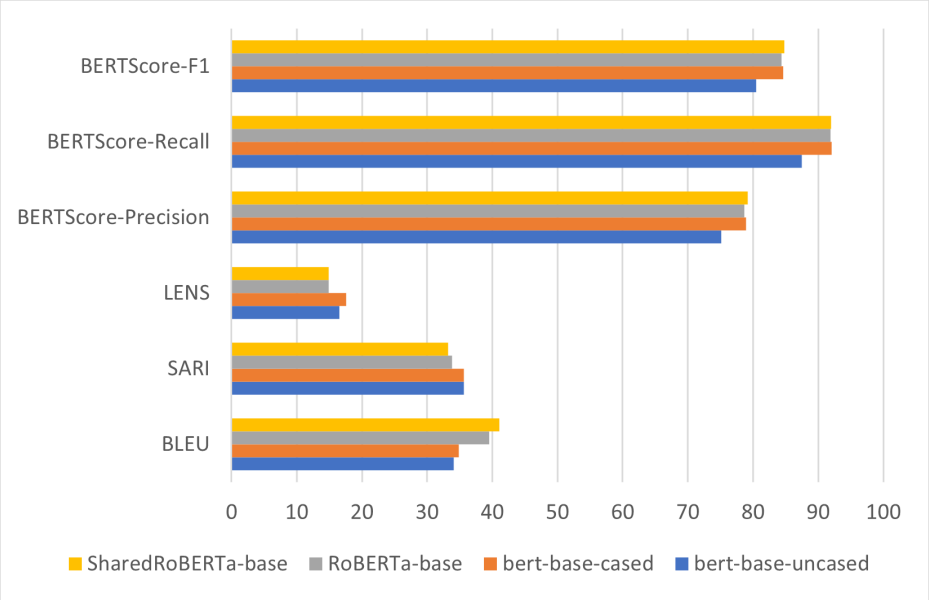


Figure 5.15 Automatic evaluation results on the performance of leveraging multiple pre-trained LLM checkpoints for Seq2Seq simplification model

However, with warm-started Transformer encoder-decoder models utilizing LLM checkpoints, the tokenizer is based on the WordPiece tokenizer [91]. WordPiece is a subword tokenization algorithm used in natural language processing (NLP) tasks. It breaks down words into smaller units called subword tokens, enabling machine learning models to handle OOV words more effectively and thereby improving performance on various NLP tasks.

We addressed the issue of Out-of-Vocabulary (OOV) tokens by replacing them in the simplified sentences with the corresponding words from the complex sentences to help preserve the original meaning. This was accomplished using a technique similar to those proposed by Luong et al. [92]. The OOV token problem was tackled by training the simplification model to track the origins of the unknown words in the simplified sentences. By identifying the source word responsible for each unknown target word, we introduced a post-processing step that copied the corresponding source word into the system's output for each OOV token. This method ensured that the meaning of the sentence remained intact, even when faced with OOV tokens.

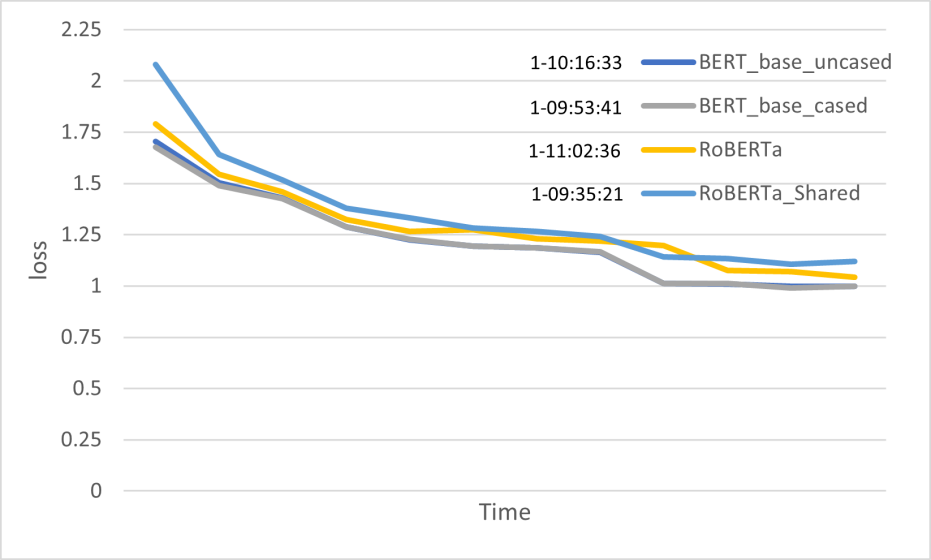


Figure 5.16 Computational efficiency of Leveraging LLMs for Seq2Seq Simplification Models

5.8.2 Text Degeneration

Another minor phenomenon observed in encoder-decoder models with attention mechanism is the repetition of words or phrases in the simplified sentences. This issue, known as text degeneration, is common in Seq2Seq models and has been analyzed in detail by Holtzman et al. [93]. The repetition can arise from the decoding process, where the model selects the most probable next word using argmax on the softmax output of the last layer, or when using beam search. Researchers have suggested that this phenomenon occurs because sequences with repetitions often have higher probabilities than sequences without repetitions. Another explanation is that the model might require additional training to effectively generalize across the diversity of the data.

To address this, we attempted to mitigate the issue by increasing the training epochs from 30 to 50 to observe any potential improvements in the output quality. However, these additional epochs did not yield improved results. Notably, this issue was not observed when using warm-started Transformer encoder-decoder models initialized with LLM checkpoints.

5.9 Qualitative Analysis

In addition to evaluating the simplified sentences using quantitative metrics, a qualitative analysis was conducted on the model outputs. To facilitate a comparative assessment of the

performance of different Seq2Seq models, the top 100 test samples were selected from the test dataset. The findings of this analysis are presented as follows:

- GRU Encoder-Decoder Model with Attention-Based Mechanism

  The model demonstrates a strong emphasis on lexical simplification, prioritizing deletion and substitution over syntactic transformations such as word reordering or sentence restructuring. As a result, the simplified sentences often omit essential contextual details, including geographical locations, proper nouns, or specific factual information. Moreover, several simplified outputs exhibit incomplete grammatical structures, which can compromise the fluency and coherence of the text. While the simplification process effectively reduces sentence length and complexity, potentially improving readability scores, excessive shortening may come at the expense of preserving important details and nuances. In some cases, the simplification leads to a loss of meaning, repetition, or awkward phrasing. Although the core ideas of the original sentences are generally retained, the reduction in complexity and vocabulary can impact the clarity and informativeness of the simplified text.

- LSTM Encoder-Decoder Model with Attention-Based Mechanism

  This model demonstrates improved lexical simplification and incorporates some degree of syntactic simplification compared to the previous model. However, the simplified sentences occasionally exhibit issues with sentence structure, grammatical accuracy, and fluency, which can hinder comprehension and reduce the overall readability of the output. Additionally, the simplification process often neglects key details or alters the original meaning, potentially shifting the focus of the sentence. These shortcomings highlight the need for a more balanced approach that preserves essential information and meaning while maintaining grammatical correctness and coherence.

- Transformer Initialized with Pre-trained LLM Checkpoint

  This model architecture achieves a balance between lexical and syntactic simplification, effectively addressing the complexities of input sentences. Complex sentences containing multiple subordinate or relative clauses are frequently simplified through reduction or elimination, resulting in more concise and accessible outputs. The simplified sentences successfully preserve the original meaning while enhancing readability and coherence. Lexical simplification is evident, with complex words and phrases replaced by simpler alternatives, while technical terms are retained and occasionally supplemented with explanations for clarity. Sentence splitting is also utilized, where complex sentences are broken down into multiple simpler sentences to improve understanding.

The qualitative analysis highlights that the simplification process involves reducing subordination, removing redundant or overly complex vocabulary, and restructuring sentences to achieve greater clarity and fluency. This approach effectively improves accessibility and readability while maintaining the integrity of the core meaning.

In conclusion, the Transformer model initialized with pre-trained LLM checkpoints emerges as the optimal choice for sentence simplification, particularly when addressing complex sentence structures and technical language. It consistently outperforms Seq2Seq Attention-based LSTM and GRU models by effectively preserving the original meaning and delivering fluent, coherent simplifications. While Seq2Seq Attention-based LSTM achieves commendable results, it faces challenges when handling highly complex or lengthy sentences. This limitation can lead to a minor loss of meaning or context in some instances, highlighting its comparative shortcomings against the Transformer model. On the other hand, Seq2Seq Attention-based GRU proves to be the least effective of the three approaches. It struggles with the nuances of highly complex sentences, often oversimplifying the input and neglecting critical details, which compromises the overall quality of the simplifications. These observations underscore the superiority of Transformer-based architectures for advanced simplification tasks.

5.10 Comparison with Related Work

In this section, we compare the performance of our top models, encompassing both the encoder-decoder with attention mechanism and the warm-started Transformer encoder-decoder with LLM checkpoints, across various configurations. These models are evaluated against the state-of-the-art models discussed in Chapter 2 (Section 2.4.2), as well as the performance threshold established in Chapter 4 (Section 4.2).

Table 5.17 presents the automatic evaluation metrics for our models alongside those from related research and the threshold. Metrics for DRESS [48] and Controllable-T5-base [45], were extracted directly from their respective research papers. In contrast, metrics for BERT-Initialized-Transformer [50] were computed as part of our work detailed in Chapter 4.

Analysis of the results depicted in Figure 5.17 indicates that our models, utilizing LSTM for encoder-decoder with attention mechanism and warm-started Transformer encoder-decoder with bert-base-cased, exhibit strong performance particularly in terms of the BERTScore metric. However, our models trail behind in LENS scores compared to BERT-Initialized Transformer and LENS established threshold. We hypothesize that increasing the learning rate warmup to 40K steps in warm-started Transformer encoder-decoder models, as implemented in the studies by Rothe at al. [88] and Jiang et al. [50], rather than our

current 2K steps, could potentially yield improved results and enhance the LENS score. Unfortunately, our ability to explore this hypothesis was constrained by the computational resources available on the Viking cluster.

Models	**BLEU**	**SARI**	**LENS**	$BERT\,Score_{Precision}$	$BERT\,Score_{Recall}$	$BERT\,Score_{F1}$
DRESS	0.2321	0.2737				
Controllable-T5-base	0.7121	0.4504				
BERT-Initialized-Transformer	0.331628	0.437544	48.04623	0.833351	0.823536	0.82751
OUR-BEST-GRU	0.12714	0.38080	16.70841	0.743567	0.750649	0.745803
OUR-BEST-LSTM	0.493080	0.718120	22.89075	0.854407	0.848909	0.850965
OUR-BEST-BERT2BERT	0.348480	0.356730	17.58866	0.789508	0.920929	0.846388
Threshold	0.360151	0.428693	41.65519	0.844935	0.840238	0.841706

Table 5.17 Automatic evaluation results comparing best of our models to related work and threshold

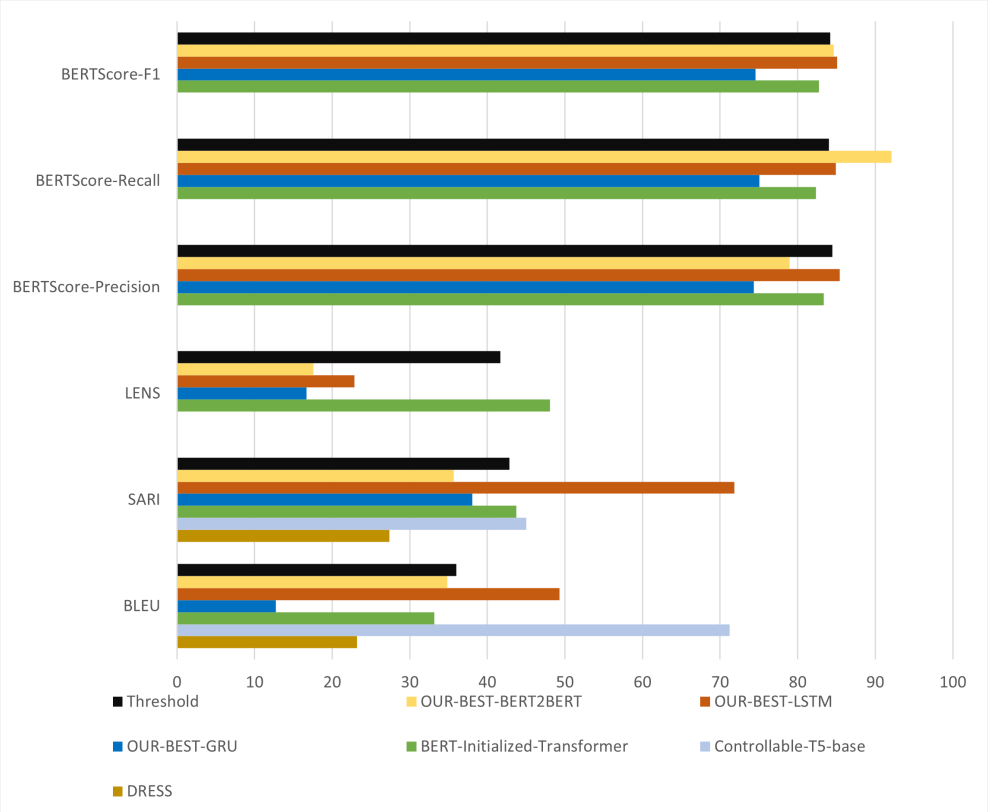


Figure 5.17 Evaluation metrics comparing best of our models to related work and threshold

5.11 Summary

This chapter covers the development of Seq2Seq models for sentence simplification task and answer the second, third and fourth research questions:

- **Second Question:** How does the performance of Gated Recurrent Units (GRUs) compare to Long Short-Term Memory (LSTM) units in Sequence-to-Sequence (Seq2Seq) sentence simplification models?

  We implemented encoder-decoder model with attention mechanism and compared the performance of GRU layer in both encoder and decoder sides with the performance using LSTM layer in both sides. The experiments cover training, validating and testing on single and multiple data sources.

  The results were as follows:

  In terms of data source, our findings indicate that for both GRU and LSTM units utilizing multiple data sources enhances the performance of our sentence simplification models compared to relying on a single data source. This suggests that a diverse dataset provides a richer variety of linguistic patterns and structures, which contributes to more robust model training. Additionally, when comparing individual data sources, training on the Wikipedia dataset consistently yields superior outcomes compared to the Newsela dataset.

  In comparing the performance of GRU and LSTM units within encoder-decoder simplification models with attention mechanism, our findings indicate that LSTM units significantly outperformed GRU units based on the evaluation metrics. This provides a clear indication that LSTM-based models are better suited for achieving higher quality simplified sentences.

  In terms of computational efficiency, LSTM units demonstrate slightly better performance in reducing the time required for training, validation, and testing cycles. Additionally, LSTM models achieve a lower loss value, indicating more effective optimization during the training process.

  Consequently, the slight edge in computational efficiency, combined with superior model performance, suggests that LSTM units are more advantageous for tasks requiring high-quality simplified sentences.

- **Third Question:** Does the choice of text embedding technique affect the performance of Sequence-to-Sequence (Seq2Seq) sentence simplification models, and if so, how?

To answer this question a comparative analysis of how pre-trained versus self-trained embeddings can influence the performance of the GRU model.

In terms of embedding techniques, our findings state that the quality of simplified sentences generated using encoder-decoder models with pre-trained embeddings measured by the evaluation metrics consistently outperformed those generated with self-trained embeddings.

In terms of computational efficiency, measured by the time spent on the entire cycle of training, validation, and testing, as well as the model's loss, we found that the two embedding techniques are comparable, with a slight advantage observed for pre-trained embeddings.

- **Fourth Question:** Can pre-trained Large Language Models (LLMs) be fine-tuned to simplify complex sentences?

  In this chapter, we successfully fine-tune Transformer encoder-decoder models utilizing Large Language Model (LLM) checkpoints for the simplification task. We investigate the effects of various LLM checkpoints, including BERT (cased and uncased) and RoBERTa, on model performance. Additionally, we examine the impact of sharing parameters between the encoder and decoder.

  In terms of data, our findings indicate that both single and multiple data sources yield comparable results. However, a single data source slightly enhances performance when the model is trained, validated, and tested on the same dataset. Additionally, when comparing individual data sources, training on the Wikipedia dataset improves performance compared to Newsela. This improvement can be attributed to Wikipedia's larger size and greater variety in syntactic and lexical simplifications, consistent with the findings from encoder-decoder models with attention mechanisms. However, with the current approach of leveraging pre-trained LLM checkpoints, the model already benefits from a robust vocabulary inherited from the initialization of these checkpoints. Consequently, fine-tuning the model on the simplification task using the Wikipedia dataset yields superior performance compared to the encoder-decoder model with attention mechanism.

  In comparing the performance of various LLM checkpoints, we found that the bert-base-cased checkpoint yielded the best performance among the different LLM checkpoints evaluated.

  In terms of computational efficiency, the time required to complete the training, validation, and testing cycle correlates with the size of the dataset, specifically the

number of sentences and vocabulary size. Regarding model loss, the best results were achieved by using multiple data sources with the bert-base checkpoints.

Warm-starting Transformer encoder-decoder models with pre-trained Large Language Model (LLM) checkpoints yields the highest quality simplified sentences. This success can be attributed to the rich semantic and syntactic knowledge embedded in the pre-trained LLMs combined with the powerful attention mechanisms inherent in the Transformer architecture. Nonetheless, extending the training duration is anticipated to further enhance the evaluation metrics, leading to even better model performance.

Chapter 6

Sentence Simplification for a Special Domain

In the previous chapter, we conducted an extensive examination of various Sequence-to-Sequence (Seq2Seq) model architectures aimed at simplifying general English sentences. The investigation revealed that employing pre-trained Large Language Model (LLM) checkpoints in Transformer encoder-decoder models leads to a substantial improvement in the quality of the simplified sentences. Specifically, these models excel in producing sentences that are not only lexically and syntactically simpler but also maintain grammatical correctness and accurately preserve the original meaning. This enhancement is attributed to the rich contextual understanding embedded within the LLM checkpoints, which allows the Transformer models to better handle the complexities of language simplification. Consequently, the resulting simplified sentences are more accessible while retaining the essence and integrity of the original text.

However, in everyday life, individuals encounter texts across a multitude of domains, ranging from books to newspaper reports, for various purposes such as acquiring knowledge, fulfilling work-related tasks, or engaging in academic studies. These texts can often be challenging to comprehend, either because they are written in a language that is not the reader's native tongue or because they pertain to a specialized domain outside the reader's area of expertise. Consequently, the need for domain-specific text simplification becomes evident. Therefore, in this chapter, we aim to address the problem of simplifying texts within specific domains by leveraging pre-trained LLM checkpoints for warm-starting Transformers, ensuring that they are more accessible and understandable to a wider audience. By focusing on domain-specific text simplification, we hope to bridge the comprehension gap for readers who might otherwise struggle with complex, specialized language, thereby enhancing their ability to engage with and benefit from the information presented.

In this chapter, we focus on the medical domain for two primary reasons. First, the significance of the medical field is paramount for individuals, communities, organizations, and countries, as it directly impacts public health and well-being. Second, our previous work demonstrated that leveraging pre-trained LLM checkpoints for Transformer encoder-decoder models, followed by fine-tuning on specific tasks, requires high-quality aligned parallel datasets. The availability of such datasets for the medical domain in English language at the sentence level provides a unique opportunity to explore and address the problem of domain-specific text simplification. This chapter aims to harness the potential of this advanced model to make complex medical texts more accessible and comprehensible to a broader audience, thereby facilitating better understanding and engagement with critical medical information.

The effectiveness of leveraging pre-trained LLM checkpoints was demonstrated through fine-tuning on a medical parallel dataset. We explored the performance of both a generic pre-trained LLM, namely BERT (both versions cased and uncased) [49], and a domain-specific LLM, BioBERT [83] and BioLinkBERT [94]. An in-depth analysis revealed that the highest performance, as indicated by the evaluation metrics, was achieved using the bert-base-cased checkpoint.

The two domain-specific LLMs are different in terms of their pre-training strategies, BioBERT is based on Mixed-Domain Pre-training, while BioLinkBERT is based on Domain-Specific Pre-training from Scratch. The choice of these LLMs was based on the Biomedical Language Understanding and Reasoning Benchmark (BLURB) [95].

BioBERT is a specialized variant of BERT (Bidirectional Encoder Representations from Transformers) tailored for biomedical text processing. BioBERT enhances the original BERT model by continual pre-training on large-scale biomedical text corpora. Specifically, this approach would initialize with the standard BERT model, pre-trained using Wikipedia and BookCorpus. It then continues the pre-training process with masked language modeling (MLM) and next sentence prediction (NSP) using biomedical text. The continual pre-training is conducted using PubMed abstracts and PubMed Central full text articles. This pre-training process helps BioBERT to acquire domain-specific language patterns, terminology, and concepts relevant to biomedical and clinical texts [83].

BioLinkBERT is a biomedical LLM trained from scratch on PubMed, which contains abstracts and citations of biomedical papers. Yasunaga et al. stated that academic papers have rich dependencies with each other via citations (references). They hypothesized that incorporating citation links can help LLMs learn dependencies between papers and knowledge that spans across them. The model was trained with two objectives. The first is the masked language modeling (MLM) objective, same as BERT, to encourage the LLM to learn multi-hop knowledge of concepts brought into the same context by document links. The

second objective is Document Relation Prediction (DPR), which classifies the relation of the second segment to the first segment (contiguous, random, or linked). DRP encourages learning the relevance and bridging concepts between documents, beyond the ability learned in the next sentence prediction objective in BERT. Pre-training from scratch derives the vocabulary and conducts pre-training using solely in-domain text [94].

6.1 Motivation

- **Why Medical Texts?**

  The ability to read and understand medical texts is a crucial component of public health. Unfortunately, many medical texts are difficult for the general population to comprehend, as they are typically targeted at highly-skilled professionals and use complex language and domain-specific terms. In this context, automatic text simplification, which aims to make such texts commonly understandable, would be highly beneficial.

  Medical text simplification holds significant potential for various applications, including the simplification of clinical reports to enhance patients' understanding of their medical conditions. This improved comprehension can empower patients to engage more effectively in their healthcare, better understand medical advice and information, and make more informed health decisions. Additionally, during health crises such as the COVID-19 pandemic, simplified medical texts can facilitate the community's understanding of reports issued by health organizations. Simplified language can make public health instructions more accessible and easier to follow, thereby promoting adherence to health guidelines and mitigating the spread of disease.

6.2 Data

In the context of working with Seq2Seq models, having a parallel aligned dataset is essential. For our experiments, we utilized the publicly available sentence-aligned dataset SELLS [54], which is described in detail in Chapter 2 (Section 2.2.2). Table 6.1 presents key statistics, including the number of training, validation, and testing sentences, as well as the number of references aligned with it.

Sentence-Aligned Dataset	Original Sentences			No. References
	Training	Validation	Testing	
SELLS	168241	42259	23416	1

Table 6.1 SELLS Dataset Statistics

6.3 Evaluation Strategy

In all experiments, similar to the previous chapter, the models were evaluated using the automatic evaluation metrics recommended in Chapter 4, with the objective of reducing dependence on human judgment and ensuring a consistent, objective assessment of model performance. BLEU and SARI scores were computed using the EASSE library [86]. For BERTScore, the evaluation utilized the microsoft/deberta-xlarge-mnli model [68]. Lastly, the LENS metric was calculated using RoBERTa [72] with LENS (k=3), following the approach suggested by the authors [73].

6.4 Methodology

In this chapter, our primary focus is on simplifying medical sentences through the fine-tuning of a Transformer model initialized with pre-trained Large Language Model (LLM) checkpoints. Our goal is to assess the quality of simplified sentences by comparing the use of domain-specific LLM checkpoints with that of generic LLM checkpoints. The model architecture employed adheres to the framework outlined in Chapter 5 (Section 5.7.1). Specifically, we performed a comparative performance evaluation of ‘bert-base-uncased‘ and ‘bert-base-cased‘, which are general-purpose LLMs, against ‘biobert-base-cased‘, a variant of ‘BERT-base-cased‘ with the same vocabulary but tailored for biomedical texts, and ‘BioLinkBERT-base‘, which is designed specifically for biomedical applications.

6.5 Training and Testing Details

In our experiments, we investigate the performance of two distinct LLMs: bert-base-uncased and bert-base-cased [49], which represent generic LLMs, and biobert-base-cased [83], and BioLinkBERT-base designed specifically for biomedical domain applications.

The choice of the base version of each model was strategic, enabling us to conduct comprehensive experiments within a feasible timeframe and with the available computational resources. This approach allowed us to effectively harness the capabilities of BERT, BioBERT and BioLinkBERT for simplifying medical sentences without incurring the prohibitive costs

associated with training and deploying the larger versions of these models. The specific hyperparameters employed in our experiments are the same as those mentioned in Chapter 5, Table 5.12.

6.6 Investigated Model Variants and Pre-trained Checkpoints

- **Investigated Model Variants**

  In this section, we describe the different combinations of initializing warm-started Transformer encoder-decoder models used in our experiments.

  - **BERT2BERT**

    A BERT-initialized encoder paired with a BERT-initialized decoder, with all weights initialized from publicly available BERT-base checkpoints [49]. The only component that is initialized randomly is the encoder-decoder attention [88].

  - **BioBERT2BioBERT**

    Same as BERT2BERT, but the encoder and decoder are initialized with the public biobert-base-cased checkpoint [83].

  - **BioLinkBERT2BioLinkBERT**

    Same as BERT2BERT, but the encoder and decoder are initialized with the public BioLinkBERT-base checkpoint [94].

- **Pre-trained Checkpoints**

  Here we enumerate all the pre-trained LLM checkpoints that were explored in our experiments.

  - **BERT-base Checkpoints**

    Similar to the approach detailed in Chapter 5 (Section 5.7.3), we tokenize our sentences using the WordPiece tokenizer [91] to match the BERT pre-trained vocabulary. Depending on the experiment, we use one of the following publicly available checkpoints: bert-base-cased, or bert-base-uncased.

  - **BioBERT-base-cased Checkpoint**

    For tokenization, we use the BioBERT tokenizer, specifically the WordPiece, which is compatible with BERT. BioBERT is case-sensitive and features a vocabulary size of 30k, the same as bert-base-cased. The model inputs consist of sequences of up to 512 contiguous tokens.

– **BioLinkBERT-base Checkpoint**

For tokenization, we use the BioLinkBERT tokenizer, which is also compatible with BERT. The model inputs consist of sequences of up to 512 contiguous tokens.

6.7 Results

Table 6.2 outlines the output of the experiments conducted utilizing the bert-base-uncased, the bert-base-cased, the biobert-base-cased-v1.1 and BioLinkBERT-base checkpoints. Along with the evaluation of the simplified sentences using evaluation metrics as shown in Table 6.3, which is further illustrated in Figure 6.1 .

Original	Endothelial cells (ECs) are highly specialized across vascular beds.
BERT-base-uncased	endothelial cells (ecs) are specialized cell types that are highly specialized in their ability to sense and respond to environmental cues, such as oxygen, nutrient availability, and oxygen availability in the bloodstream, and are involved in many physiological processes, including wound healing, inflammation, and cancer.
BERT-base-cased	Endothelial cells (ECs) are a group of cell types that are highly specialized in their ability to divide and differentiate into tissue - specific cells. " Endothelin (EC) is a specialized cell type that is highly specialized for vascular development and differentiation in vascular tissues.
biobert-base-cased	endothelial cells (ecs) are the most abundant cells in the blood vessels, but their functions are not well understood, but they are highly conserved across the blood vessel network, and their function is not well - understood, such as the maintenance of blood vessels in vivo.
BioLinkBERT-base	endothelial cells (ecs) are the most abundant cells in the blood vessels, and are important for vascular function and vascular function, but the molecular basis for this function is not well understood, especially in the vascular vasculature, in which endothelial cells are highly expressed in blood vessels and vascular endothelial cells.

Table 6.2 Sample simplifications from Transformers warm-started with pre-trained LLM checkpoints

6.8 Discussion

In our experiments we fine-tuned a Transformer model, initialized with LLM checkpoints, on the biomedical dataset SELLS. The results indicate that all the fine-tuned models initialized

Training Set	Validation Set	Testing Set	Pre-trained LLM	BLEU	SARI	LENS	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
SELLS	SELLS	SELLS	bert-base-uncased	0.03087	0.38753	20.11909	0.70034	0.73251	0.71569
SELLS	SELLS	SELLS	bert-base-cased	0.02116	0.40012	19.46215	0.70646	0.73082	0.71808
SELLS	SELLS	SELLS	biobert-base-cased-v1.1	0.02365	0.40148	18.04701	0.70171	0.72798	0.71426
SELLS	SELLS	SELLS	BioLinkBERT-base	0.01968	0.40278	17.70745	0.69474	0.72204	0.70777

Table 6.3 Results of leveraging general and biomedical pre-trained LLM checkpoints for medical domain sentence simplification model

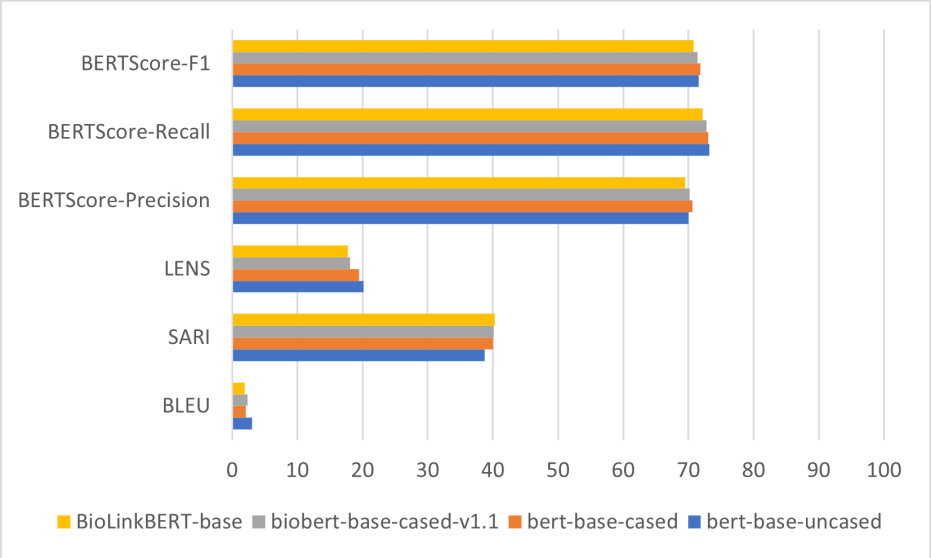


Figure 6.1 Automatic evaluation results on the performance of leveraging general and biomedical pre-trained LLM checkpoints for medical domain sentence simplification model

with various LLM checkpoints: bert-base-uncased, bert-base-cased, biobert-cased-v1.1 and BioLinkBERT-base produce comparable outcomes. However, it is important to note that while fine-tuning our model on the SELLS dataset, we set the total number of training epochs for BERT-base-uncased version to the default value of 3.0. In contrast, due to computational power limitations, we had to reduce the total number of training epoch for BERT-base-cased, BioBERT-base-cased and BioLinkBERT-base LLM checkpoints to 1.0.

The reduction in training epochs for these models was a necessary adjustment to accommodate the constraints of our available computational resources. Despite this limitation, the performance of the domain-specific LLM checkpoints BioBERT-base-cased and BioLinkBERT-base were the least compared to that of BERT as shown in Figure 6.1. Moreover, based on the experiments findings in Chapter 5, (Section 5.7.4, Figure 5.15) we hypothesize that if BERT-base-cased was trained for the same number of epochs as BERT-base-uncased, it would likely surpass BERT-base-uncased in terms of evaluation metrics. On the other hand, both Biomedical LLMs' design, which is specifically tailored for biomedical text is not better suited to simplify domain-specific nuances and terminology present in the

SELLS dataset. Therefore, with equal training conditions, BERT-base-cased delivered better Biomedical sentence simplification results.

Our conclusion highlights a nuanced finding in the application of large language models (LLMs) to sentence simplification within the biomedical domain. While one might intuitively expect that biomedical-specific LLMs would excel in simplifying domain-specific text, our analysis suggests otherwise. Biomedical-specific LLMs, although deeply trained on domain-relevant corpora, often prioritize accuracy and the inclusion of contextual details over simplifying language. This tendency can inadvertently lead to outputs that are more complex, undermining the primary goal of simplification. Simplification tasks, however, demand a balance between reducing complexity and retaining essential meaning. By strategically selecting and matching the dataset and LLM checkpoint to the task, we can significantly enhance the quality and effectiveness of the sentence simplification process.

6.9 Comparison with Related Work

In this section, we compare the performance of our models BERT2BERT-uncased, BERT2BERT-cased, BioBERT2BioBERT and BioLinkBERT2BioLinkBERT against the related models discussed in Chapter 2 (Section 2.4.3) alongside Zero-shot models reported by Joseph et al. [57], including the following:

- GPT-3 zero-shot

  Researchers evaluated zero-shot simplifications generated by the GPT-3-davinci-003 model [96] using prompts similar to the following: "My fifth grader asked me what this sentence means: [TEXT TO SIMPLIFY] I rephrased it for him in [LANG], in plain language a fifth grader can understand:"

- Flan-T5 zero-shot

  Also they evaluated zero-shot Flan-T5-base [79] performance. A simple prompt was used “Simplify this sentence: [TEXT TO SIMPLIFY]”.

Table 6.4 presents the automatic evaluation metrics for the related research as reported in their respective paper [57], alongside the results from our models, BERT2BERT-uncased, BERT2BERT-cased, BioBERT2BioBERT and BioLinkBERT2BioLinkBERT.

Figure 6.2 illustrates that the evaluation results from Joseph et al. [57] exhibit slightly superior performance compared to our models. This difference in performance may be attributed to the dataset characteristics. The MultiCochrane dataset [57] is derived from a single source, the Cochrane Library of Systematic Reviews, and is considerably smaller in

Models	**BLEU**	**SARI**	**LENS**	$BERTScore_{Precision}$	$BERTScore_{Recall}$	$BERTScore_{F1}$
GPT3-zero	0.0238	0.42111				0.8774
Flan-T5-zero	0.0812	0.39057				0.8810
mT5-unfiltered	0.0766	0.40219				0.8785
mT5-filtered	0.0882	0.39579				0.8843
Flan-T5-filtered	0.0870	0.39526				0.8875
BERT2BERT-uncased	0.03087	0.38753	20.11909	0.70034	0.73251	0.71569
BERT2BERT-cased	0.02116	0.40012	19.46215	0.70646	0.73082	0.71808
BioBERT2BioBERT	0.02365	0.40148	18.04701	0.70171	0.72798	0.71426
BioLinkBERT2BioLinkBERT	0.01968	0.40278	17.70745	0.69474	0.72204	0.70777

Table 6.4 Evaluation metrics comparing our models to related work

size compared to the SELLS dataset [54], as shown in Table 6.5. The SELLS dataset is sourced from 12 different journals, including Cochrane, and is more than twice the size of MultiCochrane. We hypothesize that training our BERT-base-cased model for more than one epoch and increasing the learning rate warmup to 40K steps, as implemented in the studies by Rothe at al. [88] and Jiang et al. [50], rather than our current 2K steps, could potentially lead to improved results, potentially closing the performance gap observed. The larger and more diverse SELLS dataset poses greater challenges in terms of variety and complexity, which could explain the current performance differences. Enhanced training with more epochs would likely allow our model to better capture and adapt to this diversity, thereby improving its evaluation metrics scores.

Sentence-Aligned Dataset	**Original Sentences**			**No. References**
	Training	**Validation**	**Testing**	
SELLS	61194	83	395	1

Table 6.5 Multicochrane Dataset Statistics

6.10 Summary

In this chapter, we address our **Fifth Research Question:** Does fine-tuning a domain-specific Large Language Model (LLM) on a domain- specific dataset lead to enhanced quality in simplified sentences? To explore this, we concentrated on the medical domain, employing the optimal model architecture identified in our experiments from Chapter 5 for the Seq2Seq sentence simplification task. This architecture involves leveraging LLM checkpoints for Transformer encoder-decoder models.

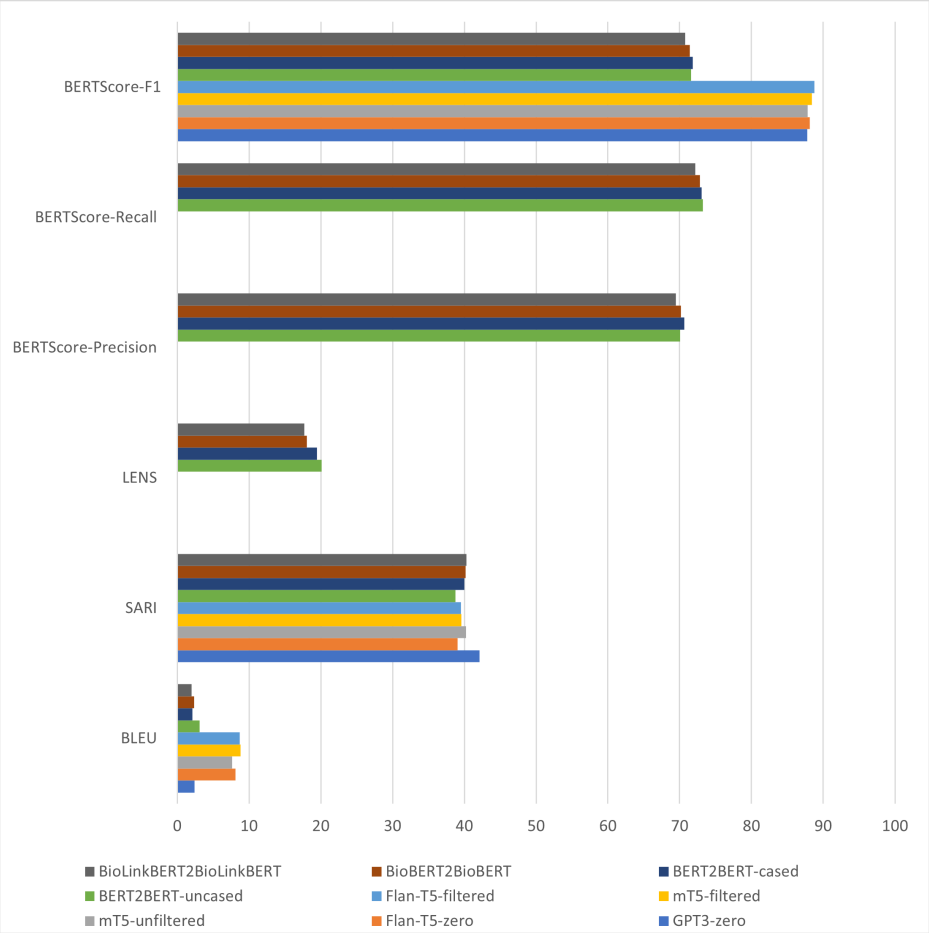


Figure 6.2 Evaluation metrics comparing our models to related work

For our experiments, we selected the SELLS dataset [54] due to its richness, which stems from its derivation from multiple sources and its substantial size. This comprehensive dataset provided a robust foundation for fine-tuning our models, enabling a thorough evaluation of their performance in generating simplified medical sentences that retain domain-specific knowledge.

We aimed to simplify medical sentences by fine-tuning a Transformer model initialized with four distinct LLM checkpoints. The first and second models were initialized with a general-purpose LLM, BERT-base (cased, trained over a single epoch and uncased, trained over three epochs). Conversely, the third and fourth models were initialized with a domain-specific LLM checkpoints, BioBERT-base-cased-v1.1 and BioLinkBERT-base, which are tailored to biomedical texts, trained for a single epoch due to computational constraints.

Despite the disparity in training duration, the evaluation metrics indicated that all the fine-tuned models achieved comparable performance. BioBERT and BioLinkBERT were the least effective models in terms of evaluation metrics. Notably, even with constrained

training, the bert-base-cased model produced high-quality simplified sentences compared to the other models. Based on our findings in Chapter 5, we hypothesize that extending the training duration of the Transformer model initialized with the bert-base-cased checkpoint to three epochs, similar to the bert-base-uncased model, would significantly enhance the quality of the simplified medical sentences. This extended training period would likely enable the model to better understand the dataset, resulting in superior simplifications tailored to the medical domain.

From these observations, we can infer that a general-purpose LLM checkpoint, such as bert-base-cased, performs optimally when fine-tuned on a broad, general dataset for the task of simplifying general sentences. Furthermore, it shows superior performance compared to domain-specific LLMs like BioBERT-base-cased-v1.1 and BioLinkBERT-base when fine-tuned on a biomedical dataset for simplifying medical domain sentences.

Chapter 7

Conclusions and Future Work

This thesis has targeted English sentence simplification problem through the development of Sequence-to-Sequence (Seq2Seq) models. The multiple Seq2Seq models developed are capable of producing simplified English sentences from complex inputs. These methods effectively streamline the original sentences. Moreover, the thesis identifies evaluation metrics that accurately measure the quality of the simplified sentences. In addition it applies these models in the medical domain. This chapter outlines the main findings and contributions of the thesis, emphasizing future work aimed at addressing existing limitations.

7.1 Conclusions

English sentence simplification has emerged as a critical challenge within Natural Language Processing(NLP), gathering attention since the late 1990s. This study centers on Sequence-to-Sequence approaches, widely acknowledged for their efficacy in addressing diverse NLP tasks. Our investigation applies these methodologies specifically to the task of sentence simplification, examining the influence of various architectural components and exploring multiple alternatives. The findings from our extensive experiments aim to provide robust insights and recommendations to steer future research in this area effectively. Additionally, a crucial aspect of developing such models involves devising methodologies to evaluate their outputs effectively, thereby facilitating continual improvement in performance. Moreover, we investigate the application of our best-performing sentence simplification model in the medical domain, aiming to enhance accessibility and comprehension of medical texts, facilitate easier communication of complex medical concepts to patients and make general medical information more accessible to the public.

Chapter 2 provides a comprehensive review of the literature on Text Simplification research, with a particular focus on the English language, which has seen significant advance-

ments in this field. The review encompasses various approaches to addressing the problem, evaluation methodologies, and available data resources. Additionally, it discusses related work and state-of-the-art English sentence simplification models, particularly those based on the Seq2Seq framework.

Chapter 3 establishes the foundational information essential for the subsequent chapters, serving as the groundwork for the contributions presented in this thesis. Our research questions are addressed in the following chapters.

In Chapter 4, we start tackling the problem of sentence simplification from the evaluation side. In order to build robust Seq2Seq models, we should have a systematic way to evaluate their outcomes comparing them and with the related work in the area. We worked on NEWSELA-LIKERT dataset [43], and started with basic readability metrics to see whether they reflect the different complexity levels of the dataset or not. The result was most of the assessed metrics did not show any significant difference among the complex sentence and the different simplified sentences. Then, we focused on the reference-based evaluation metrics commonly used in the literature namely, BLEU, SARI, BERTScore and LENS. A meta-evaluation study was conducted to evaluate these reference-based metrics.

Following this study and considering our findings we can answer **Research Question-1:** Do existing text simplification evaluation metrics correlate with human judgement? Our findings as follow:

- LENS has a medium to strong correlation with all the human evaluation aspects: Simplicity, Adequacy and Fluency.
- $BERTScore_{precision}$, $BERTScore_{Recall}$ and $BERTScore_{F1}$ have medium correlation with Fluency.
- $BERTScore_{Recall}$ has moderate correlation with Adequacy.

We conclude that LENS is recommended at the first place to measure the three distinct aspects of simplified sentences: Simplicity, Adequacy and Fluency. Additionally, for a precise evaluation of Fluency in simplified sentences, $BERTScore_{precision}$, $BERTScore_{Recall}$ and $BERTScore_{F1}$ are recommended metrics. Furthermore, $BERTScore_{Recall}$ can be used to support the assessment of meaning preservation. Nevertheless, we suggest that reporting BLEU and SARI metrics be continued to facilitate comparisons between the performance of newly developed models and previously published results by researchers.

Moreover, Thresholds were established for each recommended metric based on benchmarks set by state-of-the-art models, thereby aiding the evaluation of forthcoming simplification models.

At the end of Chapter 4, an analysis was conducted to explore the impact of the increasing number of references on the reference-based metrics scores. The number of references has a positive impact on the metrics' values. A positive correlation defined between the total number of references and the reference-based metrics. Except for SARI, the values demonstrated an increase with the four references compared to single reference, and subsequently, they remained stable or even slightly decreased with a further increase in the number of references.

Chapter 5 covers the development of Seq2Seq models for sentence simplification task and answer the second, third and fourth research questions.

Research Question 2: How does the performance of Gated Recurrent Units (GRUs) compare to Long Short-Term Memory (LSTM) units in Sequence-to-Sequence (Seq2Seq) sentence simplification models? We implemented encoder-decoder model with attention-based mechanism and compared the performance of GRU layer in both encoder and decoder sides with the performance using LSTM layer in both sides. The experiments cover training, validating and testing on single and multiple data sources. Our findings indicate that simplified sentences generated by LSTM units significantly outperformed GRU units based on the evaluation metrics. Also, in terms of computational efficiency, LSTM units demonstrate slightly better performance in reducing the time required for training, validation, and testing cycles. LSTM models achieve a lower loss value, indicating more effective optimization during the training process. This provides a clear indication that LSTM-based models are better suited for achieving higher quality simplified sentences. Additionally, the results show that utilizing multiple data sources enhances the performance of attention-based encoder-decoder sentence simplification models compared to relying on a single data source. Our attention-based LSTM model demonstrates superior performance compared to the state-of-the-art DRESS model [48], which employs reinforcement learning with an LSTM architecture. This improvement highlights the effectiveness of integrating attention mechanisms into LSTM-based sentence simplification models. Unlike traditional LSTM models that rely solely on sequential processing, the attention mechanism dynamically identifies and focuses on the most relevant parts of the input sentence, enhancing the model's ability to capture contextual dependencies. Consequently, this results in more accurate, fluent, and contextually appropriate simplifications. Surpassing the performance of DRESS also underscores the potential of attention mechanisms to eliminate the need for reinforcement learning-based optimization, offering a more straightforward and computationally efficient approach to achieving state-of-the-art results in sentence simplification tasks.

Research Question 4: Can pre-trained Large Language Models (LLMs) be fine-tuned to simplify complex sentences? To answer this question, we successfully fine-tune Transformer

encoder-decoder models utilizing pre-trained LLM checkpoints for the sentence simplification task. We investigate the effects of various LLM checkpoints, including BERT (cased and uncased) and RoBERTa, on the model performance. Additionally, we examine the impact of sharing parameters between the encoder and decoder to optimize memory usage. In comparing the performance of various LLM checkpoints, we found that the bert-base-cased checkpoint with multiple data sources yielded the best performance among the different LLM checkpoints evaluated.

Research Question 3: Does the choice of text embedding technique affect the performance of Sequence-to-Sequence (Seq2Seq) sentence simplification models, and if so, how? To answer this question a comparative analysis of how non-contextual pre-trained embeddings such as GloVe versus self-trained embeddings can influence the performance of the GRU encoder-decoder model with attention-based mechanism. Our findings state that the quality of simplified sentences generated using encoder-decoder models with pre-trained embeddings measured by the evaluation metrics consistently outperformed those generated with self-trained embeddings.

After that, we evaluate the performance of contextual embeddings like BERT, by leveraging them into Transformer encoder-decoder architectures. These models yield the highest quality simplified sentences among all the other experiments. This success can be attributed to the rich semantic and syntactic knowledge embedded in the pre-trained LLMs combined with the powerful attention mechanisms inherent in the Transformer architecture.

The choice of text embedding technique significantly impacts the performance of Seq2Seq simplification models. Embedding techniques are critical as they transform textual data into numerical vectors, capturing semantic and syntactic information that the model uses to understand complex input and generate simplified sentences. Different embedding techniques, such as GloVe, and contextual embeddings like BERT, can influence the quality of the model's outputs.

Chapter 6 addresses **Research Question 5:** Does fine-tuning a domain-specific Large Language Model (LLM) on a domain- specific dataset lead to enhanced quality in simplified sentences? To explore this, we concentrated on the medical domain, employing the optimal model architecture identified in our experiments from Chapter 5 for the Seq2Seq sentence simplification task. We aimed to simplify medical sentences by fine-tuning a Transformer model initialized with four distinct LLM checkpoints. The first and second models were initialized with a general-purpose LLM, BERT-base (cased, trained over a single epoch and uncased, trained over three epochs). Conversely, the third and fourth models were initialized with a domain-specific LLM checkpoints, BioBERT-base-cased-v1.1 and BioLinkBERT-

base, which are tailored to biomedical texts, trained for a single epoch due to computational constraints.

Despite the disparity in training duration, the evaluation metrics indicated that all the fine-tuned models achieved comparable performance. BioBERT and BioLinkBERT were the least effective models in terms of evaluation metrics. Notably, even with constrained training, the bert-base-cased model produced high-quality simplified sentences compared to the other models. Based on our findings in Chapter 5, we hypothesize that extending the training duration of the Transformer model initialized with the bert-base-cased checkpoint to three epochs, similar to the bert-base-uncased model, would significantly enhance the quality of the simplified medical sentences. This extended training period would likely enable the model to better understand the dataset, resulting in superior simplifications tailored to the medical domain.

From these observations, we can infer that a general-purpose LLM checkpoint, such as bert-base-cased, performs optimally when fine-tuned on a broad, general dataset and tasked with simplifying general sentences. Additionally, it demonstrates superior performance compared to specific-domain LLMs, BioBERT-base-cased-v1.1 and BioLinkBERT-base, when fine-tuned on a dataset tailored to a specific domain and used to simplify domain-specific sentences.

7.2 Limitations

In this research, we encountered notable computational constraints that significantly impacted our experimental outcomes. Natural Language Understanding and Text Generation tasks require extensive computational resources, both in terms of processing time and memory capacity. Due to these limitations, we were restricted to a maximum runtime of three days on a GPU for our experiments. Consequently, this constraint hindered the performance of our models, particularly when compared to those developed by larger organizations with access to more substantial computational resources.

Another significant constraint we faced was the limited availability of data resources in languages other than English, such as Arabic, and in other application domains. This limitation hindered our ability to extend and apply our research to other languages or specialized domains beyond the scope of English language and general and medical text simplification tasks. As a result, our exploration and findings primarily focused on English-language datasets and applications, limiting the generalizability of our approach to multilingual or domain-specific contexts. Future research in this area would benefit from addressing these

data limitations to enable broader applicability and validation across diverse linguistic and domain-specific settings.

7.3 Future Work

In terms of leveraging pre-trained Large Language Model (LLM) checkpoints for sentence simplification models, optimizing the training hyperparameters can lead to significantly better performance, provided adequate computational resources are available. With sufficient computational power, fine-tuning processes can be extended and hyperparameters such as learning rate, batch size, and the number of epochs can be meticulously optimized. This optimization would enable the models to more effectively learn the nuances of the dataset, resulting in enhanced quality of the simplified sentences produced.

In terms of generalization to paragraph-level or document-level simplification, fine-tuning Transformer models initialized with LLM checkpoints shows significant potential. Additionally, exploring multilingual models that can simplify text across various languages could further expand the applicability and versatility of these models. By leveraging the robust capabilities of LLMs, it becomes feasible to address more complex simplification tasks beyond single sentences and extend the benefits of text simplification to a wider array of languages, thereby broadening the impact and accessibility of this technology.

Moreover, the application of simplified text extends beyond the medical field to various other domains, such as children's books. Simplifying text in educational materials for young readers can significantly enhance comprehension and learning, making complex concepts more accessible. Additionally, simplified text can benefit non-native speakers, individuals with cognitive disabilities, and adults with low literacy levels by improving their understanding of various subjects. This broader application underscores the importance of developing robust text simplification models that can serve diverse audiences and content types, ensuring that information is universally accessible and comprehensible.

Future research can further extend the evaluation metrics BERTScore and LENS to other domains, such as medical, as the preference for word tokenization and different simplification operations can vary depending on the domain. Additionally, these metrics currently focus on sentence-level simplification, and future work can extend them to evaluate paragraph and document-level simplification. Moreover, since LENS is currently limited to the English language, future research can work on extending this metric to evaluate other languages.

References

[1] J. J. J. Brunning, *Alignment models and algorithms for statistical machine translation*. PhD thesis, University of Cambridge, 2010.

[2] S. Narayan and C. Gardent, "Hybrid simplification using deep semantics and machine translation," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (Baltimore, Maryland), pp. 435–445, Association for Computational Linguistics, June 2014.

[3] A. Siddharthan and A. Mandya, "Hybrid text simplification using synchronous dependency grammars with hand-written and automatically harvested rules," in *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics* (S. Wintner, S. Goldwater, and S. Riezler, eds.), (Gothenburg, Sweden), pp. 722–731, Association for Computational Linguistics, Apr. 2014.

[4] M. Shardlow, *Lexical simplification: optimising the pipeline*. PhD thesis, University of Manchester, UK, 2015. British Library, EThOS.

[5] W. Xu, C. Callison-Burch, and C. Napoles, "Problems in current text simplification research: New data can help," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 283–297, 2015.

[6] M. Phi, "Illustrated guide to LSTM's and GRU's: A step by step explanation," 2020. Accessed: 2024-01-09.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.

[8] A. Siddharthan, "Complex lexico-syntactic reformulation of sentences using typed dependency representations," in *Proceedings of the 6th International Natural Language Generation Conference* (J. Kelleher, B. M. Namee, and I. v. d. Sluis, eds.), Association for Computational Linguistics, July 2010.

[9] K. Woodsend and M. Lapata, "Learning to simplify sentences with quasi-synchronous grammar and integer programming," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing* (R. Barzilay and M. Johnson, eds.), (Edinburgh, Scotland, UK.), pp. 409–420, Association for Computational Linguistics, July 2011.

[10] T. Wang, P. Chen, J. Rochford, and J. Qiang, "Text simplification using neural machine translation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, Mar. 2016.

[11] N. Alfear, D. Kazakov, and H. Al-Khalifa, "Meta-evaluation of sentence simplification metrics," in *Proceedings of LREC-COLING 2024*, May 2024. Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC-COLING 2024 ; Conference date: 20-05-2024 Through 25-05-2024.

[12] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2 ed., 2010.

[13] G. Chowdhury, "Natural language processing," *ARIST*, vol. 37, pp. 51–89, 01 2005.

[14] A. Siddharthan, "A survey of research on text simplification," *ITL - International Journal of Applied Linguistics*, vol. 165, pp. 259–298, 12 2014.

[15] J. Patterson and A. Gibson, *Deep Learning: A Practitioner's Approach*. Beijing: O'Reilly, 2017.

[16] R. Chandrasekar, C. Doran, and B. Srinivas, "Motivations and methods for text simplification," in *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*, 1996.

[17] R. Chandrasekar and B. Srinivas, "Automatic induction of rules for text simplification1revised version of the article originally published in knowledge-based computer systems: Research and applications. (eds k.s.r. anjaneyulu, m. sasikumar and s. ramani) narosa publishing house, new delhi, 1997.1," *Knowledge-Based Systems*, vol. 10, no. 3, pp. 183–190, 1997.

[18] V. Seretan, "Acquisition of syntactic simplification rules for French," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)* (N. Calzolari, K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, eds.), (Istanbul, Turkey), pp. 4019–4026, European Language Resources Association (ELRA), May 2012.

[19] R. Hazim, H. Saddiki, B. Alhafni, M. Al Khalil, and N. Habash, "Arabic word-level readability visualization for assisted text simplification," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (W. Che and E. Shutova, eds.), (Abu Dhabi, UAE), pp. 242–249, Association for Computational Linguistics, Dec. 2022.

[20] S. Bott and H. Saggion, "Spanish text simplification: An exploratory study," *Procesamiento de Lenguaje Natural*, vol. 47, pp. 87–95, 01 2011.

[21] S. M. Aluísio, L. Specia, T. A. Pardo, E. G. Maziero, and R. P. Fortes, "Towards brazilian portuguese automatic text simplification systems," in *Proceedings of the Eighth ACM Symposium on Document Engineering*, DocEng '08, (New York, NY, USA), p. 240–248, Association for Computing Machinery, 2008.

[22] A. Dmitrieva and A. Y. Malafeev, "Text simplification for Russian as a foreign language," 2016.

[23] D. Klaper, S. Ebling, and M. Volk, "Building a German/simple German parallel corpus for automatic text simplification," in *Proceedings of the Second Workshop on Predicting and Improving Text Readability for Target Reader Populations* (S. Williams, A. Siddharthan, and A. Nenkova, eds.), (Sofia, Bulgaria), pp. 11–19, Association for Computational Linguistics, Aug. 2013.

[24] F. Dell'Orletta, S. Montemagni, and G. Venturi, "READ–IT: Assessing readability of Italian texts with a view to text simplification," in *Proceedings of the Second Workshop on Speech and Language Processing for Assistive Technologies* (N. Alm, ed.), (Edinburgh, Scotland, UK), pp. 73–83, Association for Computational Linguistics, July 2011.

[25] W. Daelemans, A. Othker, E. Tjong, and K. Sang, "Automatic sentence simplification for subtitling in Dutch and English," 07 2004.

[26] L. Specia, "Translating from complex to simplified sentences," in *Computational Processing of the Portuguese Language* (T. A. S. Pardo, A. Branco, A. Klautau, R. Vieira, and V. L. S. de Lima, eds.), (Berlin, Heidelberg), pp. 30–39, Springer Berlin Heidelberg, 2010.

[27] W. Coster and D. Kauchak, "Learning to simplify sentences using Wikipedia," in *Proceedings of the Workshop on Monolingual Text-To-Text Generation* (K. Filippova and S. Wan, eds.), (Portland, Oregon), pp. 1–9, Association for Computational Linguistics, June 2011.

[28] S. Wubben, A. van den Bosch, and E. Krahmer, "Sentence simplification by monolingual machine translation," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (H. Li, C.-Y. Lin, M. Osborne, G. G. Lee, and J. C. Park, eds.), (Jeju Island, Korea), pp. 1015–1024, Association for Computational Linguistics, July 2012.

[29] Z. Zhu, D. Bernhard, and I. Gurevych, "A monolingual tree-based translation model for sentence simplification," in *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)* (C.-R. Huang and D. Jurafsky, eds.), (Beijing, China), pp. 1353–1361, Coling 2010 Organizing Committee, Aug. 2010.

[30] W. Xu, C. Napoles, E. Pavlick, Q. Chen, and C. Callison-Burch, "Optimizing statistical machine translation for text simplification," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 401–415, 2016.

[31] C. Horn, C. Manduca, and D. Kauchak, "Learning a lexical simplifier using Wikipedia," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (K. Toutanova and H. Wu, eds.), (Baltimore, Maryland), pp. 458–463, Association for Computational Linguistics, June 2014.

[32] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," Jan. 2015. 3rd International Conference on Learning Representations, ICLR 2015 ; Conference date: 07-05-2015 Through 09-05-2015.

[33] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (A. Moschitti, B. Pang, and W. Daelemans, eds.), (Doha, Qatar), pp. 1724–1734, Association for Computational Linguistics, Oct. 2014.

[34] J. Chung, Çaglar Gülçehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *ArXiv*, vol. abs/1412.3555, 2014.

[35] S. Nisioi, S. Štajner, S. P. Ponzetto, and L. P. Dinu, "Exploring neural text simplification models," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (R. Barzilay and M.-Y. Kan, eds.), (Vancouver, Canada), pp. 85–91, Association for Computational Linguistics, July 2017.

[36] T. Vu, B. Hu, T. Munkhdalai, and H. Yu, "Sentence simplification with memory-augmented neural networks," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)* (M. Walker, H. Ji, and A. Stent, eds.), (New Orleans, Louisiana), pp. 79–85, Association for Computational Linguistics, June 2018.

[37] S. Zhao, R. Meng, D. He, A. Saptono, and B. Parmanto, "Integrating transformer and paraphrase rules for sentence simplification," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, eds.), (Brussels, Belgium), pp. 3164–3173, Association for Computational Linguistics, Oct.-Nov. 2018.

[38] S. Bott, L. Rello, B. Drndarevic, and H. Saggion, "Can Spanish be simpler? LexSiS: Lexical simplification for Spanish," pp. 357–374, 12 2012.

[39] G. Glavaš and S. Štajner, "Simplifying lexical simplification: Do we need simplified corpora?," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (C. Zong and M. Strube, eds.), (Beijing, China), pp. 63–68, Association for Computational Linguistics, July 2015.

[40] S. Štajner and G. Glavaš, "Leveraging event-based semantics for automated text simplification," *Expert Systems with Applications*, vol. 82, pp. 383–395, 2017.

[41] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Physics. Doklady*, vol. 10, pp. 707–710, 1965.

[42] L. Martin, É. de la Clergerie, B. Sagot, and A. Bordes, "Controllable sentence simplification," in *Proceedings of the Twelfth Language Resources and Evaluation Conference* (N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, eds.), (Marseille, France), pp. 4689–4698, European Language Resources Association, May 2020.

[43] M. Maddela, F. Alva-Manchego, and W. Xu, "Controllable text simplification with explicit paraphrasing," 2021.

[44] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.

[45] K. C. Sheang and H. Saggion, "Controllable sentence simplification with a unified text-to-text transfer transformer," in *Proceedings of the 14th International Conference on Natural Language Generation* (A. Belz, A. Fan, E. Reiter, and Y. Sripada, eds.), (Aberdeen, Scotland, UK), pp. 341–352, Association for Computational Linguistics, Aug. 2021.

[46] W. Coster and D. Kauchak, "Simple English Wikipedia: A new text simplification task," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (D. Lin, Y. Matsumoto, and R. Mihalcea, eds.), (Portland, Oregon, USA), pp. 665–669, Association for Computational Linguistics, June 2011.

[47] D. Kauchak, "Improving text simplification language modeling using unsimplified text data," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (H. Schuetze, P. Fung, and M. Poesio, eds.), (Sofia, Bulgaria), pp. 1537–1546, Association for Computational Linguistics, Aug. 2013.

[48] X. Zhang and M. Lapata, "Sentence simplification with deep reinforcement learning," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (M. Palmer, R. Hwa, and S. Riedel, eds.), (Copenhagen, Denmark), pp. 584–594, Association for Computational Linguistics, Sept. 2017.

[49] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (J. Burstein, C. Doran, and T. Solorio, eds.), (Minneapolis, Minnesota), pp. 4171–4186, Association for Computational Linguistics, June 2019.

[50] C. Jiang, M. Maddela, W. Lan, Y. Zhong, and W. Xu, "Neural CRF model for sentence alignment in text simplification," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, (Online), pp. 7943–7960, Association for Computational Linguistics, July 2020.

[51] F. Alva-Manchego, L. Martin, A. Bordes, C. Scarton, B. Sagot, and L. Specia, "ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, (Online), pp. 4668–4679, Association for Computational Linguistics, July 2020.

[52] Newsela, "Newsela | Request Newsela Data."

[53] C. Lennon and H. Burdick, "The lexile framework as an approach for reading measurement and success," *electronic publication on www. lexile. com*, 2004.

[54] Y. Guo, W. Qiu, G. Leroy, S. Wang, and T. Cohen, "Retrieval augmentation of large language models for lay language generation," *Journal of Biomedical Informatics*, vol. 149, p. 104580, 2024.

[55] A. E. W. Johnson, T. J. Pollard, L. Shen, L.-W. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark, "MIMIC-III, a freely accessible critical care database," *Scientific Data*, vol. 3, p. 160035, May 2016.

[56] J. Luo, J. Lin, C. Lin, C. Xiao, X. Gui, and F. Ma, "Benchmarking automated clinical language simplification: Dataset, algorithm, and evaluation," in *Proceedings of the 29th International Conference on Computational Linguistics* (N. Calzolari, C.-R. Huang, H. Kim, J. Pustejovsky, L. Wanner, K.-S. Choi, P.-M. Ryu, H.-H. Chen, L. Donatelli, H. Ji, S. Kurohashi, P. Paggio, N. Xue, S. Kim, Y. Hahm, Z. He, T. K. Lee, E. Santus, F. Bond, and S.-H. Na, eds.), (Gyeongju, Republic of Korea), pp. 3550–3562, International Committee on Computational Linguistics, Oct. 2022.

[57] S. Joseph, K. Kazanas, K. Reina, V. Ramanathan, W. Xu, B. Wallace, and J. J. Li, "Multilingual simplification of medical texts," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (H. Bouamor, J. Pino, and K. Bali, eds.), (Singapore), pp. 16662–16692, Association for Computational Linguistics, Dec. 2023.

[58] F. Alva-Manchego, C. Scarton, and L. Specia, "Data-driven sentence simplification: Survey and benchmark," *Computational Linguistics*, vol. 46, no. 1, pp. 135–187, 2020.

[59] R. F. Flesch, "A new readability yardstick.," *The Journal of applied psychology*, vol. 32 3, pp. 221–33, 1948.

[60] T. Tanprasert and D. Kauchak, "Flesch-kincaid is not a text simplification evaluation metric," in *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)* (A. Bosselut, E. Durmus, V. P. Gangal, S. Gehrmann, Y. Jernite, L. Perez-Beltrachini, S. Shaikh, and W. Xu, eds.), (Online), pp. 1–14, Association for Computational Linguistics, Aug. 2021.

[61] R. Gunning, *The Technique of Clear Writing*. McGraw-Hill, 1952.

[62] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

[63] G. Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," in *Proceedings of the second international conference on Human Language Technology Research*, pp. 138–145, 2002.

[64] M. G. Snover, N. Madnani, B. Dorr, and R. Schwartz, "Ter-plus: paraphrase, semantic, and alignment enhancements to translation edit rate," *Machine Translation*, vol. 23, pp. 117–127, 2009.

[65] S. Stajner, M. Popovic, H. Saggion, L. Specia, and M. Fishel, "Shared task on quality assessment for text simplification," 05 2016.

[66] F. Alva-Manchego, C. Scarton, and L. Specia, "The (un)suitability of automatic evaluation metrics for text simplification," *Computational Linguistics*, vol. 47, pp. 861–889, Dec. 2021.

[67] P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced BERT with disentangled attention," in *International Conference on Learning Representations*, 2021.

[68] T. Zhang*, V. Kishore*, F. Wu*, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with bert," in *International Conference on Learning Representations*, 2020.

[69] H. Sun and M. Zhou, "Joint learning of a dual SMT system for paraphrase generation," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (H. Li, C.-Y. Lin, M. Osborne, G. G. Lee, and J. C. Park, eds.), (Jeju Island, Korea), pp. 38–42, Association for Computational Linguistics, July 2012.

[70] E. Sulem, O. Abend, and A. Rappoport, "Semantic structural evaluation for text simplification," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, (New Orleans, Louisiana), pp. 685–696, Association for Computational Linguistics, June 2018.

[71] OpenAI, "ChatGPT." https://www.openai.com/gpt-3, 2023.

[72] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019.

[73] M. Maddela, Y. Dou, D. Heineman, and W. Xu, "LENS: A learnable evaluation metric for text simplification," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (Toronto, Canada), pp. 16383–16408, Association for Computational Linguistics, July 2023.

[74] E. Williams, *Regression Analysis*. WILEY SERIES in PROBABILITY and STATISTICS: APPLIED PROBABILITY and STATIST ICS SECTION Series, Wiley, 1959.

[75] E. Sulem, O. Abend, and A. Rappoport, "Simple and effective text simplification using semantic and neural methods," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (I. Gurevych and Y. Miyao, eds.), (Melbourne, Australia), pp. 162–173, Association for Computational Linguistics, July 2018.

[76] E. Sulem, O. Abend, and A. Rappoport, "BLEU is not suitable for the evaluation of text simplification," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, eds.), (Brussels, Belgium), pp. 738–744, Association for Computational Linguistics, Oct.-Nov. 2018.

[77] Y. Dong, Z. Li, M. Rezagholizadeh, and J. C. K. Cheung, "EditNTS: An neural programmer-interpreter model for sentence simplification through explicit editing," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (Florence, Italy), pp. 3393–3402, Association for Computational Linguistics, July 2019.

[78] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mT5: A massively multilingual pre-trained text-to-text transformer," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, eds.), (Online), pp. 483–498, Association for Computational Linguistics, June 2021.

[79] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei, "Scaling instruction-finetuned language models," 2022.

[80] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, pp. 1735–1780, 11 1997.

[81] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation* (D. Wu, M. Carpuat, X. Carreras, and E. M. Vecchi, eds.), (Doha, Qatar), pp. 103–111, Association for Computational Linguistics, Oct. 2014.

[82] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[83] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "Biobert: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.

[84] P. Schober, C. Boer, and L. A. Schwarte, "Correlation coefficients: appropriate use and interpretation," *Anesthesia & analgesia*, vol. 126, no. 5, pp. 1763–1768, 2018.

[85] L. Puka, *Kendall's Tau*, pp. 713–715. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[86] F. Alva-Manchego, L. Martin, C. Scarton, and L. Specia, "EASSE: Easier automatic sentence simplification evaluation," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, (Hong Kong, China), pp. 49–54, Association for Computational Linguistics, Nov. 2019.

[87] G. Corder and D. Foreman, *Nonparametric Statistics for Non-Statisticians: A Step-by-Step Approach*. Wiley, 2009.

[88] S. Rothe, S. Narayan, and A. Severyn, "Leveraging pre-trained checkpoints for sequence generation tasks," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 264–280, 2020.

[89] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[90] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *et al.*, "Improving language understanding by generative pre-training," 2018.

[91] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016.

[92] T. Luong, I. Sutskever, Q. Le, O. Vinyals, and W. Zaremba, "Addressing the rare word problem in neural machine translation," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (C. Zong and M. Strube, eds.), (Beijing, China), pp. 11–19, Association for Computational Linguistics, July 2015.

[93] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," in *International Conference on Learning Representations*, 2020.

[94] M. Yasunaga, J. Leskovec, and P. Liang, "LinkBERT: Pretraining language models with document links," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (S. Muresan, P. Nakov, and A. Villavicencio, eds.), (Dublin, Ireland), pp. 8003–8016, Association for Computational Linguistics, May 2022.

[95] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, and H. Poon, "Domain-specific language model pretraining for biomedical natural language processing," *ACM Transactions on Computing for Healthcare*, vol. 3, p. 1–23, Oct. 2021.

[96] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[97] T. U. of York, "Viking documentation." https://vikingdocs.york.ac.uk/index.html, 2023. Accessed: 2023-10-09.

Viking Cluster Specification

Viking is the University of York high-performance computing facility for research projects. It is a large, Linux compute cluster with many nodes, CPUs, GPUs, lots of storage and boat loads of memory [97]. Viking is hosted in a carbon-neutral data center, EcoDataCenter, in Sweden. It is powered using 100% renewable energies and utilizes heat recovery from hardware.

Cluster Configuration

Compute node only CPU cores	12,864
Total standard compute nodes	134
Processor generation	AMD EPYC3 7643
Cores per processor	48
Number of processors per node	2
Memory per compute node	512 GB
High memory node	2x 2 TB 1x 4 TB
GPUs	48x A40 12x H100
Scratch	1.5 PB
Usable NVME storage	215 TB
Interconnect type	100Gb OPA