Master Thesis

Words Made Easy: a Comparative Study of Methods for English Lexical Simplification

Irma Tuinenga

a thesis submitted in partial fulfilment of the requirements for the degree of

MA Linguistics
(Text Mining)

Vrije Universiteit Amsterdam

Computational Lexicology and Terminology Lab
Department of Language and Communication
Faculty of Humanities

Supervised by: Dr. Luís de Passos Morgada da Costa, Dr. Hennie van der Vliet
2^{nd} reader: Dr. Lucia Donatelli

Submitted: August 15, 2023

Abstract

Accessible information for all people is a fundamental societal need. However, textual complexity often hinders reading comprehension, especially for those with low literacy or language skills. The Natural Language Processing (NLP) task of Lexical Simplification aims to facilitate the comprehensibility of textual information. In this task, complex words in a text are replaced with simpler, easier-to-understand alternatives.

This thesis presents various methodologies for English Lexical Simplification, based on Masked Language Model (MLM) technology combined with additional methods. It focuses on the consecutive stages of generating, selecting, and ranking alternatives for given complex words, adhering to the requirements for the TSAR-2022 Shared Task on Multilingual Lexical Simplification. In addition, this research introduces an innovative model for Lexical Simplification, outperforming LSBert, a recent benchmark. The model uses MLM technology and BERTScore's contextualized embeddings for enhanced semantic accuracy. Despite the lack of a designated simplicity-ranking mechanism, it surpasses comparable models with such properties, suggesting a need for further investigation into the notion of 'simplicity' in this context. The various needs of different reading audiences should be incorporated in such research. These needs should be reflected in the instructions for a Lexical Simplification task, among which the task's principal system evaluation metrics. For this purpose, this study proposes a novel measure grounded on target audience requirements.

Post-evaluation improvements leverage WordNet's semantic hierarchy to determine whether substitutes serve as hypernyms for the complex word, introducing an innovative approach in Lexical Simplification. The promising results suggest additional studies into WordNet's capabilities to enhance Lexical Simplification models, possibly in conjunction with other relevant categorizations.

This thesis also explores the real-world applicability of the obtained insights to an existing readability analyzer for English and Dutch. Notably, the Dutch version could face potential challenges due to resource scarcity and morphological characteristics.

Declaration of Authorship

I, Irma Tuinenga, declare that this thesis, titled *Words Made Easy: a Comparative Study of Methods for English Lexical Simplification* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: August 15, 2023

Signed: Irma Tuinenga

Acknowledgments

This thesis project marks the end of my MA Linguistics (Text Mining) study.

I would like to express my deepest gratitude to my thesis supervisor, Dr. Luís de Passos Morgado da Costa. Your observations and encouragements triggered me to explore the task of Lexical Simplification further than I could have ever dreamed. Furthermore, I am grateful to Dr. Hennie van der Vliet, co-supervisor for my thesis. I value your pragmatical approach. Thank you to Dr. Lucia Donatelli, the second reader of my thesis. You provided useful feedback on both my thesis plan and my thesis presentation.

In addition, many thanks to Mark Breuker. I appreciate your ideas to make this thesis useful for EDIA, as well as your flexibility that enabled me to carry out this project with a lot of freedom. And Ludvig Rassmus, thank you for providing technical information about EDIA's readability analyzer.

Moreover, I would like to thank my colleagues at ORTEC. I value your support during the course of my study.

I had the pleasure of working together with many talented students during this year, for which I am grateful. Each of you brought your unique and valuable input to our group projects. I especially extend my sincere appreciation to Noah and Margriet for your friendship and support. We found each other in our ambitions to push the boundaries of our potential — which also made this year challenging for us.

Lastly, embarking on this journey would have been impossible without my family and friends. Thank you for your continuous love and support during this demanding year. I look forward to spending more cherished moments together.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

The United Nations state that all people should have the right to accessible information[1]. This is crucial to social inclusion and active participation in society, as laid out by Stajner (2021). But how do we define accessibility? Even if people are getting access to information, the information in a variety of textual sources can be too complex for people to understand. For example, this can be the case for citizens with low literacy, cognitive disabilities, or low command of the native language of their country of citizenship. Therefore, although people may have clear access to information, if they don't understand this information, they are unable to make informed choices concerning essential matters such as healthcare decisions, legal assistance, education, or democratic rights (Stajner et al., 2022). This presents a significant challenge to governments and other organizations as they strive to effectively communicate essential information to the people involved. The OECD Adult Literacy Report (OECD, 2019; Stajner, 2021) reveals that this problem is not limited to just a few people. Six proficiency levels in literacy were defined, against which the reading abilities of a representative subset from 32 countries were evaluated. The results show that 19,8% of this population did not score higher than the lowest two literacy levels (OECD, 2019, p.43), corresponding to this group's need for simplification of any text that exceeds a basic vocabulary.

Understanding and addressing the complexity of textual information to make it more comprehensible has led to the emergence of an Natural Language Processing (NLP) task known as Text Simplification. This involves summarization, making sentences shorter, using easier grammar, expressing concepts more straightforward, etc. – all with the goal to maintain the meaning of the original message. Another important subtask of Text Simplification entails methodically replacing textually complex or difficult phrases in a text with more straightforward, easily understood alternatives, while preserving the context-specific meanings of these complex phrases. This subtask is called Lexical Simplification, which plays a vital role within the field of Text Simplification, having been addressed by a variety of strategies over time.

Initial attempts to perform automated lexical simplification made use of handcrafted resources that relied on linguistic knowledge. These resources offer a wealth of knowledge that can help improve a system's understanding of natural language. WordNet (Fellbaum, 1998) represents one such resource. It is an English language database where words are classified into synonym clusters, known as synsets, each expressing a distinct concept. WordNet also maps the semantic relationships between

[1] `https://www.un.org/development/desa/disabilities/convention-on-the-rights-of-persons-with-disabilities/article-9-accessibility.html`, last accessed on 2023-08-14.

these synsets, providing a rich network of interconnected language data. In a later stage, parallel corpora became available to simplify texts, like (the more complex) English Wikipedia[2] aligned with (the simpler) Simple Wikipedia[3].

Despite the benefits of using such manually designed resources for text simplification, there are a number of limitations. Starting from the semantic angle, the exact meaning of a word is influenced by its surrounding context. This is disregarded when using a word from a "stand-alone" corpus as a replacement for a complex word, as the complex word's unique context is not taken into account. Another issue arises when considering language variation. Since language changes all the time, handcrafted linguistic corpora will always stay incomplete. These corpora may also be limited to specific domains, which may make them not suitable to simplify texts outside the scope of these domains. The scale and diversity of the over 7,000 languages in the world present another challenge, as creating and updating resources manually for every language is not feasible. It is also very costly, as linguists are needed to create and continuously update these resources. The shortcomings of manually created corpora have become more evident since the world-wide adoption of the internet in the 1990's, making it virtually impossible to keep these corpora up to date with the ever-increasing amount of available textual information. Lastly, standard applications of simplification corpora may not meet the various simplification needs that different target audiences have. In other words, what one person considers a simple word can another person consider a complex word. For instance, recent research has shown that simplification needs seem to be different for native and non-native speakers (Yimam et al., 2017, 2018). For non-native speakers, simplification requirements may not only depend on a person's language proficiency level (Lee and Yeung, 2018), but also on the specific native language of that person (Aprosio et al., 2018). A differential approach could also be appropriate for people with cognitive impairments vs. those with reading impairments (Orasan et al., 2018). The variety of all these simplification needs are not taken into account by the "one-size-fits-all" approach of the vast majority of standard linguistic resources.

These insights call for alternative solutions that can: 1) take the context into account; 2) scale up quickly to serve a wide reading audience of a variety of languages; and 3) accommodate the specific simplification needs of various subgroups within this audience. With regard to the latter requirement, especially if it would be possible to accommodate the unique reading requirements of each person involved, this could be considered as the ultimate goal to achieve for simplifying texts.

During the past decade, the NLP community has given considerable attention to advance the task of Lexical Simplification. For example, a variety of Shared Tasks have been organized around this topic, providing a platform for comparing performances across various systems executing the same task. The development of such systems was often triggered by governments and related civil organizations that have become more aware of the impact of social inclusion and participation in society, and how readable information can contribute positively to these crucial aspects of civic engagement. Several publicly funded projects have been conducted to build systems to assist people in understanding written information (Stajner, 2021). Next to governmental-related initiatives, organizations in the private sector have also found their way to NLP communities to facilitate text comprehension for a variety of audiences, such as second

[2] `https://en.wikipedia.org/wiki/Main_Page`, last accessed on 2023-08-14.
[3] `https://simple.wikipedia.org/wiki/Main_Page`, last accessed on 2023-08-14.

language learners (Saggion et al., 2022). All these developments have helped the field of Lexical Simplification to accelerate in maturity. An increasing number of researchers in NLP use hybrid approaches in which they combine the strengths of lexical resources and parallel corpora with advanced machine learning applications that can derive patterns from large textual resources containing words in their context. This progression brings us closer to achieving the United Nations' objective of enabling all people to access and understand information.

1.1 Lexical Simplification

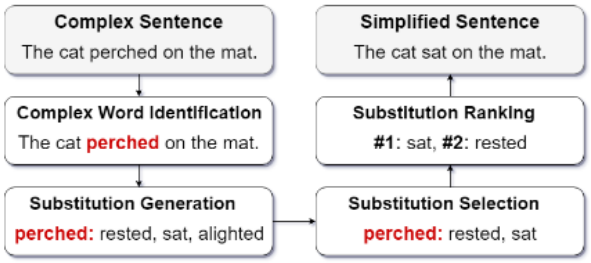


Figure 1.1: Lexical Simplification pipeline, taken from Paetzold and Specia (2017a).

As introduced in the previous section, Lexical Simplification entails methodically replacing textually complex or difficult phrases in a text with more straightforward, easily understood alternatives, while their context-specific meanings are retained. To develop a system to automate this task, the following steps are often (Paetzold and Specia, 2017a) carried out:

1. Complex Word Identification (CWI): in this step, words that are difficult to understand are detected. In the sentence "The cat perched on the mat"(Paetzold and Specia, 2017a) in figure 1.1, the word **perched** was identified as complex. Aspects that influence perceived word complexity are covered in section 2.1.

2. Substitute Generation (SG): this step pertains to generating potential candidates to replace the complex word identified in the first step. It focuses on creating potential candidates for replacing a complex word, ensuring that no promising candidates are excluded. Balancing this step is important. A strategy could be to generate a smaller set of good substitutes, whereas an alternative design decision could be producing many substitutes including less fitting ones and filter or adapt these during the next step. In the latter case, the flow of candidates produced during the SG step could also include unsimilar and grammatically incorrect candidates. Both strategies can be applied, as long as no suitable substitutes are missed. As exemplified in figure 1.1, "The cat perched on the mat" could feature several candidate replacement words for **perched**. Some of these could bear semantic similarity to the complex word, such as *rested* and *sat*, whereas others like *alighted* could not. Another potential substitute could be, for example, the verb *lounge*. However, in its current form, it does not align with the third person singular and the past tense, thus breaking the grammatical structure. The generation of such a substitute in this initial phase depends on the above-mentioned design decision.

3. Substitute Selection (SS): in this step, the only words that are retained are those that are similar in meaning to the complex word in its context and that fit the grammatical structure. The latter may also involve adapting an ungrammatical substitute to the inflection of the complex word[4]. When revisiting the sentence "The cat perched on the mat" in figure 1.1 and the complex word **perched**, the words *rested* and *sat* would be retained when checked on similar meaning in the context of the sentence. The word *alighted* will be discarded, as it expresses a motion, as opposed to the complex word that connotes a static position. An ungrammatical substitute such as *lounge* could be morphologically adapted to the inflection of the complex word, resulting in the grammatically fitting word *lounged*.

4. Substitute Ranking (SR): this step involves ranking of the selected replacement words on how easy they are to understand. From the retained words *rested* and *sat* in figure 1.1, the word *sat* has been identified as most easy, and is therefore the preferred simplification word for **perched** in the sentence "The cat perched on the mat". Factors that influence whether one word is perceived as easier than others are covered in section 2.3.

1.2 Research Objectives

As laid out in the beginning of this chapter, access to understandable information is pivotal to social inclusion and active participation in society. Lexical Simplification can promote the accessibility and understanding of reading materials for various audiences. This includes native speakers who may have reading challenges or impaired reading abilities, as well as non-native speakers with low command of the native language of their country of citizenship.

The principal objective of my thesis project is to help carry forward the continuous pursuit of improving text understanding. It focuses on steps two through four of the Lexical Simplification task, as laid out in section 1.1, for the English language. The thesis project is defined, carried out, and evaluated according to the requirements for the TSAR-2022 Shared Task on Multilingual Lexical Simplification (Stajner et al., 2022; Saggion et al., 2022). In this Shared Task, participants carried out the exact same steps of the Lexical Simplification process. The main goal of this task was how complex words can be transformed into simpler alternatives while preserving the meaning of these complex words in their context. The outcomes of this first Shared Task on Multilingual Lexical Simplification established new reference points in Lexical Simplification.

An additional value of this thesis project lies in its application, due to its development in collaboration with EDIA. EDIA is a company based in the Netherlands that provides insights on content by automating metadata associated with that content. EDIA is currently working on a project with the Dutch government to assist citizens who have a language deficiency. To do so, EDIA uses its multilingual readability analyzer Papyrus[5]. Papyrus can identify the complexity of words by attaching CEFR (Common European Framework of Reference for Languages)[6] language proficiency lev-

[4]This adaptation process may also be executed as a separate step (Saggion et al., 2022), which may be essential for languages with more morphological complexity than English.

[5]`https://www.edia.nl/papyrus`, last accessed on 2023-08-14.

[6]`https://www.coe.int/en/web/common-european-framework-reference-languages`, last accessed on 2023-08-14.

els to these words. The CEFR framework categorizes language competence into six distinct levels: A1, A2, B1, B2, C1, and C2. The A1 and A2 levels signify elementary proficiency, B1 and B2 represent intermediate proficiency, and C1 and C2 indicate advanced levels of language competence. Consequently, Papyrus can identify complex words — the higher the CEFR level, the more complex the word — corresponding to the first step in the Lexical Simplification process outlined in section 1.1. Furthermore, Papyrus generates substitutes that are related to a complex word, i.e., the second step of the Lexical Simplification process. From the generated substitutes, it selects words in accordance with the third step of the Lexical Simplification process. However, this Substitute Selection method could be improved, as Papyrus still retains many words that are not semantically similar to the complex word. With regard to the fourth step of the Lexical Simplification process, in which substitutes are ranked on simplicity, Papyrus uses the CEFR levels of the selected words — the lower the CEFR level, the more simple the word — although it does not *rank* these words based on their levels. After specifying a target CEFR level, Papyrus will provide all substitutes below that level.

The results of this thesis project will be offered to and explored by EDIA. Due to the discussed improvement needs of Papyrus on the Substitute Selection step, I have given special attention to assessing the impact of resources that accommodate retrieving semantically fitting substitutes. For ranking substitutes on simplicity, I included an approach based on CEFR language level, just like EDIA's Papyrus.

The perspectives laid out above have resulted in the following research question:

"How do different approaches for Substitute Generation, Selection and Ranking compare in the context of building a Lexical Simplification system for the English language?"

In the pursuit of advancing Lexical Simplification for the English language, I implemented multiple models for the Substitute Generation, Selection, and Ranking steps. I based my design of these models on both existing literature and pioneering concepts. For each of the steps in the Lexical Simplification process, I evaluated the individual contributions of these models. This systematically executed modular approach resulted in a final model design that obtained highly competitive results on the English track of the TSAR-2022 Shared Task on Multilingual Lexical Simplification. To connect my thesis to EDIA's project with the Dutch government on assisting citizens who have a language deficiency, part of the Discussion chapter of this thesis is dedicated to analyzing how my methodology may be applied to the Dutch language.

1.3 Outline

This paper is structured as follows: chapter 2 discusses recent approaches with regard to the task of Lexical Simplification, excluding the most recent Shared Task on this subject, the TSAR-2022 Shared Task on Multilingual Lexical Simplification (Stajner et al., 2022; Saggion et al., 2022), to which chapter 3 is dedicated. This is the Shared Task this thesis project focuses on, for the English language. In chapter 4, the method with which I aim to answer my research question is addressed. Chapter 5 follows up with the results of the applied method and the answers to my research question, which I discuss in chapter 6. A summary of this thesis in chapter 7 concludes this paper.

Chapter 2

Related Work

In this chapter, I delve into literature associated with the historical development of Lexical Simplification techniques. Over the past few decades, this field has been subject to substantial research. Therefore, the sections below are limited to a selection of recent publications. The first section discusses related work on Complex Word Identification (CWI). Due to the fact that CWI is not the focus of this thesis project, this section is kept concise. The next section covers related work on Substitute Generation (SG) and Substitute Selection (SS), which are combined here as both steps are often intertwined, as introduced in section 1.1. The subsequent section provides related work on Substitute Ranking (SR). The last section in this chapter discusses a relatively new Lexical Simplification method, where similar, contextually valid, and simpler substitutes are generated from the start of the Lexical Simplification process.

This chapter does not cover the literature and methodologies that were presented in the TSAR-2022 Shared Task on Multilingual Lexical Simplification (Stajner et al., 2022; Saggion et al., 2022), which is the task that this thesis project focuses on. Chapter 3 is dedicated to describing this Shared Task, and includes a literature review of the participating systems.

2.1 Complex Word Identification (CWI)

The initial subtask in the Lexical Simplification process is Complex Word Identification (CWI), which aims to detect words or phrases in the text that may pose difficulty for the target audience. While some approaches bypass this step and treat all content words as potential candidates for simplification, incorporating a CWI component in the beginning of the Lexical Simplification process has proven beneficial. As unnecessary simplifications and associated errors are minimized, the CWI step enhances the effectiveness of Lexical Simplification systems (Paetzold and Specia, 2015).

CWI has been investigated in a variety of NLP projects. There have been two recent Shared Tasks on CWI: SemEval 2016 CWI (Paetzold and Specia, 2016b) for the English language, and BEA 2018 CWI Shared Task (Yimam et al., 2018) for multiple languages.

In the SemEval 2016 CWI Shared Task, words which might be perceived as difficult by non-native English speakers were predicted by the participating teams. Machine learning techniques such as decision trees, using a tree-like structure to make decisions, and ensemble methods, employing multiple models to improve overall performance, performed well on this task. Machine learning models that used word frequencies were

most effective in predicting word complexity: often-occurring words were perceived as less complex than words that appear only scarcely (Paetzold and Specia, 2016b).

The second Shared Task on CWI, BEA 2018 CWI Shared Task, was conducted for multiple languages: English, German, French, and Spanish. For this task, words which might be perceived as difficult by native as well as non-native English speakers were predicted. Conventional methods, predominantly reliant on word length and word frequency, remained the most effective in predicting word complexity, similar to the results of the SemEval 2016 CWI Shared Task. However, neural networks, designed to recognize patterns from data, gained improvement compared to the neural networks that had been applied to the SemEval 2016 CWI Shared Task (Yimam et al., 2018).

2.2 Substitute Generation (SG) and Substitute Selection (SS)

In the Lexical Simplification process, the Substitute Generation (SG) step may involve generating a wide range of candidates for replacing a complex word. After carrying out the SG step, only those candidates that can substitute the complex word, while preserving the grammatical structure and the contextually appropriate meaning of the complex word, are retained during the Substitute Selection (SS) step. Methods used to deploy the SG and SS steps are frequently intertwined and carried out simultaneously, depending on the model design.

As introduced in the first chapter, traditional methods generated substitutes with manually-designed linguistic vocabularies and parallel corpora. For example, Horn et al. (2014) used English Wikipedia aligned with Simple Wikipedia, introduced in the first chapter. However, Glavaš and Štajner (2015) argued that, while simple words are indeed prevalent in simplified text, they are also found in considerable quantities in standard text. Therefore, they used English Wikipedia and Gigaword 5[1], an extensive text corpus with newswire text data, for their candidate extraction. Paetzold and Specia (2016a) also implemented an unsupervised method, generating substitutes from a corpus of movie subtitles. They chose this source due to its effective capture of word familiarity, surpassing other corpora in this regard (Brysbaert and New, 2009). One year later, Paetzold and Specia (2017b) altered their strategy. They generated substitutes with a combination of parallel corpora, among which the Newsela corpus (Xu et al., 2015) that contains professionally created simplifications divided into five reading levels. They additionally generated substitutes with word embeddings, which was later followed by Gooding and Kochmar (2019). In the years before, Glavaš and Štajner (2015) and Paetzold and Specia (2016a) had already applied word embeddings, but only during the SS step.

The technique of word embeddings involves mapping tokens, i.e. words or parts of words known as subwords, to vectors in a continuous space. In the context of lexical simplification, cosine similarity scores are calculated between the vector of the complex word and the vectors of the other words in the continuous space. The cosine of the angle between both vectors can range from -1 to 1, where -1 means that two vectors are the opposite of each other in terms of similarity, zero means that there is no similarity, and 1 means that they, as they occupy the same space, are similar. Words with a cosine close to a value of 1 would most likely be chosen as candidates for simplification.

[1] `https://catalog.ldc.upenn.edu/LDC2011T07`, last accessed on 2023-08-14.

Yet, the word embeddings in those times had a severe drawback. Words can have different meanings, i.e., polysemous words. The embeddings were unable to distinguish these different meanings, as all possible interpretations of a word were incorporated into a single numerical vector. This vector was calculated disregarding the context of the word, thereby merging all conceivable meanings into one single representation. That this posed a problem is most evident for homonymous words, which are polysemous words that have *unrelated* meanings. An example of a homonym is the word *bank*, of which two common unrelated meanings are *financial institution* and *land alongside a river*, the difference of which can only be inferred from the context in which the word is situated. For example, the sentences "The Netherlands has many banks where you can get a loan" and "The Netherlands has many banks due to its many rivers" express different ways in which the word *bank* should be interpreted. Next to the clear need for context in case of homonymous words, it is also crucial to know the context of polysemous words that have *related* meanings. Although their relatedness can be due to the same conceptual origin, the word may have an entirely different meaning based on its contextual surroundings. For instance, the word *pool* in "They all took a swim in the pool" should be associated with a synonym related to a collection of water, whereas a synonym for the word *pool* in "They all contributed to the lottery pool" should be related to a collection of monetary fundings. Apart from the above use cases, words can be polysemous based on many other aspects, such as culture, figurative speech, and ambiguity in their appearance in syntactic constructions. Context is crucial for determining meaning in these cases. Therefore, using word embeddings to retrieve appropriate simplifications for polysemous words can only be successful if the context in which these words appear is taken into account.

Paetzold and Specia (2016a) had realized the limitations of embeddings without context, as reflected in their development of a corpus annotated with the following Part of Speech (PoS) tags: verbs, nouns, adverbs, and adjectives. Using these tags, they constructed what they called 'context-aware word embeddings'. This method distinguished different meanings of a word if these were based on its PoS tag. For example, the word *park* has different meanings depending on whether it is used as a noun or a verb. The noun implies an — often public — location, usually covered with grass and trees, where people can stroll and relax, whereas the verb refers to the action of positioning a vehicle somewhere. Words like these were thus assigned distinct representations depending on their specific PoS tags. Although this method was an improvement compared to the limitations of traditional embeddings, it still faced considerable constraints. While it was capable of disambiguating words based on their PoS tags, it did not capture the variations in meaning that a word can have within the same PoS. Take, for instance, the noun *bank*, the homonymous word discussed earlier. It could indicate *financial institution* or *land alongside a river*. In fact, all factors beyond a word's PoS tag that determine word meaning, among which the aspects mentioned earlier in this section, are not taken into consideration with this method. Therefore, despite the fact that the authors labeled these embeddings as 'context-aware', these did, in reality, not effectively capture the surrounding context within which words are used.

Gooding and Kochmar (2019) were able to make substantial progress in addressing the problem of polysemy. Recall from earlier in this section that they had based their substitute *generation* on a combination of linguistic resources and non-contextualized word embeddings. However, in their substitute *selection* method, they applied contex-

tualized embeddings, by using ELMo (Embeddings from Language Models) embeddings (Peters et al., 2018). ELMo is one of the first models that incorporated contextual information into its word embeddings. Unlike conventional word embeddings, ELMo uses embeddings based on the context in which words appear in very large corpora, which are called contextualized embeddings. The benefit of contextualized embeddings is that a word can be attributed different vectors that depend on the meaning of that word in different contexts, enabling a richer understanding of word semantics. This approach allows the model to capture polysemous words much better. The vectors are derived from a biLSTM (bidirectional Long Short-Term Memory), constituting a variation of recurrent neural networks (RNNs), equipped to process sequential data such as texts. Unlike the traditional standard LSTM (Long Short-Term Memory), which is unidirectional, i.e., it processes data from the beginning of the sequence to the end only, biLSTM also considers information from the opposite direction. It consists of two separate LSTMs: one that processes the sequence in its natural order (forward), whereas the other LSTM handles it in the reverse order (backward). By analyzing the sequence in both directions, a biLSTM can capture patterns that a unidirectional LSTM might overlook. Recall the earlier-mentioned sentences "The Netherlands has many banks where you can get a loan" and "The Netherlands has many banks due to its many rivers", where crucial information for determining the meaning of the homonymous word *banks* comes after that word. A biLSTM like ELMo could be able to capture these different meanings, whereas a unidirectional LSTM could not. A widely recognized model that took ELMo's bidirectionality to a new level is BERT (Devlin et al., 2019). Section 2.4 discusses the application of BERT for the task of Lexical Simplification.

Regarding the morphological features of the substitute candidates, both Horn et al. (2014) and Paetzold and Specia (2016a, 2017b) limited the generation of these substitutes to candidates with the same PoS tag as the complex word, thereby aiming to maintain grammatical correctness in the resulting sentences. This is because words with different PoS tags often cannot be substituted without jeopardizing the grammatical correctness or the intended meaning of the sentence. Consider the sentence "The cat sat on the mat" as an example. If the word *sat*, categorized with the PoS tag 'verb', is substituted with the noun *seat*, the resulting sentence "The cat seat on the mat" would not be grammatical. Horn et al. (2014) further excluded candidates tagged as proper nouns. Proper nouns refer to specific entities such as persons, locations, or organizations. These can typically not be used as a substitute without changing the meaning of the word to be substituted. Take a sentence like "I will visit Lissabon this summer", for instance. The proper noun *Lissabon* could be replaced by another proper noun, e.g., *Portugal*, but it will lose crucial information, as it is not known which city in Portugal will be visited. Furthermore, they (Horn et al., 2014) enriched their substitute list with morphological variants of both the complex word and its substitute candidates. By adding these different word forms, they aimed at making their system more flexible and robust so that it could generalize better towards different usage scenarios.

In their respective studies, Glavaš and Štajner (2015) as well as Paetzold and Specia (2016a) implemented a method that excludes morphological derivations of the complex word from the Substitute Generation process. Their approach might stem from the idea that if a substitute simply represents another version of the complex word, the chances of successfully simplifying the complex word would be reduced. In addition, morphological derivations may convey divergent meanings. The most obvious illustration of

this is negation. A (compound) word can be the opposite of another word just by the incorporation of a negation prefix, such as 'un', 'in', and 'ir'. Word pairs like *clear* vs. *unclear*, *appropriate* vs. *inappropriate*, and *relevant* vs. *irrelevant* are examples of this. Given that these words look much alike and can be used in similar contexts as well, it may present a challenge for NLP systems to distinguish their distinct meanings. Therefore, the strategy of excluding morphological derivations of the complex word as part of the SG step can be beneficial for the success of further steps in the Lexical Simplification process.

Glavaš and Štajner (2015) adapted substitutes to the morphological form associated with the PoS tag of the complex word, to ensure grammatical consistency of the substitute with the complex word. For example, if we take a sentence like "The artist wants to *revolutionize* the art world", an ungrammatical substitute candidate for the complex word, the infinitive *revolutionize*, is the third person singular *transforms*. However, when adapted to the infinitive form, in this case, *transform*, the modified word has become a suitable substitute.

Finally, Gooding and Kochmar (2019) removed candidates that would rarely occur in the context of the preceding and following word. They obtained these results by calculating bigram frequencies of the generated substitute candidates and the previous and following word. They derived their benchmark frequencies from the COCA (Corpus of Contemporary American English) corpus[2] (Davies, 2009), and removed candidates for which the frequency in the context of that corpus was equivalent to 0.

2.3 Substitute Ranking (SR)

After the substitute candidates have been generated in the SG step, and after only those that preserve the grammatical structure and have a contextually appropriate meaning are retained in the SS step, they undergo the Substitute Ranking (SR) step, involving ranking on simplicity. Providing that only semantically similar substitutes have been selected during the SS step, the context of the original sentence is not the main issue anymore in the SR phase.

Traditional methods to carry out the SR step were often based on word frequency measures, from the general knowledge that most frequent words are usually simpler (Horn et al., 2014; Glavaš and Štajner, 2015; Paetzold and Specia, 2016a). For the purpose of ranking on simplicity, Pavlick and Callison-Burch (2016) created the Simple Paraphrase Database, which is contained within the PPDB (Paraphrase Database) developed by Ganitkevitch et al. (2013), a large-scale lexical resource and collection of paraphrases constructed by automated processes. They (Pavlick and Callison-Burch, 2016) created the Simple Paraphrase Database by adapting the PPDB for text simplification through manual simplicity annotation. Paetzold and Specia (2017b) addressed reader simplification needs by using the Newsela corpus (Xu et al., 2015) that contains professionally created simplifications divided into five reading levels. Gooding and Kochmar (2019) performed the SR step by using word complexity information from the CWI dataset of Yimam et al. (2017), which they applied to every substitute in the context of the original sentence, resulting in contextual simplicity scores.

[2] `https://www.english-corpora.org/coca/`, last accessed on 2023-08-14.

2.4 BERT: a Transformative Shift in Lexical Simplification

Until 2019, English[3] Lexical Simplification systems did not employ contextualized word embeddings, or not until after the SG step had been executed — as introduced in section 2.2, where Gooding and Kochmar (2019) had started using contextualized embeddings from the SS step onward. As these systems had not taken the context of the complex word *in its original sentence* into account during the SG step, they were likely to generate an abundance of irrelevant or erroneous substitutes, potentially disrupting the results in the next steps of the Lexical Simplification process: if there are no simpler alternatives among the substitute candidates for a complex word, then the selection and ranking steps in the Lexical Simplification process lose their purpose (Qiang et al., 2021). Another drawback of considering the context of the complex word only after the SG step is that substitutes that would have been appropriate based on the unique context of that sentence may be overlooked. Since only the results obtained in the SG step will be considered for the SS and SR steps, overlooked substitutes during the SG step are subsequently also disregarded in the ultimate simplification results.

These limitations were addressed by the groundbreaking capabilities of a novel model introduced in 2019: BERT (Bidirectional Encoder Representations from Transformers), developed by Devlin et al. (2019). Due to its unique characteristics, BERT achieved remarkable success in various NLP tasks, among which in Lexical Simplification. BERT has been pre-trained on text data contained in the BookCorpus, introduced by Zhu et al. (2015), and English Wikipedia, discussed in the opening chapter. These sources provided a substantial volume of text data, forming a considerably large training dataset. As BERT can be applied to raw text, it is widely scalable to Lexical Simplification of many languages. Like ELMo, the RNN variant introduced in section 2.2, BERT learns contextual representations for words in a bidirectional way, i.e., from the direction of both preceding and following words. Yet, the Transformer architecture (Vaswani et al., 2017) upon which BERT is based, elevated ELMo's bidirectionality to a new standard by doing this simultaneously instead of in (step-by-step) sequences. This innovative approach of considering all tokens at the same time leads to more accurate understanding of the context in which words appear, as the model is capable of capturing meanings of words influenced by other words that appear much earlier or later in the text. Another noteworthy feature is that BERT is a masked language model (MLM), able to replace certain words in the input text with [MASK] tokens, after which it can predict the original words that were replaced by these [MASK] tokens, based on the surrounding context. This masking property supports the generation of several contextually fitting words for each masked token, a characteristic that is particularly beneficial for the SG step in the Lexical Simplification process. After masking a complex word, BERT uses the contextual information of the sentence to make predictions for the masked words. The top predictions from this process can be considered as potential substitutes, offering context-aware alternatives for the complex word.

In conclusion, BERT's unique ability to grasp the semantic role of words within their specific context, along with its competence to generate context-aware substitutes directly from the initial SG step onward, has been a milestone in transforming the process of Lexical Simplification.

[3]TUNER (Ferrés et al., 2017) applied contextualized word embeddings during the SG step in 2017 for Spanish, which will be discussed in section 3.7.

2.5 From BERT to LSBert

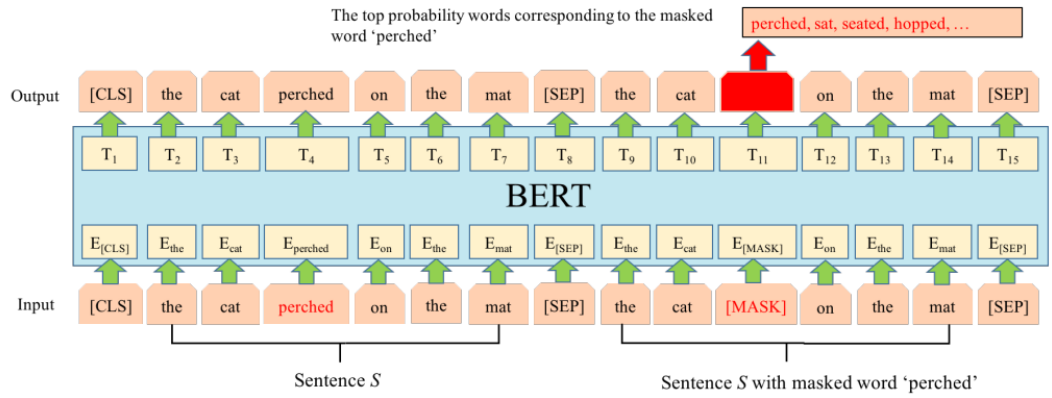


Figure 2.1: Substitute Generation with LSBert, taken from Qiang et al. (2021).

Qiang et al. (2021) developed a Lexical Simplification model, LSBert, that they had based on BERT's architecture, bidirectional approach, and masking functionality. Like BERT, LSBert is capable of retrieving context-aware substitutes directly from the start of the Substitute Generation step. From the generated substitutes, it selects the most semantically similar and the most simple alternatives, which makes it an all-in-one approach for Lexical Simplification.

LSBert, for which Bert-Large, Uncased, WWM (Whole Word Masking)[4] was chosen for the SG step, has adapted BERT's [MASK] functionality by adding the original sentence in which the complex word is not masked. To prevent only considering the context around the complex word without assessing the meaning of the complex word itself, the original sentence including the complex word is added to the sentence in which the complex word is masked, together forming a sentence pair. To compensate for the fact that the sentence pair generates double contextual information, 50% of the words in the original sentence excluding the complex word is randomly masked. Then, the sentence pair is fed to the model. Thus, from the start of the SG step, the model is enabled to generate contextually fitting substitute candidates that are also semantically close to the meaning of the complex word. At the end of the SG step, morphological derivations of the complex word are removed from the candidate list.

LSBert's Substitute Generation process is visualized in figure 2.1. The tokens [MASK], [CLS], and [SEP] are three unique tokens. While [MASK] masks a specific word, [CLS] can be found at the beginning of each piece of text, in this case, a sentence. Finally, [SEP] is a separator token that, in this case, is added between the end of a sentence and the beginning of a new sentence. For the example sentence "The cat perched on the mat", the top three candidates for the complex word **perched** would be *sat*, *seated*, and *hopped*. These candidates comply with the grammatical structure of the word **perched** in the original sentence and also fit in its context. However, they are not all equally semantically similar nor equally simpler alternatives to the complex word **perched**. The authors (Qiang et al., 2021) aim to solve this during the subsequent SS and SR steps, which they combined into one step. In that step, they use the averaged result of five features, where the first three are related to semantic fit, and the last two to simplification:

[4] `https://huggingface.co/bert-large-uncased-whole-word-masking`, last accessed: 2023-08-14.

1. BERT's cross-entropy loss of the masked word. During pre-training, BERT learns to predict masked tokens by using the context provided by the unmasked tokens. This context is defined as a symmetrical range with a size of five tokens centered around the masked complex word. How well BERT predicts masked tokens in this context is computed by cross-entropy loss (Devlin et al., 2019), a measure that determines how well the predicted probability of an event matches the actual outcome. A lower cross-entropy loss corresponds to a smaller deviation from the actual outcome, and, in the context of Lexical Simplification, results in a higher ranking of the substitute as a valid alternative for the complex word.

2. BERT's prediction order, referring to BERT's probability distribution on new input sentences after it has been pre-trained. BERT bases its prediction of the masked token on the surrounding context (Devlin et al., 2019). LSBert uses these predicted probabilities to rank substitutes for a complex word. The higher the probability predicted by BERT, the higher LSBert ranks the substitute.

3. Word semantic similarity, using the vector representations of the FastText (Bojanowski et al., 2017) word embedding model that maps words broken down into subword character n-grams to vectors in a continuous space, and calculates the cosine similarity between the vectors. The advantage of using subword character n-grams is that vectors for rare or previously unseen words can be provided as well. The more similar the vectors, the higher LSBert ranks the substitute.

4. Word frequencies obtained from large text corpora, based on a common understanding that the most frequently used words tend to be simpler, as discussed in section 2.3. The more frequent the word, the higher LSBert ranks the substitute.

5. The paraphrase database PPDB (Ganitkevitch et al., 2013), discussed in section 2.3. If the candidate occurs in the PPDB database in a pair together with the complex word, LSBert ranks it higher than if both words do not occur together.

For each substitute, its scores on the above-mentioned five features are averaged. Subsequently, the substitutes are ordered on their respective averaged scores.

Owing to BERT's architecture, LSBert achieves two of the three main objectives outlined in the opening chapter: 1) contextual understanding — in this case, during the SG step — and 2) scalability, by processing raw text without relying on hand-crafted corpora. Yet, LSBert does not meet the third objective of adapting the substitutes to readers with different simplification needs. Moreover, the FastText word embeddings, used to calculate semantic similarity during the combined SS-SR step, are uncontextualized. As stated in section 2.2, uncontextualized embeddings consolidate all meanings of a word into one static vector. Although LSBert uses uncontextualized embeddings only for substitutes that have already been generated in the context of the sentence, these embeddings might yet prove counterproductive for polysemous substitutes. For example, a contextually ideal substitute might obtain lower embedding scores because of the generalized representation of its diverse senses in uncontextualized embeddings. Consequently, such substitute might not be prioritized when it should be.

Yet, LSBert proved to be the most successful model for Lexical Simplification in 2021 (Qiang et al., 2021), and was defined as one of the two English baseline models for the TSAR-2022 Shared Task on Multilingual Lexical Simplification (Stajner et al., 2022; Saggion et al., 2022). Chapter 3 describes this Shared Task with regard to its English track, which is the task that this thesis project concentrates on.

Chapter 3

Task Description

The TSAR-2022 Shared Task on Multilingual Lexical Simplification (Stajner et al., 2022; Saggion et al., 2022) was the first Shared Task on Multilingual Lexical Simplification. Participants carried out steps 2 through 4 of the Lexical Simplification process laid out in section 1.1. The main goal was to find out how complex words can be transformed into simpler alternatives while keeping the meaning of these complex words in the context of the original sentence in place. The Shared Task featured tracks in English, Spanish, and (Brazilian) Portuguese, for which the instruction was equal: "Given a sentence/context and one target (complex) word in it, provide substitutes for the target word that would make the sentence easier to understand. It was allowed to submit up to ten substitutes, ordered from the best to the least fitting/simple one. Ties were not allowed" (Saggion et al., 2022, p.3).

The authors did not mention what target audience they had in mind regarding 'easier to understand' and 'from the best to the least fitting/simple one' (Saggion et al., 2022, p.3). Moreover, a definition of 'simpler' was not given by Stajner et al. (2022) or Saggion et al. (2022). The implications of the ambiguity of these instructions will be addressed in section 6.2.

The subsequent sections offer an exploration of the English track in this Shared Task. I discuss the used dataset and its associated constraints, along with the employed annotation guidelines. Next, I explain the evaluation metrics to assess participating systems. Subsequently, I examine the baseline models which served a the competitive benchmark for the participating systems in this Shared Task. Lastly, I cover how participating systems solved the task, followed by a review of their results.

3.1 Data Collection

Recall from section 1.1 that the identification of complex words (CWI) is the initial step in the Lexical Simplification process. As denoted in section 2.1, this step enhances the effectiveness of these systems by minimizing superfluous simplifications and related errors.

For this purpose, the TSAR-2022 Shared Task on Multilingual Lexical Simplification evaluated the submitted systems on a dataset in which certain words had previously been identified as complex. This concerned the data Yimam et al. (2017) had used for the 2018 CWI for Multilingual Lexical Simplification Shared Task (Yimam et al., 2018), introduced in section 2.1. For the English language, this data consisted of three writing genres: News (articles authored by professionals), WikiNews (articles

written by amateurs), and articles from Wikipedia. Information about the specific topics that the texts in the corpora are about is not readily available. The writers of the text corpora consisted of a mix of experts and amateurs in news writing. A total of 183 annotators participated in this task, crowdsourced by the Amazon Mechanical Turk (MTurk) crowdsourcing platform. The annotators encompassed native (134) native and non-native (49) speakers, of which the non-native speakers reported different proficiency levels (beginner, intermediate, advanced). Whether these annotators possessed supplementary linguistic skills for the annotation task was not specified. Demographic properties such as age, gender, race, ethnicity, or socioeconomic status were not disclosed either. Collectively, the annotators labeled 34,897 words based on their perception of the complexity of these words. Each word was annotated by at least ten native and at least ten non-native speakers. The result show significant differences in perceived word complexity between native and non-native speakers, indicating distinct simplification needs for these groups (Yimam et al., 2017, 2018), as introduced in the opening chapter.

For the TSAR-2022 Shared Task (Stajner et al., 2022; Saggion et al., 2022), the complex words were selected from all three text genres represented in the 2018 CWI dataset discussed above. Instances that had been marked in the CWI Shared Task 2018 as complex by at least five of the ten native English annotators were first selected for the TSAR-2022 dataset. After removing duplicates, approximately 400 instances were chosen by a native English speaker, based on personal judgment whether at least one simpler word for that instance in that context could be found. In each sentence, one complex word was selected. According to Stajner et al. (2022), this choice was made for two reasons. First, to avoid complications in the validation of annotations, as replacing multiple words within the same sentence could produce sentences that feel unnatural, due to the subtle variance in meanings between the replacements and the original complex words. Second, simplifying more than one word per sentence would conflict with the current setup of most state-of-the-art simplification systems that simplify one word per sentence per iteration. If more than one word had to be simplified in a sentence, the context would alter slightly each iteration, due to the subtle meaning change caused by the simplification of a complex word from the previous round.

The approximately 400 words marked as complex and the sentences in which they occurred were supplied for annotation via the Amazon Mechanical Turk (MTurk) crowdsourcing platform. No demographic information was collected for this annotation task. Only native English annotators were requested to perform this task, although this was not individually verified. Annotators received annotation guidelines for guidance on executing this task. In short, they had to provide one simpler synonym for each marked complex word, while preserving the meaning of the original sentence. These annotation guidelines are further discussed in the subsequent section. Each instance was annotated by 25 annotators and reviewed by at least one native English-speaking computational linguist. Where necessary, affix changes were applied to the annotations to fit the context grammatically. Instances with unsuitable annotations — i.e., where the annotation guidelines had not been followed — were removed. In those cases, further annotations were requested so that each instance had 25 annotations in the end. Instances that did not get good suggestions were removed, resulting in 383 final instances[1].

[1]The Shared Task datasets are available at `https://www.github.com/LaSTUS-TALN-UPF/TSAR-2022-Shared-Task`, last accessed on 2023-08-14.

3.2 Annotation Guidelines

One significant step in preparing data for NLP tasks is annotation, which involves labeling language data by humans. The foundation for this process is formed by annotation guidelines, which are vital for the success of NLP tasks, irrespective of whether the annotations are only used for system evaluation. Unclear guidelines can result in inconsistent annotations and unreliable evaluations. Moreover, they can make it challenging to identify system weaknesses, given the complexity in distinguishing whether errors stem from system issues or ambiguous annotations. Moreover, unclear guidelines can lead to misinterpretation, potentially fostering individual biases, which can be harmful especially in critical sectors such as healthcare.

For the TSAR-2022 Shared Task on Multilingual Lexical Simplification, Annotation Guidelines[2] (Stajner et al., 2022) were provided to guide annotators through the task. An extract is displayed in figure 3.1. The full Annotation Guidelines form that was provided to the annotators is supplied in appendix A.

Parts of these guidelines can be understood in different ways, which potentially could have introduced misinterpretation and subsequent personal biases.

For example, the guidelines did not contain a definition of 'simpler' in the given context of 'a simpler word', nor did they explain their phrase 'easier to understand' (figure 3.1). Moreover, the audience to which this should apply was not stated. Would the word to be simplified have to be easier to understand by the specific annotator only, or would the annotator have to find simplifications that in this person's opinion, would be easier to understand by other readers? If so, to which subgroup of these readers would this apply? As highlighted in the opening chapter, simplification needs seem to be different depending on the target audience.

Štajner et al.

---

1 APPENDIX I: INSTRUCTIONS FOR ANNOTATORS

Below are **N** sentences in English/Spanish/Portuguese, in each sentence there is a word marked in bold. Your task is to write, in the space below each sentence, single word that has the same meaning as the one marked, but is easier to understand. For example, in the sentence "At the same time, the rate of decline against the dollar was **attenuated**" the word **attenuated** could be replaced by the easier-to-understand word *decreased*. Write the replacement so that the replacement is valid in the given context. In our example, *decreased* is correct while decrease would not be correct. In that case that it is not possible to replace with a single word, then you can use a more complex substitution. For example in the sentence "The dresses were **Iranian**", the word **Iranian** could be replaced by "**from Iran**". Replacements that involve a gender change with respect to the marked word are also allowed in Spanish and Portuguese (Note that this is not applicable in English).

Note 1: If you cannot find a simpler word then you must write the same complex word in the answer area.

Note 2: You are allowed to use all kinds of lexical reference resources such as dictionaries, thesaurus, etc., whether books or online, to do the task.

Figure 3.1: Extract of Annotation Guidelines for English track of TSAR-2022 Shared Task, taken from Stajner et al. (2022).

[2] `https://www.frontiersin.org/articles/10.3389/frai.2022.991242/full#supplementary-material`, last accessed on 2023-08-14.

Furthermore, the following sentence in the guidelines, as shown in figure 3.1, may be perceived as ambiguous:

"In that case that it is not possible to replace with a single word, then you can use a more complex substitution. For example, in the sentence "The dresses were **Iranian**", the word **Iranian** could be replaced by **from Iran.**"

The phrase 'a more complex substitution' had not been explained. 'More complex' might erroneously be perceived as more complex than the complex word, and does furthermore not resonate in the accompanying example of 'more complex': the phrase **from Iran**, which is essentially a simplification consisting of two words. This specific example also made clear that annotators were allowed to use more than one word as a substitute, which was also confirmed in one (Stajner et al., 2022) of the Shared Task's papers. I will cover the implications of multi-word annotations in the context of this Shared Task in section 6.1.2.

In summary, the absence of definitions of what should be understood by 'simpler' and 'easier to read', together with the ambiguous example regarding 'more complex', could have introduced individual biases to the annotation task. I will reflect on the impact associated with personal biases regarding this particular Shared Task in the subsequent section, as well as in sections 6.2 and 6.5.

3.3 Data Limitations

Stajner et al. (2022) discuss two limitations of the data. First, as all instances are taken from news sites and Wikipedia, reliable lexical simplification results are limited to these specific genres only. Second, the simpler synonyms are proposed by crowdsourced contributors instead of experts. With 25 annotators for each instance, this issue is somewhat diminished by the ranking of the suggested replacements according to how often they had been proposed. However, a linguist would be required to verify this, as the synonyms had only been professionally checked on grammar and meaning, but not on their perceived simplicity (Stajner et al., 2022).

In addition to the above constraints and the ambiguities in the Annotation Guidelines reviewed in the previous section, there are further concerns regarding the collected data. The roughly 400 complex words had been selected by one native English speaker. This person had chosen these particular complex words based on whether at least one simpler word could replace the original complex word within its context. The linguistic expertise of this individual is not stated, introducing uncertainty about potential vulnerabilities. This includes the risk of personal biases and lack of linguistic precision in the selection process, which could consequently compromise the representativeness of the data.

An additional issue linked to the collected data is the absence of alignment between the expertise of the annotators and the topics of the texts. The lack of correspondence could inhibit their understanding when tasked with replacing complex words with simpler alternatives. Moreover, the fact that the annotation task was paid might have encouraged a focus on financial gain rather than a genuine engagement with the task. This could have affected the overall quality of the annotations.

The potential compromise on annotation quality became more apparent when I examined the annotated substitutes in the trial set, which is a small subset of the dataset

discussed in section 3.1. Nearly all annotations demonstrated a semantic relation to the complex word. However, a considerable portion of them did not appear to align with the primary objective of the Shared Task, which is to offer simpler alternatives. This observation was also consistent with the most frequently suggested annotation provided with that trial set, illustrated in table 3.1.

Complex Word	Most Frequently Suggested Annotation	Occurrences
compulsory	**mandatory**	11
instilled	**infused**, introduced	3
maniacs	fanatics	5
observers	watchers	8
shrapnel	bullet	4
disguised	**concealed**, dressed	4
offshoot	branch	6
symphonic	musical	12
deploy	send	5
authorities	officials	11

Table 3.1: Complex words in trial set and their most frequently suggested annotation. Annotations marked in bold do not seem simpler.

Particularly, the annotations highlighted in bold do not seem simpler than the complex word. I examined these assumptions after I had created and tested my models on the trial set. The results will be detailed in section 4.4.3.

3.4 Format of Annotated Dataset

As introduced in section 3.1, the English dataset contains 383 instances. These instances are represented as rows in a TSV (Tab Separated Values) file. The columns in the file represent the sentence containing the complex word (the first column), the complex word (the second column), and the columns with annotations. As introduced in section 3.1, 25 annotations per complex word were provided. Therefore, the file spans 27 columns[3], consisting of the sentence, the complex word, and the 25 annotations.

The annotations are systematically arranged in descending order, based on the frequency of their suggestions by annotators. For example, an annotation appearing six times would occupy the first six columns behind the initial two with the sentence and complex word. This most frequent annotation was considered the simplest alternative to the complex word. Successive columns accommodate the next highest frequent annotations, and so forth, concluding with the least frequent annotations. Notably, annotations with equal frequencies seem to have been ordered randomly in relation to each other.

The above dataset was divided into a trial and a test set, consisting of 10 and 373 instances respectively. For the participants in the Shared Task, the test set was provided with the sentence and the complex word only, whereas the trial set was additionally provided with the annotations. Consequently, participants could try their systems on a small gold-labeled (i.e., human-annotated) dataset before submitting their final outputs on the test set.

[3]Occasionally, a 28th column is filled with an annotation, marking the instances where 26 annotations were provided.

Table 3.2 presents an example of the gold-labeled trial set. For overview purposes, the Gold Labels column contains all unique annotations with their respective frequencies. However, as mentioned earlier in this section, the TSV file contains all 25 gold labels in separate columns, ordered by their frequency.

Sentence	Complex Word	Gold Labels
A local witness said a separate group of attackers disguised in burqas (the head-to-toe robes worn by conservative Afghan women) then tried to storm the compound.	disguised	concealed:4, dressed:4, hidden:3, camouflaged:2, changed:2, covered:2, disguised:2, masked:2, unrecognizable:2, converted:1, impersonated:1

Table 3.2: Example from annotated trial set, taken from Saggion et al. (2022, p.5).

Annotations equal to the complex word[4] were disregarded from the final evaluation results. For the example shown in table 3.2, the complex word **disguised** had been annotated two times as a proposed substitute for itself. Consequently, these two annotations were considered invalid and were thus not used when evaluating the submitted models.

Across the entire English dataset, the number of unique annotations for a complex word varied from two to 22, with an average of 10.55 (Saggion et al., 2022). I will discuss the influence of this average on my design decisions in section 4.3.

3.5 Submission Format for Predicted Substitutes

As mentioned in the beginning of this chapter, participating systems in this Shared Task could submit up to ten substitutes for each complex word. This implies that less than ten substitutes could be submitted as well. Furthermore, multi-word substitutes were also possible, as pointed out in section 3.2. The predicted substitutes should be ranked from "the best to the least fitting/simple one" (Saggion et al., 2022, p.3).

As outlined in the preceding section, the test set of 373 instances was used for submission of the system outputs. The required submission format for the predicted substitutes is a TSV file with 373 rows, with the initial two columns featuring the example sentence and the complex word, followed by up to ten columns for the predicted substitutes, each of the predictions occupying a separate column.

Table 3.3 visualizes the submission format for an example from the trial set with fictive predictions for the complex word. For overview purposes, the Fictive Predictions column contains all predicted substitutes, instead of each of them in a separate column.

Sentence	Complex Word	Fictive Predictions
A local witness said a separate group of attackers **disguised** in burqas (the head-to-toe robes worn by conservative Afghan women) then tried to storm the compound.	disguised	dressed, draped, masked, wrapped, cloaked, concealed, covered, hidden, decorated, camouflaged

Table 3.3: Submission format: example from trial set with fictive predictions, ranked from "the best to the least fitting/simple one" (Saggion et al., 2022, p.3).

[4]Annotators had been instructed to enter the complex word as their annotation in case they would not know a simpler word, as shown in the Annotation Guidelines in figure 3.1.

3.6 Evaluation Metrics

From the predicted substitutes in the submitted outputs, the evaluation script filtered out substitutes equal to the complex word, as well repeated substitutes. Then, the following evaluation metrics were calculated (Stajner et al., 2022; Saggion et al., 2022):

- ACC@K@top1. This metric represents the proportion of instances for which a minimum of one of the top-K ranked predicted substitutes equals the most often suggested gold label (K = 1, 2, 3). For example, for K = 3, it indicates the proportion of instances for which minimally one of the top three predictions aligns with the most often suggested gold label.

- MAP@K (Mean Average Precision@K). This measurement concerns the proportion of instances for which all top-K ranked predicted substitutes appear in the list with gold labels for that instance (K = 1, 3, 5, 10). For example, for K = 3, it represents the proportion of instances for which all top three predicted substitutes can be found in the list with gold labels.

- Potential@K. This measure reflect the proportion of instances for which a minimum of one of the top-K ranked predictions appears in the list with gold labels for that instance. (K = 1, 3, 5, 10). For example, for K = 3, it indicates the proportion of instances for which minimally one of the top three predicted substitutes can be found in the list with gold labels.

As MAP@1 and Potential@1 are factually the same as per their definitions (also called ACC@1 (Saggion et al., 2022)), in total ten different metrics were assessed: ACC@1, ACC@1@top1, ACC@2@top1, ACC@3top1, MAP@3, MAP@5, MAP@10, Potential@3, Potential@5, Potential@10.

The fact that the gold labels could vary from 2 to 22 different simplifications with an average of 10.55, as laid out in section 3.4, implies that for the MAP@K metrics, if K = 3, 5, or 10, there are not always 3, 5, or 10 different gold labels to evaluate the submitted outputs on. For example, if K=5, for MAP@5 to evaluate as positive on a particular instance, there should also be five unique gold labels for this instance. The number of different gold labels do not influence the Potential@K and ACC@K metrics in that respect, as these metrics only assess *at least one* of the top K ranked predicted substitutes, as opposed to *all* top-K ranked predicted substitutes for the MAP related metrics.

As concluded by Saggion et al. (2022), it is important to realize that the methodologies used for evaluation of the simplification submissions operate under the assumption of a singular optimal simplification universally applicable to all users. The simplification identified as 'the simplest' is derived from a broad spectrum of annotators and corresponds to the most frequently suggested gold label. As pointed out in the introductory chapter, users exhibit diverse simplification requirements based on their individual contexts and needs. Following this perspective, an efficient simplification system should produce an output tailored to the distinct requirements of the individual user, underlining the importance of personalization in the simplification process.

3.7 Baseline Models

Two baseline models were used for benchmarking the participating systems for the English language, TSAR-TUNER and TSAR-LSBert.

TSAR-TUNER, a variant of TUNER (Ferrés et al., 2017), a Lexical Simplification system designed for the Spanish language, is a non-neural lexical simplification system. For the TSAR-2022 Shared Task, it has been adapted to English. TSAR-TUNER (Stajner et al., 2022, p.11-12) performs four sequential tasks: Sentence Analysis, Word Sense Disambiguation (WSD), Synonym Ranking, and Morphological Generation. The Sentence Analysis task executes tokenization, sentence splitting, part-of-speech (PoS) tagging, lemmatization, and Named Entity Recognition. The WSD algorithm uses a word vector model which it generated from a text corpus, alongside a context vector generated by the words in immediate vicinity of the complex word. By calculating the cosine distance between the context vector and the word vector, the algorithm determines the similarity between them. Subsequently, it selects the sense with the smallest cosine distance as the most optimal semantically fitting word. The Synonym Ranking task ranks synonyms based on frequencies in Simple English Wikipedia, a resource discussed in the first chapter. Finally, the Morphological Generation task adapts the selected synonyms to the correct grammatical form of the complex word.

The second baseline model, TSAR-LSBert, is fully based on LSBert (Qiang et al., 2021), the model discussed in section 2.5. It has only been adapted for the Shared Task to match the submission format for the predicted substitutes outlined in section 3.5.

3.8 Participating Systems

In total, 13 teams submitted 31 systems. Each team was allowed to submit three systems, resulting in, on average, 2.4 systems per team. This section is limited to discussing the ten teams that had submitted a research paper (the teams that had not submitted a paper are all but one — i.e., CL Lab PICT, ranked ninth — listed in the lowest quartile of the results). Table 3.4 provides an overview of the ten teams, references to their respective research papers, and their highest-ranked contribution on the ACC@1 metric, which is the performance measure that the results on this Shared Task are sorted on. Further results of these teams are covered in section 3.9.

Team	Reference	Highest Rank
UniHD	Aumiller and Gertz (2022)	1
MANTIS	Li et al. (2022)	3
UoM&MMU	Vásquez-Rodríguez et al. (2022)	4
TSAR-LSBert	Qiang et al. (2021); baseline for Shared Task	5
RCML	Aleksandrova and Brochu Dufour (2022)	6
GMU-WLV	North et al. (2022)	8
TeamPN	Katyal and Rajpoot (2022)	11
PolyU-CBS	Chersoni and Hsu (2022)	15
PresiUniv	Whistely et al. (2022)	17
CILS	Seneviratne et al. (2022b)	19
CENTAL	Wilkens et al. (2022)	23
TSAR-TUNER	Ferrés et al. (2017); baseline for Shared Task	24

Table 3.4: Participating teams and their highest-ranked contributions. The rankings of the two baseline models, TSAR-LSBert and TSAR-TUNER, are also included.

The subsequent sections detail the highest-performing contribution from each team. Their strategies to attain these scores are outlined for every step in the Lexical Simplification process.

3.8.1 Substitute Generation (SG)

For the process of generating substitutes for the complex words, all ten teams that had submitted a research paper utilized transformer-based (Vaswani et al., 2017) models, discussed in section 2.4.

Seven of them (PresiUniv, UoM&MMU, PolyU-CBS, CENTAL, TeamPN, MANTIS, and GMU-WLV) relied on a Masked Language Model (MLM) to accomplish the SG step. Two of these seven teams, MANTIS and GMU-WLV, added the original sentence including the complex word to the sentence in which the complex word had been masked, and fed the sentence pair into their MLM, similar to LSBert discussed in section 2.5. However, they had not incorporated LSBert's strategy to randomly mask 50% of the words in the original sentence excluding the complex word. Conversely, CENTAL and TeamPN merged the substitutes obtained by their MLMs with substitutes retrieved from a variety of linguistic databases. UoM&MMU crafted prompting templates (referred to as 'prompts') to request their MLM to generate simplification candidates. They had first fine-tuned their MLM with a variety of lexical simplification corpora, among which a dataset based on CEFR levels (Uchida et al., 2018).

As opposed to the seven teams that had chosen MLMs, three teams opted for a different transformer-based model. CILS and RCML used XLNet, developed by Yang et al. (2019). This model considers all possible permutations of the input sequence and bases its predictions for each token on these permutations. RCML additionally employed the LexSubGen (Arefyev et al., 2020) database to generate substitutes for complex words in the context of the original sentence. Just like UoM&MMU, UniHD prompted their model with templates to request simplification candidates, but instead of an MLM they used GPT-3 (Generative Pretrained Transformers, version 3 (Brown et al., 2020)). GPT-based models deploy the transformer architecture as their foundation but have expanded it specifically for generative tasks. Their numerous transformer layers help effectively capture complex relationships, thereby generating contextually appropriate responses. UniHD had prompted the GPT-3 model with six different templates. These templates varied in the level of context they provided, depending on whether the sentence was given or only the complex word.

3.8.2 Substitute Selection (SS)

Recall from section 2.2 that the SS step involves selecting only those candidates that can substitute the complex word while preserving the grammatical structure and the contextually appropriate meaning of the complex word.

With regard to adherence to the grammatical structure of the complex word, CENTAL and TeamPN adapted the candidates to the morpho-syntactic form of the complex word. CENTAL also removed those for which this form did not exist. MANTIS eliminated morphological derivations of the complex word from the substitute candidate list. UoM&MMU excluded non-existing words and duplicate candidates. UniHD eliminated duplicate candidates and expressions consisting of more than one word. Furthermore, they excluded prepositions from generated infinitives, such as *to* in *to deploy*. PresiUniv filtered out words with different PoS tags than the complex word. The other four

teams (CILS, PolyU-CBS, GMU-WLV, and RCML) did not report actions related to preserving grammatical structure in their best-scoring models.

To select the generated substitutes on semantic similarity, various methods were applied. The most popular method concerned application of cosine similarity scores between the embeddings of the complex word and the embeddings of each of its substitute candidates, in some cases combined with the base probability from the results retrieved in the SG step. Cosine similarity scores were used by CILS, PresiUniv, PolyU-CBS, and RCML. They all leveraged contextualized embeddings to obtain their similarity scores, except for PresiUniv who used conventional embeddings. RCML applied the relatively new BERTScore (Zhang et al., 2020) for their contextualized embeddings. BERTScore calculates the similarity of the sentence containing the complex word with each of the sentences containing a different substitute candidate for that complex word.

Application of linguistic vocabularies and parallel corpora was also part of the approaches used for semantic similarity. CILS used WordNet (Fellbaum, 1998) to calculate WordNet similarity scores (Seneviratne et al., 2022a) alongside contextual embedding scores. UoM&MMU leveraged Wordnet to remove antonyms of the complex word from the substitute list generated by their prompt-based MLM. MANTIS applied LSBert's feature distribution as described in section 2.5, but with attribution of more weight to word semantic similarity and less weight to Bert's prediction order. Furthermore, they excluded LSBert's cross-entropy loss feature from the distribution.

Four teams (CENTAL, TeamPN, GMU-WLV, and UniHD) had not considered separate measures for selecting semantically similar alternatives, relying on the order of substitutes obtained in the SG step. As discussed in the preceding section, they had either added the original sentence with the unmasked complex word to the sentence with the masked complex word and fed the sentence pair into their MLM (GMU-WLV), or merged the substitutes acquired from their MLM with substitutes sourced from a variety of linguistic databases (CENTAL and TeamPN), or used various prompts to request GPT-3 to return simplification candidates (UniHD).

3.8.3 Substitute Ranking (SR)

As laid out in section 2.3, the SR step involves ranking the final list of candidates on simplicity. To determine this order, some teams used approaches such as word frequencies obtained from large text corpora. This approach comes from the common knowledge that in general, frequently occurring words are perceived as less complex than words that appear less often in texts.

Conversely, CENTAL used a simplification database for ranking the substitutes. TeamPN used FitBert (Havens and Stal, 2019), which is a fine-tuned BERT model for tasks where a given sentence has blank or missing words or phrases, thereby trained to predict the most appropriate words or phrases to fill in these blanks. As discussed in the previous section, MANTIS had adapted LSBert's feature distribution that contains similarity and simplicity related features, by which they combined the SS and SR step. After ranking, as a post-processing step, they removed candidates that had equivalence scores that were lower than the mean equivalence scores of all candidates. They based their equivalence scores on textual entailment. Textual entailment refers to the relationship between two text fragments, where the truth of one logically follows from the other. For the purpose of the Lexical Simplification task, it served as a measure to assess how well a substitute in a simplified sentence retained the meaning of the complex word in the original sentence. To measure textual entailment, they used

Roberta-large-mnli[5], an MLM based on the RoBERTa (Liu et al., 2019) architecture[6], but fine-tuned on the Multi-Genre Natural Language Inference corpus[7]. This model provides entailment scores, representing the probability that one text fragment logically entails another, capturing the logical connection between the two fragments. Finally, RCML ranked their substitutes on simplicity by assigning CEFR English proficiency levels to these substitutes. The CEFR levels were retrieved from English Vocabulary Profile[8], a large vocabulary with CEFR-labeled words.

Interestingly, six of the ten teams (CILS, PresiUniv, PolyU-CBS, GMU-WLV, UoM&MMU, and UniHD) did not consider explicit simplicity measures when ranking their substitutes, relying solely on the probability scores obtained during the previous steps. Two of these teams (UoM&MMU and UniHD) had applied a prompt-based model during the SG step. One (UoM&MMU) of these prompt-based models had been fine-tuned with a variety of simplification datasets, among which a dataset based on CEFR levels (Uchida et al., 2018). By fine-tuning their model in this way, they had purposefully taken the SR step into account before the very first (SG) step.

3.9 Results of Participating Systems

Figure 3.2 displays the results for English of all 33 models, which includes the results of the two baseline models TSAR-TUNER (Ferrés et al., 2017) and TSAR-LSBert (Qiang et al., 2021) outlined in section 3.7.

The models were evaluated on the ten metrics discussed in section 3.6. As illustrated in the figure, the outcomes were sorted on their results on the ACC@1 metric. For the sake of brevity and clarity, given the potential complexity that might arise from addressing ten metrics across ten systems, this section is limited to examining the performances corresponding to this particular performance measure only. As pointed out in section 3.6, the ACC@1 metric concerns the proportion of instances where the highest-ranked predicted substitute appears in the list with gold labels. For example, if a system obtained a score of 0.6000 on the ACC@1 metric, this meant that for 60% of the complex words, the substitute that was predicted as most similar to the complex word was found in the list with gold labels.

Starting with the results of the baseline models, the pre-trained MLM TSAR-LSBert ranked fifth in the results, performing significantly better than the non-neural TSAR-TUNER that obtained the 24th place. TSAR-TUNER, however, managed to outperform several other pre-trained language models.

Notably, of the 31 submitted models, only four of them topped TSAR-LSBert. Of these four, three had used prompting templates to retrieve the substitutes. These models were submitted by UniHD, ranked first and second, as well as UoM&MMU, securing the fourth place. UniHD's model, featuring a collection of six distinct prompting templates offering varying levels of context, achieved the highest ACC@1 scores among both models. Their second-ranked model employed a similar architecture but with just one standard prompt, which was based on the sentence containing the complex word. UniHD's first and second ranked GPT-based systems obtained substantially higher

[5] `https://huggingface.co/roberta-large-mnli`, last accessed on 2023-08-14.

[6] I will discuss the RoBERTa architecture in section 4.1.

[7] `https://huggingface.co/datasets/multi_nli`, last accessed on 2023-08-14.

[8] `https://www.englishprofile.org/wordlists`, Cambridge University Press (2015), last accessed on 2023-08-14.

Team	Run	ACC @1	ACC@1 @Top1	ACC@2 @Top1	ACC@3 @Top1	MAP @3	MAP @5	MAP @10	Potential @3	Potential @5	Potential @10
UniHD	2	**0.8096**	**0.4289**	**0.6112**	**0.6863**	**0.5834**	**0.4491**	**0.2812**	**0.9624**	**0.9812**	**0.9946**
UniHD	1	0.7721	0.4262	0.5335	0.5710	0.5090	0.3653	0.2092	0.8900	0.9302	0.9436
MANTIS	1	0.6568	0.3190	0.4504	0.5388	0.4730	0.3599	0.2193	0.8766	0.9463	0.9785
UoM&MMU	1	0.6353	0.2895	0.4530	0.5308	0.4244	0.3173	0.1951	0.8739	0.9115	0.9490
LSBert-baseline	1	0.5978	0.3029	0.4450	0.5308	0.4079	0.2957	0.1755	0.8230	0.8766	0.9463
RCML	2	0.5442	0.2359	0.3941	0.4664	0.3823	0.2961	0.1887	0.8310	0.8927	0.9436
RCML	1	0.5415	0.2466	0.3887	0.4691	0.3716	0.2850	0.1799	0.8016	0.8847	0.9115
GMU-WLV	1	0.5174	0.2493	0.3538	0.4477	0.3522	0.2626	0.1600	0.7533	0.8337	0.8981
CL Lab PICT	1	0.5067	0.2064	0.3297	0.4021	0.3278	0.2331	0.1369	0.7265	0.7828	0.8042
UoM&MMU	3	0.4959	0.2439	0.3458	0.4235	0.3273	0.2411	0.1461	0.7560	0.8310	0.9088
teamPN	2	0.4664	0.1823	0.3056	0.3378	0.2743	0.1950	0.0975	0.6729	0.7506	0.7506
MANTIS	3	0.4611	0.2117	0.3351	0.4235	0.3227	0.2553	0.1673	0.7747	0.8793	0.9436
teamPN	3	0.4504	0.1769	0.2841	0.3297	0.2676	0.1872	0.0936	0.6648	0.7399	0.7399
teamPN	1	0.4477	0.1769	0.2815	0.3297	0.2666	0.1874	0.0937	0.6621	0.7453	0.7453
PolyU-CBS	3	0.4316	0.2064	0.2788	0.3297	0.2683	0.1995	0.1178	0.6139	0.6997	0.7747
MANTIS	2	0.4209	0.1662	0.2654	0.3565	0.2745	0.2193	0.1507	0.7131	0.8391	0.9517
PresiUniv	1	0.4021	0.1581	0.2305	0.3002	0.2603	0.1932	0.1136	0.6568	0.7399	0.7962
PolyU-CBS	1	0.3914	0.1823	0.2627	0.3002	0.2576	0.1883	0.1113	0.5924	0.6836	0.7533
CILS	3	0.3860	0.1957	0.2627	0.3083	0.2603	0.2014	0.1267	0.5656	0.6005	0.6380
CILS	2	0.3806	0.1903	0.2600	0.3083	0.2597	0.1997	0.1262	0.5630	0.6005	0.6434
PresiUniv	3	0.3780	0.1474	0.2010	0.2573	0.2277	0.1609	0.0897	0.5656	0.6058	0.6327
CILS	1	0.3753	0.2010	0.2788	0.3109	0.2555	0.1964	0.1235	0.5361	0.5898	0.6300
CENTAL	2	0.3619	0.1152	0.2091	0.2788	0.2573	0.2056	0.1271	0.6541	0.7667	0.8418
TUNER-baseline	1	0.3404	0.1420	0.1689	0.1823	0.1706	0.1087	0.0546	0.4343	0.4450	0.4450
PolyU-CBS	2	0.3190	0.1447	0.2091	0.2573	0.1973	0.1490	0.0901	0.5120	0.6032	0.7104
GMU-WLV	2	0.2815	0.0804	0.1689	0.2493	0.1899	0.1589	0.1200	0.5630	0.7399	0.8981
CENTAL	1	0.2761	0.1313	0.1930	0.2117	0.1635	0.1183	0.0707	0.3780	0.4021	0.4182
UoM&MMU	2	0.2654	0.1367	0.2171	0.2680	0.1820	0.1307	0.0794	0.4906	0.5817	0.6756
PresiUniv	2	0.2600	0.1018	0.1313	0.1554	0.1350	0.0862	0.0439	0.3136	0.3163	0.3163
twinfalls	1	0.1957	0.0509	0.0884	0.1233	0.1175	0.0879	0.0535	0.3485	0.4235	0.5067
twinfalls	2	0.1849	0.0643	0.0911	0.1367	0.1182	0.0857	0.0514	0.3565	0.4075	0.4664
NU HLT	1	0.1447	0.0670	0.1018	0.1179	0.0902	0.0583	0.0301	0.2600	0.2815	0.2895
twinfalls	3	0.0455	0.0107	0.0348	0.0455	0.0370	0.0277	0.0182	0.1474	0.2305	0.3619

Figure 3.2: Results submitted for the English track, taken from Saggion et al. (2022)

scores (0.8096 and 0.7721, respectively) on the ACC@1 metric than their runner-ups.

The highest-performing non-GPT based model was submitted by MANTIS. They ranked third with an ACC@1 score of 0.6568. They had obtained this score with an MLM, i.e., RoBERTa's[9] base variant. During the SG step, they had added the original sentence including the unmasked complex word to the sentence in which the complex word was masked, adopted from LSBert, but without the random masking of 50% of the words in the original sentence excluding the complex word. During the SS and SR step, which they had combined into one, they had reweighted LSBert's feature distribution, as discussed in section 3.8.2. As a post-processing step, they had removed candidates that had equivalence scores that were lower than the mean equivalence scores of all candidates. They had based their equivalence scores on textual entailment, as explained in section 3.8.3. UoM&MMU, whose best model ranked fourth with an ACC@1 score of 0.6353, had fine-tuned the RoBERTa large MLM with a variety of simplification corpora (among which a corpus with simplification substitutes based on CEFR levels (Uchida et al., 2018)), before they prompted the model to return simplification candidates. They had also removed antonyms with WordNet.

The above two models secured the third and fourth out of a total of 33 places by applying an MLM based on RoBERTa during the SG step, surpassing the other MLMs on their respective performances, indicating the valuable contribution of this specific MLM to this Shared Task.

[9] I will discuss the RoBERTa architecture in section 4.1.

Furthermore, the above-mentioned results illustrate the valuable contribution of combining unsupervised approaches with supervised linguistic resources. Yet, the vast difference between the discussed second-ranking (0.7721) GPT based model — with just a standard prompting template — and MANTIS' third-ranking (0.6568) RoBERTa MLM — obtained with a series of manual modifications — may indicate that GPT-based models are not only the most effective, but also the most efficient way forward in future Lexical Simplification tasks.

TSAR-LSBert's fifth place represented a score of 0.5978. The teams ranking below TSAR-LSBert exhibited a range of scores, with the highest being 0.5442 and the lowest 0.0455. With scores of 0.5442 and 0.5415, the sixth and seventh ranks were taken by RCML. Their sixth ranked model, explained in section 3.8, featured the transformer-based, yet non-MLM, XLNet model that bases its predictions on permutations of the input sequence. The SS step was performed by selecting the generated substitutes on their BERTScores, after which the SR step ranked the substitutes on CEFR level. RCML's other model, 7th ranked, was nearly similar, as the only difference was an additional morphological measure to remove unfitting substitutes, based on a comparison of their PoS and morphological features with those of the complex word.

Among the teams that had submitted research papers, PresiUniv recorded the lowest score, occupying the 29th position with an ACC@1 score of 0.2600. However, they had also submitted two other systems with higher scores of 0.4021 and 0.3780, securing the 17th (described in section 3.8) and 21st positions, respectively. These two scores were still superior to TSAR-TUNER's score of 0.3404.

It is remarkable that, among all submitted MLMs, GMU-WLV managed to achieve the eighth place with an ACC@1 score of 0.5174, despite having disregarded the separate SS and SR steps. During the SG step, as explained in section 3.8.1, their MLM was only supplemented with the original sentence including the unmasked complex word, adopted from LSBert (next to the sentence in which the complex word was masked). This illustrates that MLMs have the capability to attain reasonable ACC@1 scores with minimal adjustments. Yet, as indicated earlier, hybrid approaches that combined MLMs with supervised linguistic resources, such as the model ranked third (MANTIS), demonstrated better results.

3.9.1 The ACCuracy of ACC@1

As introduced in the preceding section, the outcomes of the participating systems were organized based on the ACC@1 metric results. Consequently, I sorted my final models in chapter 5 on this metric to facilitate the comparison of their relative performances. In hindsight, I discovered significant constraints associated with using ACC@1 as the primary metric for evaluating lexical simplification system outputs. I will reflect on these limitations in section 6.5.

Chapter 4

Methodology

In the preceding two chapters, I covered a multitude of approaches that have been applied to the task of Lexical Simplification. This thesis project aims to explore and compare the contributions of a selection of these methods, while also introducing and evaluating new concepts to accomplish lexical simplification. As introduced in section 1.2, the research question investigated in this thesis is the following:

"How do different approaches for Substitute Generation, Selection and Ranking compare in the context of building a Lexical Simplification system for the English language?"

The subsequent sections provide an overview of the methods designed and the experiments conducted to address my research question, considering steps two through four of the Lexical Simplification task for the English language: generating, selecting, and ranking substitutes for complex words. These steps are carried out according to the requirements for the TSAR-2022 Shared Task on Multilingual Lexical Simplification (Stajner et al., 2022; Saggion et al., 2022), described in the previous chapter.

All experiments described in the current chapter are conducted on the trial set, as this dataset was provided including gold labels, as denoted in section 3.4. The results on the trial set gave me a slight indication of how my models could potentially perform on the test set, which was the evaluation dataset for this Shared Task. To determine the performance for each experiment on the trial set, I accumulated the results on all ten metrics, explained in section 3.6, into one total score. Given that there are ten metrics, each with a maximum score of 1, the cumulative total score could maximally reach an upper limit of 10. The necessity for this comprehensive score arose from the fairly small size of the trial set. Focusing on a single metric, such as @ACC1, could result in overfitting, where models are disproportionately optimized for that particular metric, compromising their generalizability on other datasets. The inclusion of multiple metrics could help mitigate this risk of overfitting, facilitating a more balanced evaluation of a models' performance across various evaluation metrics.

After each Lexical Simplification process step, I assessed my models on the trial set to assess the distinct impacts of the strategies employed in that step. My method sequentially covers the steps of Substitute Generation (SG), Substitute Selection (SS), and Substitute Ranking (SR). However, the design of my models required a division of the SS step into two phases. The first phase contains a set of general selection criteria. The highest-performing models resulting from this first phase of the SS step proceeded

to the second phase, where I employed three separate strategies to further refine the semantic similarity of the substitutes with the complex word. For each of these three strategy types, the model exhibiting the highest performance systematically advanced to the SR step. During the SR step, I applied two separate simplification strategies. Consistent with the methodology for the prior steps, the highest-performing model for each of these two strategy types was methodically carried forward to the next phase — in this case, evaluation on the final test set.

This modular approach facilitated maintaining track of performance advancement of the diverse strategy types throughout the steps in the Lexical Simplification process, as well as an evaluation of all used strategy types on the test set. Nevertheless, it is important to consider that this method might mean that if one type of strategy, applied across multiple models, resulted in numerous top scores on the trial set, only the top-performing model of that strategy would advance to subsequent stages and ultimately undergo evaluation on the final test set. Other high-performing models within the same strategy would be discarded. Conversely, if a different strategy led to lesser scores than the previously mentioned one, its top model would still progress further in the process despite lower scores. Although this might have negatively impacted the final test set outcomes, I prioritized my assessment of diverse strategies — and the potential insights that this would bring — over obtaining the highest possible score on the test set.

In addition to the above-mentioned method, the best-performing model per strategy after phase two of the SS step was also directly evaluated on the test set, bypassing the 'ranking on simplicity' SR step. This choice draws on the findings in section 3.3, where a proportion of the most frequently suggested gold labels in the trial set did not seem to be simpler. Consequently, evaluation on the test set of the models that scored highest after the SS step — where substitutes are selected on their similarity with the complex word — could provide valuable insights into the extent to which annotators used similar words, regardless of whether these words would be simpler.

The following sections cover the specific strategies implemented to design a Lexical Simplification model.

4.1 Substitute Generation (SG)

Section 2.4 elaborates on the advantages of MLMs, notably their capability to acquire bidirectional and concurrent contextual representations for words. Their masking property enables the generation of multiple contextually appropriate words for each masked token, a feature that ultimately lends itself to the SG step of the Lexical Simplification process. These advantages influenced my decision to use MLMs for the generation of substitutes. I chose a total of six MLMs[1], consisting of the base and large variants of three main models: BERT (Devlin et al., 2019), discussed in section 2.4, RoBERTa (Liu et al., 2019), and Electra (Efficiently Learning an Encoder that Classifies Token Replacements Accurately, Clark et al. (2020)). While these models employ the Transformers (Vaswani et al., 2017) architecture and BERT's (Devlin et al., 2019) masking property, they differ in training data size and masking strategy. In terms of training data, RoBERTa utilizes the largest corpus for training among the three models. With regard to the masking strategy followed, BERT applies a single static mask during data preprocessing, consistently using the same mask for each training instance

[1]All selected MLMs are pre-trained and publicly available at `https://github.com/huggingface/transformers`, last accessed on 2023-08-14.

throughout the epochs. In contrast, RoBERTa employs dynamic masking, generating a new masking pattern for each sequence fed into the model. This dynamic masking approach is particularly advantageous when pre-training for longer durations or with larger datasets. Electra, on the other hand, applies a distinct approach called 'replaced token detection', as it uses a generator to replace tokens with alternatives and a discriminator to discern between original and replaced tokens. This approach renders Electra computationally more efficient compared to the other two models.

I chose BERT due to its status as the pioneering MLM model. Furthermore, I selected RoBERTa as it is BERT's successor, claiming (Liu et al., 2019) state-of-the-art results with its pre-training on the largest corpus of the three models, supported by its impressive performance on the TSAR-2022 Shared Task for the English language, as presented in section 3.9[2]. Finally, I opted for Electra to assess the effectiveness of its unique 'replaced token detection' strategy.

My models generated 30 substitutes, as my experiments with other amounts (20, 50, 100) obtained inferior results. The amount of 30 proved to maintain an adequate balance between the generation of a smaller set of good substitutes vs. a larger number that could possibly generate more suitable candidates, but could include less fitting substitutes as well which would then have to be filtered out later.

I provided the above MLMs with the sentence in which the complex word had been masked, supplemented by the original sentence including the unmasked complex word, resulting in a sentence pair. I had derived this strategy from LSBert, described in section 2.5. LSBert had also randomly masked 50% of the words in the original sentence excluding the complex word, which I did not apply in my experiments. Feeding the sentence pair to the model enabled it to consider both the context without the complex word (i.e., the sentence with the masked complex word) and the complex word in its context (i.e., the original sentence, with the complex word unmasked). Table 4.1 shows an example of results (for this purpose, temporarily cut off to the ten highest-ranked substitutes) obtained with and without the original (unmasked) sentence.

Sentence	Complex Word	Without Original (Unmasked) Sentence	With Original (Unmasked) Sentence
UK police were expressly forbidden, at a ministerial level, to provide any assistance to Thai **authorities** as the case involves the death penalty.	**authorities**	nationals, citizens, victims, refugees, authorities, prisoners, migrants, immigrants, people, officials	authorities, officials, police, authority, people, government, prosecutors, governments, agencies, investigators

Table 4.1: Example (trial set) of ten highest-ranked substitutes after SG step, predicted by Electra's large variant, for versions with and without original sentence with (unmasked) complex word. Note that duplicates and inflected forms of the complex word will be removed in the first phase of the SS step.

Without the context of the complex word, although the model produced a variety of alternatives that did suit the context, the model showed unawareness of the meaning of the complex word. However, when the complex word in its context of the sentence

[2]I will mention my knowledge about RoBERTa's performance on the Shared Task as a limitation of my research in section 6.1.3.

was additionally provided, the model was enabled to generate more words that would accurately reflect the meaning of the complex word.

To ensure uniformity among the used models, I converted the substitutes to lowercase to address the casing distinction present in RoBERTa's identical substitutes. As an additional pre-processing action, I eliminated empty elements and unwanted characters like hashtags from the resulting list. This cleaning process resulted in the possibility of having fewer than 30 candidates for the final input into the SS step.

SG — Trial Set Results

To enable calculating preliminary evaluation scores in this phase of the process, I temporarily sized down the substitute list (consisting of a maximum of 30 substitutes) to the ten highest-ranked substitutes. The scores obtained by my six models with and without the original (unmasked) sentence are shown in table 4.2. As explained in the introductory section of this chapter, these scores are accumulated scores, composed of the aggregated individual scores on all ten metrics discussed in section 3.6.

Substitute Generation (SG)	Without Original (Unmasked) Sentence	With Original (Unmasked) Sentence
bertbase	2.165	4.242
bertlarge	2.1365	4.5206
electrabase	1.9819	4.1810
electralarge	1.6004	4.9233
robertabase	2.3809	4.9977
robertalarge	4.6356	4.7027

Table 4.2: Accumulated scores (trial set) after SG step, model versions with and without original sentence with (unmasked) complex word. The scores only consider the ten highest-ranked substitutes.

The results strongly confirm my design decision to provide the MLMs with these sentence pairs instead of only with the sentence where the complex word was masked. RoBERTa's large variant was an exception, as it managed to achieve a high score without such clues, surpassing three other models that were provided with them. However, the table also shows that addition of the complex word in its context to this particular model hardly contributed to higher results.

The six models shown in table 4.2 that had additionally been provided with the original (unmasked) sentence were advanced to the first phase of the SS step.

4.2 Substitute Selection (SS) — Phase One

As pointed out in the introductory section of this chapter, my model design required a division of the SS step into two phases. This section discusses the first phase, comprised of a standard set of selection criteria. The six models that had progressed to this phase were expected to gain from this phase, as it can be perceived as a semantic pre-processing step.

From the list of maximally 30 substitutes obtained after the SG step, I removed occurrences and inflected forms of the complex word. If the lemmas of the complex word and its substitute were equal, the substitute was removed. This process took care of removing duplicates of the complex word as well as its inflected forms. In

addition, I eliminated antonyms of the complex word. Antonyms, i.e., words that have a contrasting meaning as opposed to another word, often occur in similar contexts as their synonyms. This makes them susceptible to being generated by MLMs as substitute candidates for a complex word. Due to their dissimilarity to the complex word, I excluded them from the predictions. I used the antonyms defined by WordNet to determine whether a substitute candidate was an antonym of the complex word. As introduced in the first chapter, WordNet organizes words with equivalent meanings into sets of synonyms called synsets. Various relationships interlink these synsets, including antonymy, a relation that maps words of opposite meanings to each other. I leveraged this structure by filtering out antonyms from the substitute candidates. I compared all synsets of both the lemmatized complex word and the lemmatized substitute candidates, and only kept the substitutes without antonymic relationships with the complex word. I based my approach (i.e., of comparing all synsets) on the assumption that my systems do not know to which synset a specific word belongs. This required a comparison of all synsets of a substitute with all synsets of the complex word. Table 4.3 shows an example of the highest-ranked substitutes before and after this first phase in the SS process. For this purpose, the substitute list is temporarily cut off to the ten highest-ranked substitutes.

Sentence	Complex Word	Before SS Step Phase 1	After SS Step Phase 1
UK police were expressly forbidden, at a ministerial level, to provide any assistance to Thai **authorities** as the case involves the death penalty.	**authorities**	authorities, officials, police, authority, people, government, prosecutors, governments, agencies, investigators	officials, police, people, government, prosecutors, governments, agencies, investigators, officers, courts

Table 4.3: Example (trial set) of ten highest-ranked substitutes, predicted by Electra's large variant, before and after SS step phase 1, in which duplicates and inflected forms of a complex word are removed, as well as its antonyms.

The table illustrates that the substitute *authorities* was excluded, as it was identical to the complex word. The substitute *authority*, an inflected form of the complex word, was adequately removed as well. This allowed for two other substitute, *officers* and *courts*, to be included in the top ten substitutes (recall that the SG step initially generates 30 substitutes). Note that the substitute lists generated for the — relatively small — trial set coincidentally did not include any antonyms of the complex word.

SS Phase 1 — Trial Set Results

Equal to the method employed in the SG step, to compute preliminary evaluation scores in this phase of the process, I temporarily sized down the (maximum of) 30 substitutes to the ten that were ranked highest. The accumulated scores on the trial set, obtained by my six models before and after SS step phase one, are shown in table 4.4. As expected from a semantic pre-processing step, the models demonstrated improvements, yet marginal. Electra's base model showed no difference. Consistent with the method discussed in the opening section of this chapter, the two most promising models, Electra's large variant and RoBERTa's base variant, their respective scores of 4.9413 and

5.1161 marked bold in the table, were advanced to phase two of the SS step.

Substitute Generation (SG)	Before SS Step Phase 1	After SS Step Phase 1
bertbase	4.242	4.4439
bertlarge	4.5206	4.6492
electrabase	4.1810	4.1810
electralarge	4.9233	**4.9413**
robertabase	4.9977	**5.1161**
robertalarge	4.7027	4.8159

Table 4.4: Accumulated scores (trial set) before and after SS step phase 1; models with scores in bold are advanced to phase 2 of the SS step. The scores only consider the ten highest-ranked substitutes.

4.3 Substitute Selection (SS) — Phase Two

This section discusses the second phase of the SS step, where the best two models resulting from the first phase were further refined on their capabilities to select semantic similar substitutes for the complex words. I used three separate strategies for this: shared synsets, shared hypernyms, and BERTScore. I executed all these experiments separately to evaluate the individual contributions of each of these three methods. The substitute selection experiments were conducted on the full list (up to 30) of substitute candidates resulting from phase one of the SS step. After implementing each of these three strategies, I trimmed the resulting lists of (up to 30) substitute candidates to final lists that contained the ten highest-ranked substitutes. As this marked the conclusion of the SS step's second and final phase, these ten substitutes were deemed to be most semantically similar to the complex word.

My choice of selecting ten substitutes was guided by knowledge about the average number of unique annotations (10.55) for a complex word, provided by Saggion et al. (2022), discussed in section 3.4. This average value was revealed after the execution of the Shared Task, although the participants had been provided in advance with an average of 9.64 (Stajner et al., 2022) unique annotations across the entire multilingual dataset. Consequently, any advantage I might have had from knowing the average for the English language is substantially limited, given the minimal difference with the presupplied average for the multilingual dataset. In addition, average values can be skewed by outliers, i.e., values that significantly deviate from the rest of the dataset. This inherent limitation of average values makes them less informative than other — not supplied in advance — measures of variation such as the median, which represents the middle value of the unique annotations.

The following sections discuss the relevance and structure of my three strategies to select substitute candidates based on their semantic similarity to the complex word, as well as their individual contributions to the trial set results.

4.3.1 Synset(s) Shared

My first strategy to execute SS step phase two involves the identification of shared synsets between the complex word and each of its substitute candidates. WordNet, introduced in the opening chapter, classifies words into synonym clusters known as synsets, each expressing a distinct concept. When words belong to the same synset,

they have an equivalent meaning, although there may be slight variations in how they are used or in the feelings they convey. Finding out whether the complex word and a substitute share a synset may help determine the semantic similarity between the complex word and the substitutes. This could lead to a more accurate selection of substitutes that preserve the meaning of the complex word.

I compared all synsets of both the lemmatized complex word and the lemmatized substitute candidates, and gave priority in ranking to substitutes that shared at least one synset with the complex word, with their mutual sequence determined by their placement in the original list resulting from SS step phase 1. Substitutes that did not share a synset with the complex word or that were not present in WordNet (the latter instances usually involved words generated during the SG step that were either very rare or did not constitute real words) were subsequently added, retaining their original sequence. Next, I trimmed the substitute list to the ten highest-ranked substitutes.

When revisiting the example of table 4.3, the list with substitutes for the complex word **authorities** would now start with the substitutes that share at least one synset with the complex word. Table 4.5 visualizes this example.

Sentence	Complex Word	Prediction = shared synset	Before SS Step Phase 2	After SS Step Phase 2 **Synset(s) Shared**
UK police were expressly forbidden, at a ministerial level, to provide any assistance to Thai **authorities** as the case involves the death penalty.	**authorities**	agencies	officials, police, people, government, prosecutors, governments, agencies, investigators, officers, courts	agencies, officials, police, people, government, prosecutors, governments, investigators, officers, courts

Table 4.5: Example (trial set) of top ten predictions with Electra's large variant, before and after SS step phase 2; first strategy, where shared WordNet synsets are given priority in ranking.

As *agency*, which is the lemmatized form of *agencies*, belongs to the same synset[3] as **authority**, the lemmatized form of the complex word **authorities**, the substitute *agencies* will be ranked first, after which the remaining substitutes are appended.

Upon further reflection, checking WordNet on non-lemmatized complex words and their non-lemmatized substitute candidates might have improved synset coverage, potentially leading to better outcomes. This is because WordNet includes non-lemmatized words with unique senses different from their lemmatized versions. If words in the form in which they originally appear, such as **authorities**, are found in WordNet, their individual senses would be better reflected in their synsets than in the synsets belonging to their lemmatized forms. For example, **authorities** is listed in a synset[4] that also lists the word *government*. As *government* is one of the predictions, it would be ranked as the substitute most semantically similar to the complex word. In this particular case, *government* may indeed be more semantically similar to the complex word than *agencies*. I further reflect on this topic in section 6.1.3.

[3] in the sense of *an administrative unit of government.*

[4] in the sense of *the organization that is the governing authority of a political unit.*

SS Phase 2 — Trial Set Results — Synset(s) Shared

For both models resulting from phase one of the SS step, i.e., Electra's large variant and RoBERTa's base variant, the shared synsets between the complex word and their substitutes were identified. This led to the creation of two new models. Table 4.6 shows their accumulated scores based on shared synsets.

Substitute Selection (SS)	Before SS Step Phase 2	After SS Step Phase 2 **Synset(s) Shared**
electralarge - synset(s) shared	4.9413	5.1402
robertabase - synset(s) shared	5.1161	**5.3215**

Table 4.6: Accumulated scores (trial set) before and after SS step phase 2, strategy 1 (synsets shared); model with score in bold is systematically advanced to the SR step.

The results show, yet slight, improvements compared to the outcomes after phase one of the SS step.

For the reasons given in the introductory section of this chapter, the model that would attain the highest accumulated score for a specific strategy type was systematically advanced to the Substitute Ranking step. For this particular strategy involving shared synsets, this applied to the model that was grounded on RoBERTa's base variant before SS step phase two, and shared synsets after this phase, its score of 5.3215 marked bold in the table.

4.3.2 Hypernym(s) Shared

The previously used strategy compared whether the complex word and each of it substitute candidates were synonyms of each other, by retrieving their shared WordNet synsets. While synonym relations maintain semantic similarity, this comparison might overlook candidates that do not share a WordNet synset with the complex word due to subtle variations in meaning, but that may still act as valid substitutes. The strategy employed in the current section aims to overcome this limitation.

In this second strategy to execute SS step phase two, substitute candidates are selected based on whether they share a hypernym with the complex word. Hypernyms refer to the broad term used to categorize a collection of more specific items known as hyponyms. To illustrate the concept of semantic hierarchy, consider the relationship between the term *poodle* and its direct hypernym in WordNet, *dog*. As *dog*[5] resides one level higher in the semantic hierarchy, it indicates its broader meaning compared to its hyponym *poodle*[6]. Further up the semantic chain, *dog* serves as a hyponym for its direct hypernym *domestic animal*[7]. Consequently, *domestic animal*, reflecting a more abstract semantic categorization, becomes a two-level up hypernym for *poodle*. By examining shared hypernyms, the broader category to which both the complex word and a substitute belong can be identified. This consideration may aid in determining the coherence of a substitute's meaning with the meaning of the complex word.

[5] `http://wordnetweb.princeton.edu/perl/webwn?o2=&o0=1&o8=1&o1=1&o7=&o5=&o9=&o6=&o3=&o4=&s=dog&i=0&h=00000000#c`, last accessed on 2023-08-14.

[6] `http://wordnetweb.princeton.edu/perl/webwn?s=poodle&sub=Search+WordNet&o2=&o0=1&o8=1&o1=1&o7=&o5=&o9=&o6=&o3=&o4=&h=0`, last accessed on 2023-08-14.

[7] `http://wordnetweb.princeton.edu/perl/webwn?o2=&o0=1&o8=1&o1=1&o7=&o5=&o9=&o6=&o3=&o4=&s=domestic+animal`, last accessed on 2023-08-14.

To identify hypernyms for the complex word and its substitutes, I consulted WordNet again. I extracted shared hypernyms between the lemmatized complex word and each lemmatized substitute, and gave priority in ranking to substitutes that shared at least one hypernym with the complex word. I applied this method to hypernyms at either one or at two levels up in the hierarchy (separately), and also checked whether the complex word and each substitute shared either one-level up or two-level up hypernyms, thereby expanding the search realm. The ranking process is similar to the ranking process described for the synonym relations. Substitutes that shared a hypernym with the complex word were given precedence in ranking, with their order decided by their position in the initial list resulting from SS phase 1. Substitutes that did not share a hypernym with the complex word or that were not present in WordNet were subsequently added, retaining their original sequence. Next, I trimmed the substitute list to the ten highest-ranked substitutes. Table 4.7 contains an example of predictions that share one or several one-level up hypernyms with the complex word.

Sentence	Complex Word	Prediction (Has Shared Hypernym(s))	Before SS Step Phase 2	After SS Step Phase 2 **Hypernym(s) Shared**
UK police were expressly forbidden, at a ministerial level, to provide any assistance to Thai **authorities** as the case involves the death penalty.	**authorities**	agencies, investigators, nationals	officials, police, people, government, prosecutors, governments, agencies, investigators, officers, courts	agencies, investigators, nationals, officials, police, people, government, prosecutors, governments, officers

Table 4.7: Example (trial set) of top ten predictions with Electra's large variant, before and after SS step phase 2; second strategy, where shared WordNet hypernyms are given priority in ranking.

The substitute *agencies*, previously identified as a synonym of the complex word in table 4.5, now appears in table 4.7 as a substitute sharing one or several one-level up hypernyms with the complex word. This relationship can be inferred from its synonymous relation with the complex word. As per WordNet's structure, synonyms in a particular synset share their one-level up hypernym(s) unless they reside at the top of the semantic hierarchy — meaning that they are not associated with a hypernym.

Furthermore, the substitutes *investigators* and *nationals* share one or several one-level up hypernyms with the complex word, despite not being its synonyms. This observation underscores that the use of shared hypernyms can yield a broader coverage of semantically related words compared to solely shared synonyms, although the latter might imply a greater degree of semantic similarity. The names of the hypernyms that each of these three substitutes share with the complex word are displayed in table 4.8.

SS Phase 2 — Trial Set Results — Hypernym(s) Shared

The identification of shared one-level up, two-level up, and one- or two-level up hypernyms for each of both models resulting from phase one of the SS step, i.e., Electra's large variant and RoBERTa's base variant, led to the creation of six new models. Table 4.9 shows their accumulated scores.

Complex Word	Prediction	Shared Hypernym(s)
authorities	agencies	administrative_unit, administrative_body
	investigators	expert
	nationals	somebody, someone, soul, person, individual, mortal

Table 4.8: Example of one-level up shared hypernyms between complex word and predictions in WordNet.

Substitute Selection (SS)	Before SS Step Phase 2	After SS Step Phase 2 **Hypernym(s) Shared**
electralarge		
hypernyms (one-level up) shared	4.9413	4.7641
hypernyms (two-level up) shared	4.9413	5.0929
hypernyms (one- or two-level up) shared	4.9413	4.3687
robertabase		
hypernyms (one-level up) shared	5.1161	4.535
hypernyms (two-level up) shared	5.1161	**5.8015**
hypernyms (one- or two-level up) shared	5.1161	4.8551

Table 4.9: Accumulated scores (trial set) before and after SS step phase 2, strategy 2 (hypernyms shared: one level up, two levels up, and either one or two levels up); model with score in bold is systematically advanced to the SR step.

Both the Electra and RoBERTa model, italicized in the table, exhibited score improvements after implementation of the two-level up shared hypernyms.

Consistent with the procedure described in the introductory section of this chapter, the model yielding the highest accumulated score for a particular strategy type was systematically advanced to the Substitute Ranking step. For this specific strategy on shared hypernyms, this applied to RoBERTa's base variant before SS step phase two, and two-level up shared hypernyms after this phase, its score of 5.8015 marked bold in the table. Notably, the RoBERTa model obtained a significantly higher improvement on two-level up shared hypernyms than the Electra model.

4.3.3 BERTScore

The third and last strategy to execute SS step phase two involves leveraging contextualized word embeddings. As elaborated in section 2.2, contextualized word embeddings capture individual word meanings in their contextual surroundings. This comprehensive understanding facilitates the identification of appropriate substitutes that preserve the intended meaning and grammatical structure of the original text.

BERTScore (Zhang et al., 2020) is a metric specifically designed to assess the similarity between two pieces of text by using contextualized word embeddings. In the context of the SS step, BERTScore can be applied to compare the contextualized embeddings of the complex word with those of its substitute candidates. It calculates the degree of similarity between the sentence containing the complex word and each sentence that includes a potential substitute candidate for the complex word. The BERTScore pipeline is visualized in figure 4.1.

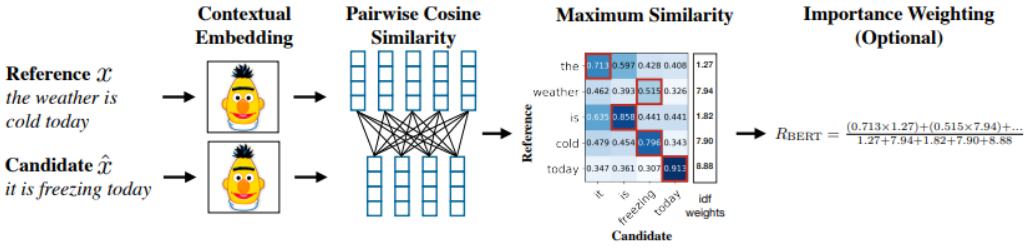


Figure 4.1: BERTScore Pipeline (Zhang et al., 2020)

BERTScore encodes the complex word and its substitute by using a pre-trained MLM, generating contextualized word embeddings that capture the semantic information in the sentence context. Subsequently, pairwise comparisons are conducted between the encoded embeddings of the complex word and those of each substitute to evaluate their semantic similarity. The cosine similarity is then calculated based on these contextualized embeddings. To maximize the matching similarity score between tokens in the sentence with the complex word and tokens in the sentence containing the substitute candidate, greedy matching is used. Greedy matching involves pairing tokens from the 'complex word sentence' with their most similar counterparts from the 'substitute candidate sentence', using the cosine similarity between the embeddings. Next to its default application, BERTScore provides an optional feature, i.e., importance weighting[8], by using inverse document frequency (idf) scores, assigning higher weights to rare words when calculating the similarity between the 'complex word sentence' and the 'substitute candidate sentence'. This allows it to account for the potential impact of less frequent words on the overall similarity calculation.

Table 4.10 shows an example of the top ten predicted substitutes after phase one in the SS step vs. the ten substitutes retrieved with BERTScore in phase two.

Sentence	Complex Word	Before SS Step Phase 2	After SS Step Phase 2 **BERTScore**
UK police were expressly forbidden, at a ministerial level, to provide any assistance to Thai **authorities** as the case involves the death penalty.	**authorities**	officials, police, people, government, prosecutors, governments, agencies, investigators, officers, courts	officials, investigators, prosecutors, police, magistrates, officers, agencies, government, courts, prisons

Table 4.10: Example (trial set) of top ten predictions with Electra's large variant, before and after SS step phase 2; third strategy, where BERTScore determined the priority in ranking.

In this example, the first position in the list stayed unaltered. The highest BERTScore for the complex word **authorities** was obtained with the word *officials*, equal to this substitute's top-ranked position retrieved after the SG step. Furthermore, the words

[8]not implemented in my models.

magistrates and *prisons* have now found their place[9] in the top ten, at the cost of the less appropriate *people* and *governments* (*government* is still adequately represented).

SS Phase 2 — Trial Set Results — BERTScore

Although the authors (Zhang et al., 2020) had presented RoBERTa's large variant as their default model due to its most promising results during their experiments, the BERTScore design allows calculating BERTScore with other models. This enabled a BERTScore calculation with each of the six models that I had used in the SG step. When applied to my best two models resulting from the phase one of the SS step, i.e., Electra's large variant and RoBERTa's base variant, this process resulted in 12 new models. Table 4.11 provides the results.

Substitute Selection (SS)	Before SS Step Phase 2	After SS Step Phase 2 **BERTScore (BS)**
electralarge		
BS with bertbase	4.9413	4.8968
BS with bertlarge	4.9413	4.8241
BS with electrabase	4.9413	4.7928
BS with electralarge	4.9413	4.7482
BS with robertabase	4.9413	5.0545
BS with robertalarge	4.9413	5.2386
robertabase		
BS with bertbase	5.1161	4.5225
BS with bertlarge	5.1161	4.3017
BS with electrabase	5.1161	4.4871
BS with electralarge	5.1161	4.4776
BS with electralarge	5.1161	4.9036
BS with robertalarge	5.1161	**5.2656**

Table 4.11: Accumulated scores (trial set) before and after SS step phase 2, strategy 3 (BERTScore (BS)); model with score in bold systematically advanced to the SR step.

Both the Electra and RoBERTa model, italicized in the table, exhibited the highest score improvements after implementation of BERTScore with RoBERTa's large variant. These two most improved models are indicated as 'BS with robertalarge'. The superior performance of RoBERTa's large variant to compute BERTScore is consistent with Zhang et al. (2020)'s results discussed in the previous paragraph.

As described in the introductory section of this chapter, the model that would achieve the highest accumulated score for a specific strategy type was systematically progressed to the SR step. For the strategy type 'BERTScore', this applied to the model featuring RoBERTa's base variant before SS step phase two, and BERTScore with RoBERTa's large variant after this phase, its score of 5.2656 marked bold in the table.

4.3.4 Final Models Resulting from Substitute Selection — Phase Two

In sections 4.3.1 through 4.3.3 above, I described three strategies in SS step phase two, designed to select substitutes on their semantic similarity to the complex word.

[9] As discussed in section 4.1, the SG step originally generates 30 substitute candidates, which are ultimately refined to a final ten most similar substitutes at the end of the second phase in the SS step.

After having implemented each strategy, I trimmed the substitutes, sorted on semantic similarity to the complex word, to a final top ten of most similar substitutes to the complex word. The highest-performing model on the trial set for each investigated strategy during phase two of the SS step – i.e., synsets shared, hypernyms shared, and BERTScore – was systematically advanced to the concluding phase of the Lexical Simplification process, i.e., Substitute Ranking. These models are aggregated in table 4.12. Remarkably, all three models stem from Substitute Generation with RoBERTa's base variant.

SG	SS	Model Name	After SS Step Phase 2
robertabase	Synsets shared	RB_Syns-shared	**5.3215**
robertabase	Hypernyms (2 levels up) shared	RB_Hyper2-shared	**5.8015**
robertabase	BERTScore with robertalarge	RB_BSrl	**5.2656**

Table 4.12: Accumulated scores (trial set) of best three models after SS step phase two.

4.4 Substitute Ranking (SR)

The main objective of the SR step is to rank alternatives for a complex word, retrieved after execution of the SS step, on their comparative simplicity. To assign simplicity rankings to the lists of ten substitutes retrieved with the final models after phase two of the SS step, I implemented two distinct strategies: hypernym-hyponym relations and CEFR levels. I executed the experiments for each of both strategies separately to evaluate their individual contributions to the trial set results. As explained in the introductory section of this chapter, the best-performing model for each of these two strategies was systematically carried forward to final test set evaluation.

The following sections discuss the structure and relevance of these two simplicity ranking strategies, as well as their individual contributions to the trial set results.

4.4.1 Hypernym-Hyponym Relations

In devising my first strategy for ranking substitutes based on simplicity, I sought inspiration from the exploration of shared hypernyms that I had performed in phase two of the SS step. In that specific model design, substitutes were prioritized if they held a horizontal relationship with the complex word by means of their shared hypernyms. However, for the SR step, I shifted focus to vertical hypernym-hyponym relationships between each substitute and the complex word.

My approach fundamentally leans on the linguistic principle of how these hierarchical relationships influence perceived complexity. A related concept is the idea of "Basic Level Categories" in cognitive psychology and psycholinguistics, introduced by Rosch et al. (1976). Basic level categories represent terms in the linguistic hierarchy that are neither too general nor too specific, and they are usually the terms most familiar to people. For instance, while most people can easily visualize and draw a bird (belonging to a basic level category), sketching specific species of birds can be more challenging. This suggests that words belonging to these categories are simpler due to their general nature. Their hyponyms, being more specific, may be perceived as more complex.

Beyond a certain height of the semantic hierarchy, hypernyms may be too general or abstract to be perceived as simpler. I will adress this topic in section 6.3.

Revisiting the 'dog' example of section 4.3.2, I explored if hypernyms such as *dog* are present in the list with substitutes and match a more complex word such as *poodle*, its one-level down hyponym. To identify hypernym-hyponym relationships, I determined whether each of the ten substitutes retrieved with the final models after phase two of the SS step served as a WordNet hypernym for the complex word. I distinctly verified this for substitutes functioning as one-level up and two-level up hypernyms. I additionally examined substitutes that were either one-level up or two-level up hypernyms, thereby broadening the scope of the search. Substitutes serving as hypernyms for the complex word were prioritized in the simplicity ranking process, their respective orders determined by their original positions in the top ten list of the selected substitutes resulting from SS step phase two. Subsequently, I added the substitutes that were not a hypernym of the complex word, also ordering them using the original position criterion.

SR — Trial Set Results — Hypernym-Hyponym Relations

Table 4.12 in section 4.3.4 presented the best three models resulting from phase two of the SS step. These were advanced to the SR step. I applied my hypernym-hyponym ranking method to the substitutes provided by these three models. As I experimented with one-level up, two-level up, and combined one- or two-level up hypernym-hyponym relations, I crafted nine new models. Table 4.13 provides the scores of these models. The three italicized headers indicate the best three models resulting from SS step phase two to which this ranking method was applied. As discussed in section 4.3.4, these models all stem from Substitute Generation with RoBERTa's base variant.

Substitute Ranking (SR)	Before SR Step	After SR Step **Hypernym-Hyponym Relations**
Synonyms shared (from SS step phase 2)		
Hypernyms1-Hyponym	5.3215	4.9818
Hypernyms2-Hyponym	5.3215	5.1837
Hypernyms1or2-Hyponym	5.3215	4.844
Hypernyms2 shared (from SS step phase 2)		
Hypernyms1-Hyponym	5.8015	5.3273
Hypernyms2-Hyponym	5.8015	5.5699
Hypernyms1or2-Hyponym	5.8015	5.0956
BScore-robertalarge (from SS step phase 2)		
Hypernyms1-Hyponym	5.2656	4.8738
Hypernyms2-Hyponym	5.2656	**5.5379**
Hypernyms1or2-Hyponym	5.2656	5.0978

Table 4.13: Accumulated scores (trial set) before and after SR step with strategy 1 (hypernym-hyponym relations); model with score in bold is systematically advanced to evaluation on the test set.

The results show, regardless of which of these three models was used, that substitutes that were ordered first on whether they functioned as a two level up hypernym of the complex word obtained the highest performance.

In line with the method discussed in the opening section of this chapter, the model that would score best on a particular strategy was advanced to test set evaluation. For this specific strategy on hypernym-hyponym relations, this applied to the model using

SS with BERTScore computed by RoBERTa's large variant, and SR with two-level up hypernym - hyponym relations, its score of 5.5379 marked bold in the table.

Note that the model employing the SR step with two-level up hypernym - hyponym relations, based on SS with shared two-level up hypernyms, featured a slightly higher score (5.5699). However, this score was lower than the score obtained with its baseline model (5.8015), which was also directly advanced to test set evaluation, as explained in the opening section of this chapter. Since the SR-using model scored lower than the model upon which it was based, it was not taken forward to evaluation on the test set.

Only one of the nine models that had implemented the SR step with hypernym-hyponym relations attained higher scores than their respective baseline models. I will reflect on this outcome in section 4.4.3.

4.4.2 CEFR Levels

I based my second SR strategy on my endeavors to align my thesis project with EDIA's readability classifier Papyrus, introduced in section 1.2. Papyrus uses CEFR language proficiency levels to identify word complexity. I investigated a variety of CEFR-labeled datasets on their applicability for simplicity ranking of the ten substitutes retrieved with the final models after phase two of the SS step. I detail my approach below.

CEFR-J

The first dataset employed was the CEFR-J Wordlist Version 1.5 (Tono, 2020), retrieved from CEFR-J Vocabulary Profile version 1.5[10]. This wordlist originates from English textbook corpora used at primary and secondary schools (years three to ten) in China, Korea, and Taiwan, in the period between 2004 and 2007. The corpora were sorted on their respective CEFR levels and commonly used words across all textbooks were extracted, each word accompanied by its Part of Speech (PoS) tag and CEFR level.

Since this dataset lacks words labeled with C1 and C2 CEFR levels, I added the CEFR-J Octanove vocabulary profile[11], created by Octanove Labs[12]. This vocabulary[13] contains words annotated with C1 and C2 levels. After merging both files into a single file, I obtained approximately 9,600 unique Word-PoS tag-CEFR level combinations. Subsequently, I converted these CEFR levels into numeric values: level A1 to 1, level A2 to 2, level B1 to 3, level B2 to 4, level C1 to 5, and level C2 to 6. Then, I replaced the complex word in the original sentence with each of its substitutes to retrieve the PoS tag of the substitute in the context of the sentence, for which I used the NLTK library[14]. Obtaining these PoS tags was needed since the CEFR-labeled dataset contained equal words that had different CEFR levels assigned based on their respective PoS tags. For instance, the word *address* was assigned the A1 level if it was used as a noun, whereas the same word used as a verb was attributed the B1 level.

If a substitute (after lemmatization) was located in the CEFR-labeled dataset, and the PoS tag of this substitute coincided with the PoS tag of the word in that dataset, the numeric value corresponding to that word's CEFR level was attributed to the

[10] `https://github.com/openlanguageprofiles/olp-en-cefrj/blob/master/cefrj-vocabulary-profile-1.5.csv`, last accessed 2023-08-14.

[11] `https://github.com/openlanguageprofiles/olp-en-cefrj/blob/master/octanove-vocabulary-profile-c1c2-1.0.csv`, last accessed 2023-08-14.

[12] `http://www.octanove.com/`, last accessed 2023-08-14.

[13] licensed under `https://creativecommons.org/licenses/by-sa/4.0/`, last accessed 2023-08-14.

[14] `https://www.nltk.org/index.html`, last accessed 2023-08-14.

substitute. With regard to the ranking method, substitutes that had been attributed the smallest numeric values, indicating the lowest CEFR levels, were placed at the top of the substitute list. Substitutes with equal numeric values were arranged according to their original positions in the top ten list of the selected substitutes resulting from SS step phase two. Next, any substitutes not found in the combined vocabulary were added, also maintaining their initial order resulting from phase two of the SS step.

CEFR-LS

The second dataset leveraged was CEFR-LS (Uchida et al., 2018)[15]. The dataset contains input sentences and words occurring in those sentences. The sentences were sourced from introductory chapters of university textbooks[16], accessible on the OpenStax[17] website, a digital initiative by Rice University (Houston, U.S.A). The words occurring in those sentences were labeled with CEFR levels stemming from CEFR-J wordlist version 1.3 (Tono, 2016) and English Vocabulary Profile[18].

My approach for this dataset, in which all six CEFR levels were represented, was largely similar to the strategy adopted for the combined CEFR-J and Octanova dataset. The key difference was that, although the sentences containing CEFR-labeled words were included in the CEFR-LS dataset, the corresponding PoS tags for these words were not. Therefore, I leveraged the NLTK library, as previously for CEFR-J, yet now to identify the PoS tags of the CEFR-labeled words in this particular dataset. After removing duplicates, the dataset featured approximately 1,300 unique Word-PoS tag-CEFR level combinations. The remainder of the process followed was equal to the approach adopted for the CEFR-J dataset, explained in its last paragraph.

CEFR-EFLLEX

I used the EFLLEX (Dürlich and François, 2018) database as the third dataset to rank substitutes on simplicity. It features approximately 15,300 Word-PoS tag combinations, along with their frequencies in CEFR-graded text corpora across five CEFR levels (excluding the C2 level). The corpora consist of 17 textbooks, 33 graded readers, and 7 online materials, all tailored for second language English learners. The materials were sourced from publishers such as Cambridge University Press, Oxford University Press, and Exam English Ltd.

Since the Word-PoS tag combinations in the EFLLEX database only had CEFR level frequencies assigned, I implemented two distinct approaches to rank the substitutes on CEFR level. The first focused on the mode (most frequently occurring) CEFR level for the substitute words appearing in that database. To each substitute occurring in that dataset, I assigned the mode CEFR level. A limitation of this method is that it does not account for a word's association with the other CEFR levels. A word might be prevalent at a particular level but also have significant occurrences at other levels, which are neglected in this method. In my second approach, I aimed to overcome this drawback by considering the distribution of CEFR levels. For each substitute, I calculated a weighted average of its frequencies across the different CEFR levels in the

[15] licensed under `https://creativecommons.org/licenses/by-sa/4.0/`, last accessed 2023-08-14.
[16] licensed under `https://creativecommons.org/licenses/by-sa/4.0/`, last accessed 2023-08-14.
[17] `https://openstax.org/`, last accessed 2023-08-14.
[18] `https://www.englishprofile.org/wordlists`, Cambridge University Press (2015), last accessed 2023-08-14.

dataset. This method assigned a more representative level to each substitute, based on a word's usage across various language proficiency levels. The remainder of the process followed was equal to the approach adopted for the CEFR-J dataset, explained in its last paragraph.

CEFR: All Datasets Combined

As highlighted in the opening chapter, one of the issues when using manually curated datasets is their scope of coverage. In an attempt to reach a broader range of CEFR-tagged words, I merged all previously mentioned CEFR datasets into one single dataset, obtaining an initial total of roughly 26,000 instances. Next, I eliminated identical duplicates and I averaged the CEFR levels of instances with matching Word-PoS pairs but different CEFR levels, using the mapped numerical values discussed in the previous sections. This process reduced the collective dataset from 26,000 to 19,000 unique Word-PoS tag-CEFR level combinations.

Table 4.14 shows an example of how a list of substitutes would be ranked on CEFR level for this CEFR dataset.

Sentence	Complex Word	Before SR Step	After SR Step **CEFR Levels**
Syria's Sunni majority is at the forefront of the uprising against Assad, whose minority Alawite sect is an **offshoot** of Shi'ite Islam	**offshoot**	extension, affiliate, arm, offspring, outpost, echo, evolution, imprint, branch, adjunct	arm: 1.97, branch: 3.37, evolution: 4.25, extension: 4.5, offspring: 5.0, echo: 6.0, affiliate, outpost, imprint, adjunct

Table 4.14: Example (trial set), predicted substitutes before (SG with roberta-base, SS BERTScores with Roberta-large) and after SR step, their ranking based on the collective CEFR dataset, and numeric values mapped to CEFR levels (1: A1, 2: A2, 3: B1, 4: B2, 5: C1, 6: C2).

The words represented with a CEFR level in the CEFR database were put first in the list. As the Word-PoS combinations for *affiliate, outpost, imprint* and *adjunct* had not been listed in that CEFR database, they were appended at the end of the substitute list. To enable calculation of evaluation scores, only the ranked words without their CEFR levels were provided to the evaluation script.

SR — Trial Set Results — CEFR Levels

Table 4.12 in section 4.3.4 presented the best three models resulting from phase two of the SS step. These were advanced to the SR step. I applied my CEFR ranking method on these three models. As I experimented with in total five different CEFR datasets, I crafted 15 new models. Table 4.15 provides the scores of these models. The three italicized headers indicate the best three models resulting from SS step phase two to which this ranking method was applied. As discussed in section 4.3.4, these models all stem from Substitute Generation with RoBERTa's base variant.

As explained in the introductory section of this chapter, the applied method involves a systematic advancement to test set evaluation of the model scoring best on a particular strategy type. For the strategy of applying CEFR levels, only the model grounded on

Substitute Ranking (SR)	Before SR Step	After SR Step **CEFR Levels**
Synonyms shared (from SS step phase 2)		
CEFR-J	5.3215	4.5136
CEFR-LS	5.3215	4.6283
CEFR-EFLLEX_mode	5.3215	4.7042
CEFR-EFLLEX_weighted	5.3215	4.3798
CEFR-All	5.3215	3.9378
Hypernyms2 shared (from SS step phase 2)		
CEFR-J	5.8015	5.1728
CEFR-LS	5.8015	5.0678
CEFR-EFLLEX_mode	5.8015	5.3633
CEFR-EFLLEX_weighted	5.8015	5.0389
CEFR-All	5.8015	4.5969
BScore-robertalarge (from SS step phase 2)		
CEFR-J	5.2656	*5.6832*
CEFR-LS	5.2656	*5.4299*
CEFR-EFLLEX-mode	5.2656	5.1089
CEFR-EFLLEX-weighted	5.2656	*5.8195*
CEFR-all	5.2656	***6.0322***

Table 4.15: Accumulated scores (trial set) before and after SR step with strategy 2 (CEFR levels); models with italicized scores performed better than their baseline; the model with score in bold is systematically advanced to evaluation on the test set.

SG with RoBERTa's base variant, SS with BERTScore with RoBERTa's large variant, and SR with the collective CEFR database in which all used CEFR datasets had been integrated, was progressed to test set evaluation. This model had obtained the highest score (6.0322) on the trial set.

Remarkably, the top four performing models out of the 15 models all originate from the model that had performed Substitute Selection with BERTScore. Moreover, only these four models, italicized in table 4.15, surpassed their respective baseline models in their scores. A similar situation had occurred for the models employing hypernym-hyponym relations, as denoted in the concluding paragraph of section 4.4.1. I will reflect on this phenomenon in the subsequent section.

4.4.3 Substitute Ranking: Essential or Excess?

During the SR step, I implemented a total of 24 models: nine for the hypernym-hyponym approach and 15 for the CEFR method. However, out of these 24 models, only five surpassed their predecessor (baseline) models from the SS step phase two in terms of performance scores. Of these five successful models, one stems from the hypernym-hyponym ranking approach, and four originate from the CEFR ranking method.

There might be several explanations for this remarable outcome, and I cover two that seem most plausible. First, the limited size of the trial set might have potentially restricted the scope of conclusions to be drawn. I will further discuss this point in relation to test set evaluation in section 5.1. Second, the strategies used to rank potential simpler substitutes during the SR step may not have been as effective on this particular Shared Task. It may underpin my observation from section 3.3: a significant propor-

tion of the gold labels did not necessarily seem simpler. This might have contributed to the stronger performance of their underlying models resulting from the SS phase. In that particular phase, the focus is primarily on semantic similarity to the complex words. This focus seemed consistent with the nature of the annotations, that clearly demonstrated a semantic relation to the complex word, as remarked in section 3.3.

To investigate my observation that a significant proportion of the gold labels did not seem simpler, I compared the CEFR levels of the most frequently suggested gold labels with the CEFR level of the complex word they were intended to simplify. Based on the collective CEFR dataset introduced in section 4.4.2, I compared the CEFR levels of the lemmatized, PoS tagged complex words, with those from the lemmatized, PoS tagged substitutes. To make sure the right Word - PoS tag combinations were retrieved, I had first PoS tagged and lemmatized both the complex word and the substitutes within the context of the original sentence. Table 4.16 shows the results.

Complex Word	**CEFR**	**Most Freq. Gold Label**	**Occurrences**	**CEFR**
compulsory	4.5	**mandatory**	11	5.0
instilled	4.0	infused	3	not listed
		introduced	3	2.36
maniacs	not listed	fanatics	5	5.0
observers	4.09	watchers	8	3.76
shrapnel	not listed	bullet	4	3.76
disguised	4.5	concealed	4	4.5
		dressed	4	2.77
offshoot	not listed	branch	6	3.37
symphonic	not listed	musical	12	2.42
deploy	5.0	send	5	2.58
authorities	3.87	**officials**	11	4.36

Table 4.16: Complex words (trial set), most frequent gold label, and numeric values mapped to CEFR levels (1: A1, 2: A2, 3: B1, 4: B2, 5: C1) from collective CEFR database. Gold labels with higher CEFR level than complex word are marked in bold.

The table shows instances where the most frequently suggested gold label holds a higher CEFR level than the complex word it had intended to simplify. Even if the gold labels with equal occurrences are held out of the calculation, there are two out of ten complex words, i.e., *compulsory* and *authorities*, whose top-ranked gold labels carry a higher CEFR level. Complex words not found in the CEFR database were not considered in this comparison. However, it is worth noting that the substitute *fanatics*, given its relatively high CEFR score, will, most likely, not have a CEFR level lower than the complex word *maniacs* which it intended to make simpler.

The outcomes triggered a further analysis of this phenomenon. As the more expansive scale of the test set might yield more statistically reliable patterns, I analyzed the test set on CEFR levels after I had evaluated my models on this test set. I report my findings in section 6.4.

4.4.4 Final Models Resulting from Substitute Ranking

With each of the above-mentioned two strategies, hypernym-hyponym relations and CEFR levels, the substitutes were ranked on their relative simplicity. In the previous sections, I identified the best model for each of both strategies, which I systematically carried forward to evaluation on the test set. Both models are aggregated in table 4.17.

SG	SS	SR	Model Name	Score
robertabase	BERTScore (robertalarge)	Hyper2-Hypo	RB_BSrl_Hyper2-Hypo	**5.5379**
robertabase	BERTScore (robertalarge)	CEFR level	RB_BSrl_CEFR-all	**6.0322**

Table 4.17: Accumulated scores (trial set) of best two models after SR step.

4.5 Summary

In this chapter, I described my approach to design a variety of models for the task of Lexical Simplification. I embraced a modular approach to evaluate my models after each individual step in the Lexical Simplification process. I adopted this method to derive the unique contributions of each strategy used in my journey towards rendering an effective simplification system. To account for the small size of the trial set that I used during my experiments, I applied an accumulated score for my evaluations, consisting of a combination of all ten metrics used in the Shared Task.

During the SG step, I crafted 12 models, half of them supplemented with the additional context of the original sentence including the unmasked complex word. This added property showed major improvements over only providing the model with the sentence that had the complex word masked. I took the six models with the added context forward into the SS step. As a first phase in this step, duplicates, inflected forms, and antonyms of the complex word were removed, resulting in six new models. Of these new models, I selected the two best-performing models to execute the second phase of the SS step with. In this step, three similarity-based approaches were implemented: shared synsets, shared hypernyms and BERTScore's contextualized embeddings. In total, 20 additional models were designed. For each of these three strategies, the best model was systematically progressed to the SR step. In this SR step, two separate simplicity ranking methods were applied: hypernym-hyponym relations and CEFR levels. This process resulted in the design of 24 new models.

From the resulting total of 62 models, five were methodically advanced for final evaluation on the test set based on predefined criteria. Two of these were progressed to the test set based on their performance after implementation of the SR phase, whereas the remaining three were selected based on their outcomes following SS step phase two — refrained from specific simplicity ranking methods. I selected the latter three models based on my observations that a significant portion of the most frequent gold labels in the trial set did not necessarily indicate simpler words. Evaluation on the expansive test set of the models that scored highest after execution of the SS step — where substitutes are selected on their similarity to the complex word — could provide valuable insights into the extent to which annotators used similar words, irrespective of their simplicity. Table 4.18 displays the five models progressed to final test set evaluation.

SG	SS	SR	Model name	Accumulated Score
RB	Syns shared	n/a	RB_Syns-shared	**5.3215**
RB	Hyper2 shared	n/a	RB_Hyper2-shared	**5.8015**
RB	BSrl	n/a	RB_BSrl	**5.2656**
RB	BSrl	Hyper2 - Hypo	RB_BSrl_Hyper2-Hypo	**5.5379**
RB	BSrl	CEFR level	RB_BSrl_CEFR-all	**6.0322**

Table 4.18: Accumulated scores (trial set); models with best scores, all to be evaluated on the test set (RB = robertabase, BSrl = BERTScore with robertalarge).

All five models had used RoBERTA's base model to conduct the SG step, supplemented with the additional context of the original sentence including the unmasked complex word. In the SS step, three of the five models used the BERTScore strategy, with RoBERTa's large variant. The fourth had employed shared synonyms, whereas the fifth had implemented shared two-level up hypernyms. The SR step was carried out for two models: one used two-level up hypernym-hyponym relations for simplicity ranking, and the other CEFR levels. The latter model (model name: RB_BSrl_CEFR-all) outperformed the other four models on the trial set.

In the next chapter, I evaluate these five models on the official evaluation dataset of the Shared Task, i.e., the test set.

Chapter 5

Results

As outlined in section 1.2, this thesis addresses the following research question:

"How do different approaches for Substitute Generation, Selection and Ranking compare in the context of building a Lexical Simplification system for the English language?"

In the preceding chapter, I presented my method to investigate this research question. I designed a total of 62 models based on various strategies for the separate SG, SS, and SR steps, and evaluated these models on the trial set by using an all-encompassing measure consisting of the accumulated scores of the ten individual metrics explained in section 3.6. The models demonstrating superior performance across various strategies in different phases of the Lexical Simplification process were systematically advanced to the subsequent stage. This culminated in the progress of five models to evaluation on the test set, which is the official evaluation dataset of the Shared Task.

The present chapter is dedicated to addressing my research question through an evaluation of the performance metrics of the above-mentioned five models on the test set. My analysis starts with a comparison of the accumulated scores of these models on both the trial and test sets. Subsequent evaluations focus exclusively on the test set outcomes. Initially, I compare the accumulated scores of my models to their scores on the ACC@1 metric — the performance measure by which the models submitted for this Shared Task had been sorted, discussed in section 3.9. Following this, I benchmark the ACC@1 scores of my models against those of the models submitted for the Shared Task in 2022. I proceed by deconstructing my highest-performing model into separate modules, in coherence with the various steps I had performed to develop it. This showed the individual contributions of these modules to the final ACC@1 score. Concluding this chapter, I discuss post-evaluation experiments conducted with my highest-performing model, aiming to contribute to a broader discussion on whether associated experiments could enhance simplification results in future research.

5.1 Trial vs. Test Set Results

This section compares the accumulated scores on the trial and test sets of the five best models that had progressed to test set evaluation. Table 5.1 presents these results.

Interestingly, the findings on the test set revealed a significant deviation from the patterns observed in the trial set results. This deviation is particularly evident

SG	SS	SR	Model name	Trial	Test
RB	BSrl	n/a	RB_BSrl	5.2656	**5.4804**
RB	BSrl	Hyper2 - Hypo	RB_BSrl_Hyper2-Hypo	5.5379	5.3851
RB	Syns shared	n/a	RB_Syns-shared	5.3215	5.2803
RB	Hyper2 shared	n/a	RB_Hyper2-shared	5.8015	4.8371
RB	BSrl	CEFR level	RB_BSrl_CEFR-all	**6.0322**	4.7078

Table 5.1: Accumulated scores (trial vs. test set), ranked on test set scores. Topscore per dataset is marked bold (RB = robertabase, BSrl = BERTScore with robertalarge).

when comparing the ranking inversion between two specific models. The model that had secured the first position with an accumulated score of 6.0322 on the trial set (i.e., the model based on simplicity ranking by CEFR levels, carrying model name RB_BSrl_CEFR-all) descended to the fifth place in the test set evaluation, due to its accumulated score of 4.7078 on this dataset. Conversely, the model that had occupied the fifth place on the trial set with an accumulated score of 5.2656 (i.e., the model based on BERTScore without additional simplicity ranking properties, with model name RB_BSrl), ascended to the first position in the test set rankings, securing an accumulated score of 5.4804. Both models were developed using RoBERTa's base variant during the SG step and BERTScore with RoBERTa's large variant during the SS step. The only distinction between the two models is whether or not the substitutes had explicitly been ranked on simplicity (in this case, by their respective CEFR levels). The inclusion of the SR step employing the CEFR strategy heavily decreased the model's performance on this Shared Task. I will reflect on factors that potentially influenced the performance of this model in section 6.4.

Since the most successful model on the test set did not use a separate simplicity ranking method, outperforming two models that had included explicit simplicity ranking methods, I devoted section 6.2 to a review of the absence of simplicity ranking methods in relation to the Shared Task. My earlier observations in sections 3.2, 3.3, and 4.4.3 also required a discussion of this phenomenon.

Lastly, it should be noted that the diverging results between the trial and test sets may be related to the small size of the trial set. The trial set might not represent the complexity and diversity of language as accurately as a larger set, thereby leading to discrepancies in model performance. I will discuss this as one of the limitations inherent to my research in section 6.1.3.

5.2 Accumulated Scores Compared to ACC@1 Scores

As shown in figure 3.2 in section 3.9, the results of the systems participating in the Shared Task had been sorted on their respective ACC@1 scores. However, I had assessed my models on their respective accumulated scores of all ten metrics described in section 3.6. To verify how my models would perform on the ACC@1 metric, I compared their accumulated scores to their ACC@1 scores. Table 5.2 displays these results.

The outcomes show that the model rankings based on the accumulated scores on all ten metrics were in close alignment with those based on the ACC@1 metric only. This may potentially be related to the inherent correlations among these ten metrics themselves. Several of the metrics evaluate performance aspects related to ACC@1, or their criteria for success may even be directly satisfied due their results on this metric.

SG	SS	SR	Model name	Test (Accum.)	Test (ACC@1)
RB	BSrl	n/a	RB_BSrl	5.4804	**0.6263**
RB	BSrl	Hyper2 - Hypo	RB_BSrl_Hyper2-Hypo	5.3851	0.6075
RB	Syns shared	n/a	RB_Syns-shared	5.2803	0.5752
RB	Hyper2 shared	n/a	RB_Hyper2-shared	4.8371	0.5268
RB	BSrl	CEFR level	RB_BSrl_CEFR-all	4.7078	0.4327

Table 5.2: Accumulated vs. ACC@1 scores (test set), ranked on ACC@1 scores. Highest-ranked model is marked bold (RB = robertabase, BSrl = BERTScore with robertalarge).

An example of the latter is the Potential@K metric. If the criterion for a positive score on ACC@1 (also called Potential@1 and MAP@1, as described in section 3.6) is met, indicating that the top-ranked prediction matches one of the gold labels, the criteria for Potential@3, Potential@5, and Potential@10 are inherently satisfied as well. This is because the Potential metric measures the proportion of instances where a minimum of one of the top-K ranked predictions appears in the list with gold labels. If the minimum of one of the 'top-1' ranked condition is met by a successful score on ACC@1, the minimum of one of the 'top-3', 'top-5', and 'top-10' ranked conditions are automatically met as well.

Furthermore, the MAP@K metric, with K=3, 5, or 10, may be indirectly influenced by the ACC@1 (= MAP@1 and Potential@1) metric, although this influence decreases by ascending values of K. Whereas ACC@1 measures whether the top-ranked substitute is present in the gold labels, MAP@3, for example, measures if the three top-ranked substitutes are present in the gold labels. A successful score on ACC@1 means that, for MAP@3 to count as a success, only the second and third ranked substitutes additionally need to be present in the gold labels. The MAP@3, MAP@5, and MAP@10 metrics are stricter than ACC@1, with their strictness increasing by an increased value of K, by which its correlation with the ACC@1 metric decreases.

The ACC@Ktop1 metric may also indirectly be affected by a positive ACC@1 score. As opposed to the Map@K and Potential@K metrics that look at whether a substitute appears in the list with gold labels, this metric measures whether a minimum of one of the top-K ranked predicted substitutes equals the most often suggested gold label. The chance that this metric is affected by a positive score on ACC@1 is inversely correlated with the number of unique elements in the list with gold labels: the more unique annotations, the lesser the chance that the highest-ranked substitute present in the gold labels also matches the most frequently suggested one.

I will discuss the applicability of these metrics for future Lexical Simplification tasks in section 6.5.

5.3 Comparison of Test Set Results with TSAR-2022 Submissions

To analyze how my five models would compare to the other models submitted for the Shared Task, I integrated their test set results in the original table of 33 models, previously shown in figure 3.2 in section 3.9. The addition of my five models resulted in a ranking of 38 models. I present their relative rankings in table 5.3, limited to the top half of the results, as all five of my models are situated within that segment.

No.	Model name	ACC	ACC			MAP			Potential		
		@1	@1Top1	@2Top1	@3Top1	@3	@5	@10	@3	@5	@10
1	UniHD	0.8096	0.4289	0.6112	0.6863	0.5834	0.4491	0.2812	0.9624	0.9812	0.9946
2	UniHD	0.7721	0.4262	0.5335	0.5710	0.5090	0.3653	0.2092	0.8900	0.9302	0.9436
3	MANTIS	0.6568	0.3190	0.4504	0.5388	0.4730	0.3599	0.2193	0.8766	0.9463	0.9785
4	UoM&MMU	0.6353	0.2895	0.4530	0.5308	0.4244	0.3173	0.1951	0.8739	0.9115	0.9490
5	**RB_BSrl**	0.6263	0.2715	0.4059	0.4784	0.4293	0.3264	0.2035	0.8467	0.9247	0.9677
6	**RB_BSrl_Hyper2-Hypo**	0.6075	0.2500	0.3736	0.4596	0.4205	0.3244	0.2023	0.8521	0.9274	0.9677
7	TSAR-LSBert	0.5978	0.3029	0.4450	0.5308	0.4079	0.2957	0.1755	0.8230	0.8766	0.9463
8	**RB_Syns-shared**	0.5752	0.2607	0.4005	0.4784	0.3953	0.3024	0.1877	0.8172	0.9086	0.9543
9	RCML	0.5442	0.2359	0.3941	0.4664	0.3823	0.2961	0.1887	0.8310	0.8927	0.9436
10	RCML	0.5415	0.2466	0.3887	0.4691	0.3716	0.2850	0.1799	0.8016	0.8847	0.9115
11	**RB_Hyper2-Shared**	0.5322	0.2311	0.3413	0.4112	0.3455	0.2640	0.1692	0.7795	0.8602	0.9327
12	GMU-WLV	0.5174	0.2493	0.3538	0.4477	0.3522	0.2626	0.1600	0.7533	0.8337	0.8981
13	CL Lab PICT	0.5067	0.2064	0.3297	0.4021	0.3278	0.2331	0.1369	0.7265	0.7828	0.8042
14	UoM&MMU	0.4959	0.2439	0.3458	0.4235	0.3273	0.2411	0.1461	0.7560	0.8310	0.9088
15	teamPN	0.4664	0.1823	0.3056	0.3378	0.2743	0.1950	0.0975	0.6729	0.7506	0.7506
16	MANTIS	0.4611	0.2117	0.3351	0.4235	0.3227	0.2553	0.1673	0.7747	0.8793	0.9436
17	teamPN	0.4504	0.1769	0.2841	0.3297	0.2676	0.1872	0.0936	0.6648	0.7399	0.7399
18	teamPN	0.4477	0.1769	0.2815	0.3297	0.2666	0.1874	0.0937	0.6621	0.7453	0.7453
19	**RB_BSrl_CEFR-all**	0.4327	0.1801	0.3064	0.3978	0.3101	0.2553	0.175	0.7768	0.9059	0.9677
20	PolyU-CBS	0.4316	0.2064	0.2788	0.3297	0.2683	0.1995	0.1178	0.6139	0.6997	0.7747

Table 5.3: TSAR-2022 scores (test set), top 20. Models marked bold were developed in the context of this thesis.

This section exclusively focuses on the ACC@1 metric performance, consistent with the examination of system outcomes presented for the Shared Task in section 3.9. Moreover, the inherent discussion is confined to the evaluation of the top two of my models in comparison with other well-performing models.

My two best-performing models, with rankings of 0.6263 and 0.6075, respectively, outclassed the best baseline model TSAR-LSBert, which holds an ACC@1 score of 0.5978. TSAR-LSBert had originally ranked fifth, but the introduction of these two models demoted it to the seventh position.

My best-performing (0.6263) model scored slightly lower than UoM&MMU's best submission (0.6353). As described in section 3.8, their approach had bypassed the SR step through fine-tuning RoBERTa's large MLM with a diverse range of simplification corpora. Their slightly better score might imply a potential post-evaluation investigation regarding updating my models in the SG step to RoBERTa's large variant. I will revisit this option in section 5.5.

Furthermore, MANTIS' best submission (0.6568), described in section 3.8, topped my best model's score. Similar to my model, they had used RoBERTa's base variant during the SG step, to which they had added the original sentence including the unmasked complex word to the sentence in which the complex word was masked. During the SS and SR step, which they had combined into one, they had reweighted LSBert's key attributes, among which lexical resources and simplification corpora. As a post-processing step, they had removed candidates that had equivalence scores that were lower than the mean equivalence scores of all candidates. Yet, as my best model had merely used BERTScore for the SS step and no ranking measures for the SR step, it had secured a score that was only 3% lower than MANTIS.

Concluding my discussion about the best-performing MLMs on this Shared Task, a combination of the approaches of MANTIS, UoM&MMU, and my best model could potentially enhance future efforts in the context of Lexical Simplification. For example, this could be achieved through the following subsequent steps:

1. Through the utilization of high-quality simplification resources, finetuning (UoM&MMU) RoBERTa's base variant (MANTIS and my best model);
2. Supplementing the model with the original sentence including the unmasked complex word (MANTIS and my best model) to perform the SG step;
3. Calculating BERTScore computed with RoBERTa's large variant (my best model) to execute the SS step. BERTScore could either be used as a sole method to execute the SS step, or MANTIS' reweighted distribution of LSBert's key attributes could be implemented, with BERTScore's contextualized embeddings replacing FastText's uncontextualized embeddings as a key attribute. As explained in section 2.5, uncontextualized embeddings in the SS step might be counterproductive for polysemous substitutes, even if the SG step had generated substitutes in the context of the sentence.

Nevertheless, as highlighted in section 3.9, the substantial difference between the second-ranking GPT-based model, scoring 0.7721 with a basic prompting template, and MANTIS' third-ranking RoBERTa MLM, scoring 0.6568 after several modifications, indicates that GPT-based models migt represent not only the most effective, but also the most efficient direction for future endeavors in Lexical Simplification tasks.

5.4 Best Model Deconstructed

To understand the contributions from the various methods I had implemented in my best-performing model, I evaluated the merits of each method on this model's performance. The model had executed the SG step with RoBERTa's base variant including the context of the original unmasked sentence, and the second phase of the SS step with BERTScores computed with RoBERTa's large variant (model name: RB_BSrl). I analyzed the performance of this model on the ACC@1 metric across the applied steps in the Lexical Simplification process. Its results are shown in table 5.4.

Description	Test (ACC@1)
SG with roberta-base, excl. original (unmasked) sentence	0.3602
SG with roberta-base, incl. original (unmasked) sentence	0.4865
SS phase 1: removal of duplicates, inflections, antonyms	0.5376
SS phase 2: BERTScore with roberta-large	0.6263

Table 5.4: ACC@1 scores (test set) based on best model (RB_BSrl) deconstructed per method used.

The results strongly support my decision to supplement RoBERTa's base variant with the unmasked original sentence. The addition of this sentence to the sentence where the complex word was masked, seems to have effectively guided the system to not only generate contextually appropriate substitutes, but also substitutes that had taken the meaning of the complex word into account. As introduced in section 3.6, the ACC@1 metric represents the proportion of instances where the top-ranked predicted substitute is present in the gold labels. Therefore, the change of score from 0.3602 to 0.4865 in table 5.4 suggests that the addition of the original sentence with the complex word in it resulted in an increased accuracy. For nearly 49% of the complex words, the substitute that had been identified by the model as most similar was included in the list of gold labels, as opposed to the initial 36%.

Moreover, the removal of duplicates, inflected forms, and antonyms of the complex word also enhanced the model's performance, although slightly. Antonyms, for example, given their tendency to appear in similar contexts, are often generated as substitutes by MLMs. The relatively small trial set did not contain antonyms in the substitute lists. Among the substitutes produced for the test set, I found a total of 19 antonyms. This figure is rather small when compared to the total number of substitutes generated for the test set — approximately 4,000, corresponding to the average of 10.55 substitutes for each of the near 400 complex words discussed in section 3.4. This minimal quantity of antonyms, pertaining to less than 0.5% of the substitutes, affirms the system's proficiency in excluding antonyms from the substitutes to be generated. The strategy of providing the system with both the unmasked and the masked sentence during the SG step may have contributed to this effectiveness. However, a causal relation should not be inferred, as the nature of the complex word itself plays a considerable role as well. For example, if a complex word is rather abstract or philosophical, there may just not be an antonym for it. Consider the word 'infinity'[1], for example: although it appears in WordNet, it does not have an antonym associated with it. Furthermore, there are numerous non-abstract words, like 'poodle' from my previous examples, for which finding an antonym logically is not possible.

Finally, sorting the substitutes based on the BERTScore metric provided the model with a concluding boost towards its final score of 0.6263. This may be attributed to BERTScore's unique design to assess the similarity — with contextualized embedding scores — between, on the one hand, the sentence with the complex word in it, and on the other, the sentence in which the complex word was replaced by the substitute.

5.5 Post-Evaluation Experiments

On the best-performing model consisting of SG with RoBERTa's base variant and SS with BERTScore computed with RoBERTa's large variant, I performed post-evaluation experiments, aiming to better understand model performance and to contribute to a broader discussion on whether these experiments could foster enhanced simplification results in future research.

I started this process by replacing RoBERTa's base variant during the SG step by RoBERTa's large variant, motivated by UoM&MMU's slightly higher score on the ACC@1 metric with RoBERTa's large variant, discussed in section 5.3. However, my results on this experiment showed an inferior score on ACC@1, namely, 0.6021. The marginally superior performance of UoM&MMU is likely attributable to their fine-tuning process, using a varied assortment of simplification corpora, rather than stemming from the SG step with RoBERTa's large variant. Consequently, I did not use RoBERTa's large variant in the SG step in my further post-evaluation experiments.

With regard to the SS step, I purposefully did not post-evaluate my best-performing model with different models to compute BERTScores. The evidence already pointed to the significant effectiveness of RoBERTa's large variant, as substantiated by both my own research findings on the trial set (section 4.3.3) and the BERTScore evaluations conducted by its authors (Zhang et al., 2020), highlighted in the same section. Therefore, any additional exploration with different models for BERTScore computation seemed redundant and unlikely to yield substantial performance improvements.

[1] `http://wordnetweb.princeton.edu/perl/webwn?o2=&o0=1&o8=1&o1=1&o7=&o5=&o9=&o6=&o3=&o4=&s=infinity&i=0&h=0#c`, last accessed on 2023-08-14.

Inspired by WordNet's fine-grained organization, possibly providing a rich resource of potential substitutes at higher hypernym levels, I decided to turn to WordNet once more for further experimentation within the SR step. I expanded my previous investigation of hypernym-hyponym relations to include hypernyms up to the fifth level. I also investigated combinations of these various hypernym levels, aiming to generate a broader collection of hypernyms to align with the substitutes.

This post-evaluation experiment led to the emergence of nine hypernym-hyponym models. It is constituted of six newly designed models and three models previously examined during the trial set experiments, detailed in table 4.13 in section 4.4.1. Out of these nine, only the two-level up hypernyms model (RB_BSrl_Hyper2 - Hypo), had previously been assessed on the test set before the post-evaluation phase. Consequently, table 5.5 presents eight models assessed on the test set for the first time. Furthermore, it lists two models already evaluated on the test set prior to post-evaluation. One is the above-mentioned two-level up hypernyms model (RB_BSrl_Hyper2 - Hypo), and the other is the best model before post-evaluation (RB_BSrl) that had not incorporated a simplicity ranking method. For clarifty purposes, these two models are italicized.

Model name	Test (ACC@1)
RB_BSrl_Hyper1 - Hypo	**0.6720**
RB_BSrl_Hyper1or2 - Hypo	**0.6666**
RB_BSrl_Hyper1or2or3 - Hypo	**0.6586**
RB_BSrl_Hyper1or2or3or4 -Hypo	**0.6505**
RB_BSrl_Hyper1or2or3or4or5 -Hypo	**0.6451**
RB_BSrl (best model evaluated before post-evaluation)	0.6263
RB_BSrl_Hyper3 - Hypo	0.6102
RB_BSrl_Hyper2 - Hypo (evaluated before post-evaluation)	0.6075
RB_BSrl_Hyper5 - Hypo	0.6048
RB_BSrl_Hyper4 - Hypo	0.5967

Table 5.5: Post-evaluation ACC@1 scores (test set) based on hypernym-hyponym relations. Including best two models before post-evaluation (italicized). Scores higher than the best model before post-evaluation are marked bold.

Interestingly, the use of this method resulted in remarkably improved outcomes, underlining the potential significance of applying WordNet's semantic hierarchy to the Lexical Simplification process. Despite my initial hypothesis in this section that favored higher hypernym levels, the highest-performing model post-evaluation simply employed one-level up hypernyms. This model had not displayed contributions to the improvement of its base model scores when applied to the trial set, as table 4.13 in section 4.4.1 demonstrated. According to my methodology, it had not been progressed to test set evaluation. This underscores the possibility that the limited size of the trial set might have inhibited the formulation of reliable conclusions, which I will address as one of the limitations of my research in section 6.1.3.

The models that had applied a combination of hypernym levels outperformed those that did not, except from the one-level up hypernym model. Yet, as the number of combinations increased, the performance of the models decreased. This implies that the superior performance of these combination-based models may be attributed to a large extent to the one-level up hypernyms used in the highest-performing model.

The scores of my hypernym-based models influenced the rankings of the other models submitted in 2022 for the Shared Task to a large extent. These changes are illus-

trated in table 5.6, showing the top 20 of the systems. Note that eight models, before post-evaluation listed in the top 20 (table 5.3 in section 5.3), have now fallen out of the top 20, including my model using CEFR levels for the SR step, RB_BSrl_CEFR-all.

No.	Model name	ACC	ACC			MAP			Potential		
		@1	@1Top1	@2Top1	@3Top1	@3	@5	@10	@3	@5	@10
1	UniHD	0.8096	0.4289	0.6112	0.6863	0.5834	0.4491	0.2812	0.9624	0.9812	0.9946
2	UniHD	0.7721	0.4262	0.5335	0.5710	0.5090	0.3653	0.2092	0.8900	0.9302	0.9436
3	**RB_BSrl_Hyper1 - Hypo (P)**	0.6720	0.2741	0.4166	0.4892	0.4551	0.3453	0.2112	0.8629	0.9274	0.9677
4	**RB_BSrl_Hyper1or2 - Hypo (P)**	0.6666	0.2661	0.4139	0.4865	0.4514	0.3463	0.2110	0.8575	0.9354	0.9677
5	**RB_BSrl_Hyper1or2or3 - Hypo (P)**	0.6586	0.2661	0.4059	0.4892	0.4495	0.3439	0.2103	0.8575	0.9354	0.9677
6	MANTIS	0.6568	0.3190	0.4504	0.5388	0.4730	0.3599	0.2193	0.8766	0.9463	0.9785
7	**RB_BSrl_Hyper1or2or3or4 -Hypo (P)**	0.6505	0.2607	0.3951	0.4865	0.4432	0.3411	0.2091	0.8575	0.9327	0.9677
8	**RB_BSrl_Hyper1or2or3or4or5 -Hypo (P)**	0.6451	0.2607	0.3951	0.4892	0.4413	0.3403	0.2086	0.8575	0.9301	0.9677
9	UoM&MMU	0.6353	0.2895	0.4530	0.5308	0.4244	0.3173	0.1951	0.8739	0.9115	0.9490
10	**RB_BSrl**	0.6263	0.2715	0.4059	0.4784	0.4293	0.3264	0.2035	0.8467	0.9247	0.9677
11	**RB_BSrl_Hyper3 - Hypo (P)**	0.6102	0.2715	0.3844	0.4704	0.4215	0.3239	0.2018	0.8413	0.9247	0.9677
12	**RB_BSrl_Hyper2-Hypo**	0.6075	0.2500	0.3736	0.4596	0.4205	0.3244	0.2023	0.8521	0.9274	0.9677
13	**RB_BSrl_Hyper5 - Hypo (P)**	0.6048	0.2688	0.4086	0.4704	0.4233	0.3238	0.2023	0.8413	0.9220	0.9677
14	TSAR-LSBert	0.5978	0.3029	0.4450	0.5308	0.4079	0.2957	0.1755	0.8230	0.8766	0.9463
15	**B_BSrl_Hyper4 - Hypo (P)**	0.5967	0.2580	0.4059	0.4731	0.4184	0.3210	0.2010	0.8387	0.9220	0.9677
16	**RB_Syns-shared**	0.5752	0.2607	0.4005	0.4784	0.3953	0.3024	0.1877	0.8172	0.9086	0.9543
17	RCML	0.5442	0.2359	0.3941	0.4664	0.3823	0.2961	0.1887	0.8310	0.8927	0.9436
18	RCML	0.5415	0.2466	0.3887	0.4691	0.3716	0.2850	0.1799	0.8016	0.8847	0.9115
19	**RB_Hyper2-Shared**	0.5322	0.2311	0.3413	0.4112	0.3455	0.2640	0.1692	0.7795	0.8602	0.9327
20	GMU-WLV	0.5174	0.2493	0.3538	0.4477	0.3522	0.2626	0.1600	0.7533	0.8337	0.8981

Table 5.6: TSAR-2022 scores (test set), top 20. Models marked bold were developed in the context of this thesis. Models obtained after post-evaluation with various hypernym levels (and combinations of them) are additionally marked with the letter P.

My best model that had implemented one-level up hypernyms was catapulted to the third position of the English track in the TSAR-2022 Shared Task, being surpassed only by UniHD's two GPT based models, which may be considered an excellent performance. Two of my other hypernym-based models directly followed, securing the fourth and fifth places. After MANTIS, which was consequently demoted from the third to the sixth place, two additional hypernym-based models filled the seventh and eighth ranks. UOM&MMU dropped to the ninth position, succeeded by my best model before post-evaluation — without the use of hypernyms — that took the tenth place. The remaining hypernym-based models occupied the 11th, 12th, 13th, and 15th places. My two models that had applied shared synonyms and shared two-level up hypernyms before post-evaluation now ranked 16th and 19th, respectively. Intriguingly, TSAR-LSBert, originally ranked 5th, was degraded to a mere 14th position.

In conclusion, the improved outcomes resulting from leveraging WordNet's semantic hierarchy accentuate the effectiveness of combining unsupervised methodologies such as MLMs and BERTScore with high-quality supervised linguistic resources such as WordNet. I will present a more detailed discourse about the potential of WordNet's hierarchic structure for future Lexical Simplification endeavors in section 6.3.

Despite these encouraging outcomes, it is important to note that this re-ranking is based on a post-evaluation experiment. Although this approach is helpful in understanding model performance, it may not generalize well to unseen data, as it does not account for potential overfitting to the test set. Moreover, my methods — this applies to all my models, including those before post-evaluation — have not been evaluated across varied datasets. Consequently, strong conclusions should not be drawn.

5.6 Summary

In this chapter, I provided the test set results of five Lexical Simplification models. The evaluation of these models was based on thorough experimentation with a total of 62 models on the trial set, discussed in chapter 4. The test set results displayed a divergence from the patterns observed in the trial set results. The model employing CEFR levels to perform ranking on simplicity obtained inferior results, whereas a model that had not implemented any simplicity ranking properties outperformed the other four models. In sections 6.2 and 6.4, I will further analyze this phenomenon in relation to this particular Shared Task.

My findings indicate that RoBERTa's base variant, supplemented with the original sentence with the complex word unmasked, together with BERTscore calculated with RoBERTa's large variant, achieved competitive scores. This model outranked TSAR-LSBert, the baseline model which had previously secured the fifth out of 33 places in this Shared Task.

Additionally, superior scores were realized during a post-evaluation experiment conducted with this best-performing model, resulting in eight models newly evaluated on the test set. These models featured a Substitute Ranking method in which substitutes were given priority in ranking if they served as a hypernym, up to the fifth level, of the complex word. The three highest-performing models were only exceeded by two GPT-based models. The potential contributions of hypernym-hyponym relations to future Lexical Simplification tasks will be discussed in section 6.3.

Chapter 6

Discussion

In the previous chapters, I presented the NLP task of Lexical Simplification, related research conducted for this task, as well as the nature and results of my experiments with various methods that can be applied during this process. I focused on steps two through four of the Lexical Simplification task for the English language, applying these to the TSAR-2022 Shared Task on Multilingual Lexical Simplification.

In this chapter, I reflect on my main findings and connect the various aspects of my research by discussing their limitations and examining their potential contributions to the broader domain of Lexical Simplification. Building upon these insights, I propose potential directions for future research.

The concluding section of this chapter offers perspectives into how the discoveries from my study could potentially contribute to the improvement of EDIA's multilingual readability analyzer Papyrus, introduced in section 1.2. Moreover, I suggest strategies on how the top-performing model from my investigations can be generalized from English to Dutch. With this proposition, I seek to assist EDIA in their endeavors to aid the Dutch government in supporting citizens with language deficiencies.

6.1 Limitations

In the pursuit of academic integrity and a comprehensive evaluation of my research, this section is dedicated to addressing the constraints associated with this study. The following sections outline the limitations that I am aware of, which may have influenced the results of my studies and subsequent conclusions. These limitations encompass several aspects: my existing knowledge about the TSAR-2022 Shared Task, the current MLM framework's deficiency to generate multi-word simplifications, as well as constraints inherent to the conducted experiments along with their subsequent evaluations.

6.1.1 Prior Knowledge

As I had studied the evaluation of the Shared Task provided by Saggion et al. (2022) before I conducted my research, I could have derived clues that were unavailable to the other participating systems at the time of the Shared Task.

One of the reasons to choose RoBERTa as one of my models for the SG step was supported by its impressive performances on the TSAR-2022 Shared Task for the English language. Furthermore, I may have been influenced to design my models by my knowledge about GMU-WLV's eighth ranking on the Shared Task. This team had

achieved this rank primarily by leveraging LSBert's contextual property, without the use of further SS or SR steps. This property entailed feeding an MLM with both the sentence containing the masked complex word and the original sentence in which the complex word had not been masked. Moreover, regarding BERTScore's contextualized embeddings, it was RCML, initially ranked sixth and seventh on the Shared Task, who had used this method, triggering my experiments with BERTScore during the SS step.

Nevertheless, although the initial design of my systems was inspired by information about the results of the Shared Task, they are unique in their combined use of a RoBERTa model, the additional context of the original sentence including the complex word, and the BERTScore metric. Furthermore, to the best of my knowledge, implementing shared hypernyms during the SS step, as well as hypernym-hyponym relations in the SR step, has not been applied earlier.

Furthermore, I conducted additional experiments as a post-evaluation. As explained in section 5.5, this process might result in a possible overfit on the test set, since these experiments were based on (knowledge about) the top-performing model on that dataset, risking that generalization to unseen data can be problematic. I was well-aware of this drawback, as I was striving for a better understanding of model performance. My subsequent objective was to contribute to a broader discourse on the potential of these experiments to foster enhanced simplification results in future research.

6.1.2 Multi-Word Simplifications

Most MLMs, such as the RoBERTa model that I had used, are by their design not able to generate simplifications of more than one word. However, annotators were allowed to submit multi-word simplifications for this Shared Task. This is illustrated by the example in the Annotation Guidelines of the complex word *Iranian* and the allowed substitute *from Iran*, which section 3.2 highlighted. To understand the extent to which multi-word annotations impacted this Shared Task, I calculated the number of annotations in the test set that consist of more than one word. Of the approximately 10,000 annotations, roughly 300 of them concern more than one word, pertaining to 3% of all annotations. Although this number may only marginally impact the results on this particular Shared Task, if systems are capable of generating multi-word expressions as alternatives for a complex word, it can enhance their real-world applicability, especially when considering languages that use many words to express one concept. Newer models, like those based on the GPT architecture, are equipped to generate multi-word expressions, as UniHD's (Aumiller and Gertz, 2022) research shows. However, the authors concluded that substitutes consisting of more than one word were sometimes unnecessarily chosen over one-word alternatives.

6.1.3 Experimentation and Evaluation

In my endeavors to extensively explore a variety of methods to perform the generation, selection, and ranking of substitutes, I have overlooked or underestimated certain aspects that could have had potential impacts on the outcomes of my experimentation. These aspects range from unexplored potentials of utilized resources, to constraints in the applied methodology, and the choice of assessment metrics.

After conducting my experiments, I discovered that WordNet includes both lemmatized words and their unlemmatized versions, in case they convey a different meaning. My discussion regarding the example in table 4.5 in section 4.3.1 supports the notion

that prioritizing words in the form in which they originally appear, if available in WordNet, can lead to more accurate semantic substitutes. My oversight in not fully utilizing WordNet might have influenced the final results. I suggest investigating whether, if a word in its original unlemmatized form is found in WordNet, its lemmatized form should additionally be included to maximize synset coverage. Each synset represents a unique sense, and not every sense might align well with the intended simplification.

Furthermore, I used the small trial set as a means of determining which models to advance to final test set evaluation. Prior to performing my research, I had realized that the size of the trial set could pose a possible obstacle, which had prompted me to incorporate multiple metrics into one accumulated score, enabling a more holistic model evaluation. Despite these adjustments, the differences in results between the trial and test sets, as presented in section 5.1, suggest that I may have underestimated the trial set's representational capacity for test set outcomes. In addition to using the trial set, alternatives exist for deciding which models to transition to the test set. An alternate strategy could entail the random selection of a number of complex words and their substitutes produced by my models — based on the test set file but without the annotated gold labels — followed by human evaluation of the quality of these substitutes. This approach, although inherently subjective, could at least unveil proposed substitutes that would definitely not be applicable. These insights could lead to a more comprehensive understanding in how to improve a model, before evaluating it on the gold labels of the test set. In this context, however, it should be noted that the systematic modular approach of my methodology resulted in a model architecture that obtained a fifth place in the TSAR-2022 Shared Task on Multilingual Lexical Simplification, illustrating the relative effectiveness and competitive standing of this approach in relation to the submitted models of the other participants.

Lastly, my choice of evaluation metrics was based on the ACC@1 metric, which measures whether the highest-ranked predicted substitute is present in the list with gold labels. Although this was also the prime sorting method used by the organizers of the Shared Task to which I conformed my evaluations on the test set, it is important to realize that, in hindsight, this metric does not seem to effectively measure the effectiveness of a Lexical Simplification model. I will discuss the significant constraints associated with using ACC@1 as a primary evaluation metric in section 6.5.

6.2 Similarity vs. Simplicity

Throughout the course of this study, I found myself thinking about the precise implications of 'simpler' within the context of the TSAR-2022 Shared Task on Lexical Simplification. This was due to a combination of factors. First, the lack of an explicit definition of 'easier to understand' and 'simpler' in the Shared Task's main papers (Stajner et al., 2022; Saggion et al., 2022) and no mentions of the target audience, which I discussed in the opening section of chapter 3. Second, comparable ambiguities in the Shared Task's Annotation Guidelines, elaborated in section 3.2. Third, my initial observations of the annotated trial set, discussed in section 3.3, later supported by my experiments on the trial set, covered in section 4.4.3. The latter section showed that several substitutes were not simpler than the complex word they had intended to simplify, based on their respective CEFR levels.

As these elements raised questions about the relationship between similarity and simplicity in this particular Shared Task, I purposefully evaluated three models on

the test set in which I had not incorporated distinct properties to rank substitutes on their respective simplicity. As remarked in sections 5.1 and 5.2, the relevance of my resasoning was affirmed by the fact that my best-performing model on the test set had not used a separate simplicity ranking method. It even outperformed TSAR-LSBert that was specifically designed for lexical simplification. Apparently, even in the absence of explicit steps to prioritize 'simpler' substitutes, a model can still perform remarkably well compared to other models on this Shared Task if it effectively captures semantic similarity without specific measures to rank on simplicity. These results imply a need for further research about what 'simple' means in the context of Lexical Simplification.

However, when considering the role of MLMs and BERTScore in the Lexical Simplification process, it should be noted that these models implicitly encapsulate simplicity information. MLMs, for example, determine the order of substitutes based on their contextualized likelihood of appearing in the masked position within the sentence. The words with the highest likelihood of fitting the sentence context, as per the model's learned language patterns, are placed at the top. As more common words appear more frequently, and frequency determines the perception of word complexity to a large extent, discussed in section 2.1, the resulting lists often favor simpler words.

In addition to the above observations from a technical perspective, there are various subjective aspects associated with simplicity. It is important to remember that simplification requirements may vary based on the target audience, as highlighted in the introductory section of the first chapter. Designing lexical simplification systems that can effectively cater to the diverse requirements of various target audiences requires a balance between semantic similarity and the subjective aspects of simplicity required by these audiences. The significant role of annotated data should not be overlooked in that process. Since annotations serve as the foundation for system evaluations, annotations of poor quality may lead to a misinterpretation of a system's performance. Therefore, it is essential to create clear and unambiguous annotation guidelines prior to the annotation task. As remarked in section 3.2, these should define terms such as 'simpler' and 'easier to read,' as well as the specific target audience that should benefit from the Lexical Simplification task. Participants constructing Lexical Simplification systems will also benefit from having this information in advance, as it guides them to design systems that effectively meet the task's specific simplification requirements.

In the subsequent section, I will revisit the significance of understanding the target audience and additionally propose potential subdivision measures.

6.3 Hypernym - Hyponym Relations

The promising results of my post-evaluation experiments, described in section 5.5, revealed an interesting contribution to the Lexical Simplification domain. To the best of my knowledge, the ranking of simplification substitutes based on whether they serve as a WordNet hypernym of the (hyponymic) complex word has not been researched before. As the use of these vertical semantic relationships resulted in the highest performing MLM across all MLMs submitted for the Shared Task, I propose additional studies into WordNet's capabilities to enhance future Lexical Simplification models.

Hypernyms refer to broad categories, thereby covering a wide semantic space. Due to their broader and more universal reach, they may be perceived as simpler, while still preserving the essence of the original complex word. However, the concept of attaining simplicity by using hypernyms is not uniformly applicable across all levels of

the hierarchy. Hypernyms at the highest levels tend to be abstract or represent general concepts. Despite their broad reach, they may be perceived as too vague to clearly convey the intended meaning. It is crucial to carefully balance the use of hypernyms in the Lexical Simplification process, between generalization on the one hand, and keeping enough specificity to accurately convey the meaning of a word on the other.

As highlighted in the opening chapter, human perception of simplicity can be influenced by several factors. Recall that these may include a person's language proficiency, native language, and any cognitive or reading impairments that they have. Additional factors, such as educational level, age, cultural influences and domain-specific knowledge should also be considered. Depending on these factors, a term might be simplified differently. For example, a person with specific background knowledge may find a specialized term (a hyponym) simple, whereas another person lacking that knowledge may find a more general term (a hypernym) simpler. To illustrate this, a technical term in the medical domain like *arrhythmia*, i.e., "an abnormal rate of muscle contractions in the heart"[1], could be simplified into the more general term *heart disease* — a direct hypernym — for a general audience, regardless of their English language proficiency level. However, an audience with medical knowledge might perceive the technical term *arrhythmia* as simple enough, yet even crucial for its specific meaning, since this term would be more effective in accurately conveying information about the specific type of heart disease. Consequently, choosing the appropriate level of generality when simplifying text heavily relies on the audience's knowledge and background. Deciding whether to replace a term with a more general hypernym depends on how familiar and understandable that hypernym is expected to be for the specific audience. This requires understanding the intended audience before starting a Lexical Simplification task.

Fine-tuning MLMs on WordNet's semantic structure including hypernym-hyponym relations could enhance a model's deeper understanding of language semantics, potentially leading to improved substitute generation, also in case of new or rare words. A fine-tuning task could also be performed on the simplification needs of specific audiences as discussed above. However, the process of fine-tuning for specific audiences can be quite complex due to the varied levels of knowledge and understanding among individuals. It would require dedicated datasets that reflect the specific simplification needs of these audiences to appropriately classify them. This goal may be extremely difficult to accomplish, if not unachievable, since individuals typically don't conform to just one category. More research in this area is recommended to address these challenges.

Extending the discussion of target audience segmentation to other categorization types, I propose an exploration of how categorization based on Basic Level Categories (Rosch et al., 1976), introduced in section 4.4.1, can contribute to Lexical Simplification. Mills et al. (2018), for instance, developed a system aimed at recognizing Basic Level Categories within WordNet. This work demonstrates that identifying these categories in WordNet is feasible. By integrating this categorization into Lexical Simplification frameworks, models could generate substitutes that not only align with word semantics but also reflect the cognitive categorization processes that people employ in their use of language. This might make the generated substitutes more intuitive and easier to understand. Such research could potentially be aligned with a categorization on target audience, examining, for example, how Basic Level Categories in WordNet might influence the perception of simplicity among varied audience subgroups.

[1] `http://wordnetweb.princeton.edu/perl/webwn?o2=&o0=1&o8=1&o1=1&o7=&o5=&o9=&o6=&o3=&o4=&s=arrhythmia&i=2&h=1000#c`, last accessed on 2023-08-14.

6.4 CEFR Model Performance

The model that had implemented CEFR levels to rank substitutes on simplicity, with model name RB_BSrl_CEFR-all, had obtained inferior results on the test set, as sections 5.1 and 5.2 indicated. This model had used the collective CEFR database in which all used CEFR datasets had been integrated, discussed in section 4.4.2.

To some extent, the insufficient effectiveness of this model might be attributed to my observation in section 6.2 that semantic similarity seemed to have played a more significant role for the gold labels provided with this Shared Task than their notions of simplicity. Further elements that could have contributed to the inferior performance of this model are discussed in the subsequent sections. Note that the CEFR levels referenced in the following sections are sourced from the collective CEFR database in which all used CEFR datasets had been integrated, introduced in section 4.4.2.

6.4.1 Measures of Variation

To obtain insights about how simplicity in terms of CEFR levels would vary within and across complex words, gold labels, and predicted substitutes[2], I performed a number of statistical analyses regarding the variation in CEFR levels within and across these distributions. After assigning CEFR levels to the lemmatized words[3] in these distributions, I determined the mean (average), median, and standard deviation for each of these three distributions. The median — the middle value when sorted — can counteract the effect of outliers, as this measure is not affected by extremely high or low values. The standard deviation reveals how close datapoints cluster around the mean. In a normal distribution[4], approximately 68% of the data lies within one standard deviation on either side of the mean, 95% is situated within two standard deviations, and 99.7% is contained within three standard deviations. In addition, I measured kurtosis, or 'tailedness', which signifies the extent of outliers. I specifically assessed the excess kurtosis, where positive values indicate more extreme values, and negative values mean less of these values than in a normal distribution. Therefore, a 'perfect' normal distribution has an excess kurtosis of 0, also called mesokurtic. Values below -1 and above 1 typically indicate a significant deviation from a normal distribution. The results of my analyses are shown in table 6.1.

	Mean	Median	Standard Deviation	Excess Kurtosis
Complex Word	4.32	4.0	0.72	0.42
Gold Label	3.57	3.63	0.91	-0.12
Prediction	3.66	3.73	0.86	0.20

Table 6.1: measures of variation (test set), for complex words, gold labels, and predictions from model RB_BSrl_CEFR-all; numeric values mapped to CEFR levels (1: A1, 2: A2, 3: B1, 4: B2, 5: C1, 6: C2).

The means of these three distributions show that the average complexity of words is higher for complex words than for the gold and predicted labels. For a Lexical Simplification task, this is an expected result, although the CEFR value differences

[2]for the model using CEFR levels in the SR step, named RB_BSrl_CEFR-all.

[3]for those words to which a CEFR level could be attributed, which will be reviewed in section 6.4.3.

[4]a well-known concept in statistics, visualized by a bell-shaped curve, below which datapoints are situated.

between the Complex Word distribution on the one hand, and the Gold Label and Prediction distributions on the other, are considerably small. When mapped to CEFR levels, the Complex Word distribution average would correspond to a B2-C1 level, although closer to B2, whereas the Gold Label and Prediction distribution averages would each align with a B1-B2 level. The medians are quite close to their respective means, suggesting that each average represents the middle value of its distribution, especially for the gold and predicted labels. Regarding the spread of the CEFR levels around the mean, about 95% of the values in the Complex Word distribution falls within the interval of 2.88 to 5.76, relating to a range from nearly B1 to nearly C2. For the Gold Label distribution, the comparable range is from 1.75 (nearly A2) to 5.39 (between C1 and C2). This is quite similar to the range for the Prediction data, that ranges from 1.94 (nearly A2) to 5.46 (between C1 and C2).

These insights show that the differences in CEFR levels between the complex words on the one hand, and the gold labels and predicted substitutes on the other, are considerably small for a task designed for providing simpler words. For the complex word and prediction datasets, the excess kurtosis is somewhat above 0, suggesting that the distributions have marginally heavier tails than a normal distribution. This means that extreme values are a little more frequent. The gold label dataset shows an excess kurtosis slightly below 0, with extreme values a little less frequent. However, all these values comfortably lie within the discussed range of -1 to 1, and should thus not be considered significant deviations from a normal distribution.

In conclusion, the results suggest that the distributions are nearly normal, and that both the gold labels and the predicted substitutes are, overall, indeed simpler — as far as CEFR levels — than the complex word, although to a relatively small extent.

6.4.2 Gold Label Ranking

The measures of variation discussed in the preceding section provided information that the Gold Label and Prediction distributions were quite similar in terms of CEFR levels, and both distributions showed substitutes that generally had lower CEFR levels than the Complex Word distribution. Therefore, this statistical analysis had not given any indication for the insufficient effectiveness of the RB_BSrl_CEFR-all model.

Although this particular analysis was required as a first step, to confirm that the gold and the predicted labels had consistent CEFR level distributions across both datasets, it does not provide information about the rankings of the individual labels for each instance in the dataset. Since the performance metrics for the Shared Task, explained in section 3.6, use rankings to determine the effectiveness of a model, I delved into the ranking process.

Whereas the gold labels were ranked on frequencies, the predicted substitutes were ranked on CEFR levels. Therefore, I examined how the gold label sequences based on their frequencies would align with their CEFR levels. This investigation could provide valuable insights into the degree of correspondence between the ranking of gold labels on (descending) frequency and their ranking on (ascending) CEFR levels. To enable such comparison, my inspection considered instances for which a minimum of two unique gold labels were represented within the CEFR database, resulting in a reduction of the initial 373 instances to 223.

Table 6.2 provides a comparison between the first and the last gold label that had been assigned CEFR levels. The distribution reveals a nearly balanced presence of instances for which the first CEFR-labeled gold label had either a higher (48,0 %) or a

	Instances	% (of 223 instances)
first gold label has lower CEFR than the last	110	49,3%
first gold label has higher CEFR than the last	107	48,0%
first gold label has equal CEFR level to the last	6	2,7%
Totals	223	100,0%

Table 6.2: Gold labels (test set), assessed on CEFR levels, for instances that have least two unique gold labels with a CEFR level assigned.

lower (49,3%) CEFR level than the last CEFR-labeled gold label. The data seems to indicate that there is no systematic link between the CEFR levels of the most and least frequently suggested gold labels. Given that the ACC@1 metric evaluates the prediction accuracy of the most frequently suggested gold label, this lack of relationship seems to be directly responsible for the inferior performance of the model that consistently places substitutes with the lowest CEFR levels at the top of its ranking lists. It is important to note, however, that not all gold labels could be assigned a CEFR level, given their absence from the database, discussed earlier in this section. On the contrary, the ACC@1 metric considers the most frequently suggested gold labels regardless of whether they have a CEFR level assigned. Consequently, while there appears to be an impact on the model's performance on the ACC@1 metric due to the above-mentioned lack of relationship, a causal association should not be inferred: the most frequently suggested gold label is not necessarily the first CEFR-labeled gold label.

Table 6.3 provides information about a more detailed ordering of the gold labels on CEFR levels.

	Instances	% (of 223 instances)
first gold label has lowest CEFR level of all	43	19,3%
all gold labels have ascending CEFR levels	11	2,9%

Table 6.3: Gold labels (test set), assessed on CEFR levels, for instances that have least two unique gold labels with a CEFR level assigned.

A mere 19.3% of the instances show their top-ranked CEFR-labeled gold label as the lowest among all its CEFR-labeled gold labels, implying that the vast majority of instances do not adhere to an ordering framework based on ascending CEFR levels. Furthermore, in only 2.9% of the instances do all CEFR-labeled gold labels align with a sequential order of ascending[5] CEFR levels. This observation, in line with the findings from table 6.2, suggests a lack of systematic relationship between the frequency-based (descending) ordering of gold labels and an order on (ascending) CEFR levels. Nonetheless, this interpretation shares the same constraints as mentioned when discussing the data from table 6.2, i.e., the issues related to CEFR level coverage.

In the following section, I reflect on the extent of these constraints and propose ways to overcome them.

6.4.3 CEFR Level Coverage

One aspect that might have imposed additional constraints on the effectiveness of the RB_BSrl_CEFR-all model may be the fact that the majority of the provided substitutes — i.e., six out of ten, on average — did not find representation within the CEFR

[5]or equivalent, for those subsequent gold labels with equal CEFR values.

database. As substitutes labeled with CEFR levels were prioritized in the ranking process, the remaining substitutes were positioned in the lower ranks. This methodology may particularly distort simplification rankings for a complex word for which only one (or very few) simplifications are represented within the CEFR database. The model will invariably place this substitute at the highest rank. If this substitute would accidentally carry a relatively high CEFR level, it might either exhibit a higher complexity level than a valid substitute not found in the CEFR database, or even exceed the complexity of the original complex word.

These issues could be mitigated by extending the CEFR-labeled words in the database. For example, CEFR levels for words may be derived from high-quality, CEFR-graded language resources such as EFCAMDAT (Davies, 2009; Shatz, 2020). The weighted averages of word frequencies distributed across the CEFR-graded texts could be computed, comparable to how I executed this for the EFLLEX database discussed in section 4.4.2. Furthermore, fine-tuning an MLM on these resources — considering the words in their individual contexts — may yield the advantages similar to those discussed for WordNet's semantic structure in section 6.3. The enhanced linguistic comprehension of the fine-tuned model might improve the generation of valid substitutes, without requiring additional reference to a CEFR-labeled dataset. This capability could be beneficial for assigning CEFR levels to new or rare words.

Regarding the risk where the complexity level of the complex word would be exceeded by a substitute, this situation could be resolved by exclusively considering substitutes that hold CEFR levels lower than that of the complex word. Crucially, this requires the presence of the complex word in the CEFR database. For this purpose, I extracted the number of (lemmatized) complex words in the test set that were not listed in the CEFR database. Out of the 373 complex words, 253 of them were not listed, corresponding to a total of 68% of the complex words. I inspected these complex words, which seemed highly infrequent words with a high level of complexity, such as 'detonating', 'adamantly', 'insurgents', 'enactment', 'impugned', and numerous others. This observation illustrates why such complex words may often not be listed in CEFR-labeled vocabularies. The purpose of these language resources is to provide learners with the most useful words to learn at different proficiency levels, often being high-frequency words that can be used in different contexts. Consequently, I advocate for assigning complex words not listed in CEFR-labeled resources a default high CEFR level. This would enable considering simplifications beneath that level, in alignment with my proposition presented at the start of this paragraph.

6.5 Beyond ACC@1

As elaborated in section 3.9, the Shared Task results were ranked according to the ACC@1 metric, evaluating whether the highest-ranked predicted substitute was found within the gold label list. Consequently, I had also evaluated my experiments on this measure. In hindsight however, I identified substantial limitations inherent to using ACC@1 as the primary metric for ranking system outcomes.

I had pointed out in section 3.1 that the gold labels were composed of 25 annotations per complex word. This factually means that if a single annotator among the 25 had proposed a unique simplification, not shared by any other of the 25 annotators, a system predicting the same simplification as its best candidate would still be deemed successful due to the design of the ACC@1 metric. Individual variability in

annotator judgments might thus highly influence ACC@1 metric results, potentially rewarding system outputs that align with outlier annotations rather than with simplifications achieved by consensus among annotators. Therefore, this scenario challenges the credibility of ACC@1 as an unbiased performance metric. Regarding this particular Shared Task, as pinpointed in sections 3.2 and 3.3, the execution of the annotation task encompassed several elements that might have intensified these personal biases. This additionally strengthens the unsuitability of a metric influenced by individual annotation variations as the primary evaluation measure in this particular Shared Task.

Furthermore, there is an additional risk involved. The ACC@1 metric only focuses on the top-ranked prediction, overlooking potentially valuable simplifications that are not ranked highest. This observation may have hindered a true assessment of each system's capability to generate a range of valid substitutes. The different perceptions people have of the meaning of the word 'simple', introduced in the opening section of the first chapter, underscore the importance of evaluating Lexical Simplification Systems on their ability to provide several simplification alternatives.

These observations put UNiHD's exceptional ACC@1 score (0.8096) into a new perspective. In approximately 81% of the instances, its top-ranked substitutes corresponded with at least one of the annotations in the list of 25, which could as well be just a single one. When considering the ACC@1top1 metric, a stricter measure which assesses whether the top-ranked substitute equals the most frequently suggested annotation, UniHD's system performance dropped to 0.4289. This indicates that in about 57% of the cases, it failed to align its top-ranked substitute with the majority vote of the annotators. Since UniHD's inferior score on the ACC@1top@1 metric was still the highest of all 33 submitted systems, it is clear that the goal of aligning the top-ranked substitute with the most frequently suggested annotation presented a substantial challenge in this particular Shared Task.

In light of the new perspectives my research provides regarding the Shared Task's contributions to Lexical Simplification, I advocate for more carefulness in determining appropriate metrics for future tasks. A more rigorous metric than ACC@1 should be the prime metric, to challenge systems to strive for alignment with the majority of annotations, while also generating a variety of suitable simplifications to accommodate the fact that there can be several 'best' simplifications for a comple word.

Consequently, I suggest a greater emphasis on the above-mentioned ACC@1@top1 metric, which can serve as a general measure of a system's capability to align its top-ranked prediction with the majority vote of the annotators, thereby minimizing the influence of individual biases. However, since the ACC@1@top1 metric assesses substitutes only against the most frequent annotation, it does not evaluate a system's ability to generate a range of valid substitutes. Therefore, I suggest complementing this metric with a novel measure that assesses a system's ability to predict multiple valid simplifications, but that minimizes the influence of individual annotator bias. The proposed metric, MAP@X@topY, matches its top X predictions against the top Y most commonly proposed annotations. Only instances where all top X predictions are present in the top Y most frequent annotations will be evaluated as successful in this measure. The values of X and Y should be derived from the precise objective of the Lexical Simplification task. This objective should be grounded on the requirements of the intended target audience for whom the task is designed. These requirements should be reflected in clear Annotation Guidelines and subsequently in an adequate number of annotations per complex word.

Given the preceding discourse, I propose a discontinuation of the use of not only ACC@1 (=Potential@1 and MAP@1, discussed in section 3.6), but also Potential@3, Potential@5, and Potential@10 in upcoming Lexical Simplification tasks. These Potential metrics, being less strict than the — already not strict — ACC@1 metric, rely on the proportion of instances where a minimum of one of the top-K ranked predictions appears in the list with gold labels. With K =5, for example, the Potential metric (Potential@5) calculates the proportion of instances that have minimally one of the five highest-ranked predictions appearing in the gold label list. Just like ACC@1, the Potential metrics are influenced by individual annotation variations. Even more than for ACC@1, these metrics offer very limited insights into a system's effectiveness to simplify words. In addition, I suggest discontinuing the MAP@3, MAP@5, and MAP@10 metrics. Although these metrics are more strict for predicted substitutes — as the MAP@K metric evaluates the proportion of instances for which *all* top-K ranked predictions are listed as annotations — these measures are still influenced by individual annotation variations.

Summarizing, I propose using only two primary evaluation metrics for future Lexical Simplification tasks: ACC@1@top1, measuring a system's capability to align its top-ranked prediction with the most frequently suggested annotation, and MAP@X@topY, a newly suggested metric evaluating a system's capability to predict multiple valid substitutes with a minimized influence of individual annotator bias.

Finally, I would advise organizers of future Lexical Simplification tasks to communicate the primary metric(s) intended for the ranking of results prior to the development of the simplification systems. Concurrently, a clear definition of the target audience and its perception of 'simpler' should be provided, as advocated in section 6.2. Providing early insights into a task's main objectives helps participants to design their systems accordingly.

6.6 EDIA's Readability Analyzer Papyrus

In the opening chapter, I highlighted the primary aim of my thesis project, i.e., to contribute to the ongoing quest for enhancing text comprehension. In the present section, I seek to translate this objective into practical solutions by discussing how the discoveries from my study may advance the capabilities of EDIA's readability analyzer Papyrus, introduced in section 1.2. Next to proposing methodologies for the English language, I suggest how these may be generalized towards the Dutch language. The latter proposition is intended to strengthen EDIA's current initiative to assist the Dutch government in helping citizens with language deficiencies.

6.6.1 System Design Recommendations for English

As elaborated in section 1.2, EDIA's readability analyzer Papyrus could potentially be improved on the way it identifies semantically similar alternatives for complex words. Based on my findings, I propose strategies which could potentially enhance Papyrus' capabilities in this domain.

Considering the architecture of Papyrus, grounded on BERT base (cased) variant[6], one significant enhancement for the Substitute Generation (SG) step may involve supplementing this model with the context of the original sentence including the complex

[6] `https://huggingface.co/bert-base-cased`, last accessed on 2023-08-14.

word, in conjunction with the sentence in which the complex word is masked. My experiments showed that providing my models with this additional contextual clue resulted in improved scores on the Shared Task, as highlighted in table 4.2 in section 4.1 for the trial set and in table 5.4 in section 5.4 for the test set. The advantage of this approach is further visualized in the example provided in table 4.1 in section 4.1. In the absence of the complex word situated within its context, the model yields a range of alternatives that align with the context of the sentence, however unaware of the complex word's meaning. Conversely, upon introducing the model to the complex word in its specific context, the model is informed to specifically generate words that could accurately reflect the meaning of the complex word within that context.

Furthermore, considering the superior performance of RoBERTa's base variant in my experiments, EDIA could consider replacing their BERT base variant with RoBERTa's base variant. However, this modification may not prove as substantial as the above-mentioned contextual enhancement.

Concerning the SS step, Papyrus uses word embeddings sourced from SpaCy[7], a Python library equipped with a variety of features for NLP tasks. The word embeddings generated by SpaCy are not contextualized, signifying that each word is mapped to one single numerical vector that represents the meaning of that word across all contexts. As explained in section 2.2, this approach encounters considerable limitations when dealing with words that have multiple meanings — polysemous words — which are inherently context-dependent. Consequently, I advise replacing Papyrus' non-contextualized embeddings by contextualized embeddings. Contextualized embeddings, unlike their non-contextualized counterparts, can take the surroundings of words into account, allowing to capture polysemous words in their context. As outlined in section 4.3.3, BERTScore employs contextualized embeddings. The implementation of the BERTScore mechanism in my top-performing model prior to post-evaluation resulted in notably higher scores on the Shared Task, visualized by the model deconstruction table 5.4 in section 5.4.

Papyrus performs the SR step by assigning CEFR levels to words, although it does not rank the words based on these levels. In Papyrus, a target CEFR level can be set, and the model returns all substitutes that fall below that target level. As a result, this process primarily functions as a filtration mechanism rather than as a ranking method. Additionally, substitutes that are not present in EDIA's CEFR database are included in the model's output. To alleviate the issue of coverage, EDIA could consider my suggestion in section 6.4.3 that advocates increasing the number of CEFR-labeled words. To realize this objective, additional databases could be leveraged. For instance, CEFR levels for words could be inferred from CEFR graded linguistic resources. Fine-tuning the BERT model on which Papyrus is based on such resources that encompass words within their specific contexts may improve the generation of substitutes without the necessity of directly referencing the CEFR dataset. This strategy could be instrumental in assigning CEFR levels to new or infrequently used words. However, some of these graded resources, including the EFCAMDAT database mentioned in section 6.4.3, may not be used for commercial purposes. This restriction also applies to the majority of the CEFR-labeled databases that I incorporated in my models. These resources are exclusively intended for academic use and will not be shared with EDIA.

[7] `https://spacy.io/`, last accessed on 2023-08-14.

6.6.2 Generalization to Dutch

To strengthen EDIA's current initiative to assist the Dutch government in helping citizens with language deficiencies, I propose a set of methodologies that may enable generalization of my model design from English to Dutch. I conclude this section by discussing challenges with regard to the implementation in this language.

System Design Recommendations

For the Dutch language, Papyrus uses a multilingual BERT[8] model for the execution of the SG step. EDIA could consider comparing the performing of this model with monolingual MLMs adapted to the Dutch language such as BERTje. This model outperformed its multilingual counterpart on a variety of Dutch NLP tasks (de Vries et al., 2019). Subsequently, BERT's successor RobBERT — a RoBERTa-based pre-trained Dutch MLM —outperformed BERTje, especially when fine-tuned on small datasets (Delobelle et al., 2020). Similar to the methodology recommended for English, I advise complementing the chosen model with the context of the original unmasked sentence.

For the SS step, BERTScore's contextualized embeddings could be a valid choice, as suggested for English. Its technical compatibility extends to a range of languages[9] including Dutch. As a potential alternative or as a complementary tool to BERTscore, EDIA could consider exploring Open Dutch WordNet[10] (Postma et al., 2016), a Dutch lexical semantic database modelled on the WordNet structure. An investigation into the usecase of shared synonyms and hypernyms for the Dutch language, similar to my explorations in section 4.3.1 and 4.3.2 for English, might yield benefits for selecting semantically similar substitutes during the SS step.

Regarding the SR step, Papyrus does not specifically rank its Dutch substitutes on CEFR level, equal to its English version. It may be worthwile to address existing CEFR level coverage issues by investigating the existence of CEFR-graded language resources in the Dutch language that contain words in their specific contexts, and fine-tuning the model on such resources, as discussed for English.

Implementation Challenges

Implementing Dutch Lexical Simplification systems presents distinct challenges that appear in most, if not all, NLP tasks that pertain to this language.

One potential limitation involves the reduced effectiveness of Dutch MLMs when contrasted with their English equivalents. This can largely be ascribed to the scarce availability of training data in Dutch. With fewer datasets to learn from, Dutch NLP models may fail in matching the proficiency of their English counterparts. If there is less data available, the model has less opportunities to learn the language structure, vocabulary, and its nuances. This impacts its usefulness in the real world. Moreover, the lack of Dutch datasets may also negatively influence the progression of Dutch NLP tasks that rely on these datasets for training and evaluation.

The language differences between Dutch and English also form an issue. A considerable challenge is the morphological divergence between English and Dutch. English is generally considered an analytic language, as it uses many separate words and relatively few inflections. Dutch is more synthetic, applying more inflections to indicate

[8] `https://huggingface.co/bert-base-multilingual-cased`, last accessed on 2023-08-14.
[9] `https://huggingface.co/spaces/evaluate-metric/bertscore`, last accessed on 2023-08-14.
[10] `https://github.com/cltl/OpenDutchWordnet`, last accessed on 2023-08-14.

grammatical relations or meaning differences between words. In the paragraphs below, I discuss a selection of these morphological differences.

The first morphological divergence also influences syntactic structures. The Dutch language uses a broad range of separable verbs, composed of a base form and a prefix. This prefix can be disassociated from the base form and relocated within the sentence, depending on its grammatical structure and occasional nuances in meaning. An illustrative example is the Dutch verb 'aankomen', corresponding to 'arrive' in English, with 'komen' as the base form and 'aan' as the prefix. In Dutch, if the complex word would be 'arriveert' (equivalent to 'arrives' in English), a highly similar and simpler substitute would be 'komt aan'. However, in Dutch syntax, adverbials may be positioned between these two verb forms. The English sentence "He [arrives] in Amsterdam" can be translated into Dutch as either "Hij [komt] in Amsterdam [aan]" or "Hij [komt] [aan] in Amsterdam". The two sentences offer subtle semantic distinctions. Where the first puts more emphasis on the location ('in Amsterdam'), the second leans more towards emphasizing the action of arriving ('komt aan'). The potential variability in the placement of the prefix may present challenges for correctly identifying the combination of the base form and the detached prefix together as one valid substitute. Only performing lexical simplification of verbs without knowledge of the Dutch grammatical structure will, most likely, generate inferior results.

Another characteristic of the Dutch morphology is the tendency to form compound words. For example, Dutch compound words can be assembled from verbs, such as 'leesbril' (reading glasses), from 'lees' (read) and 'bril' (glasses). Furthermore, subsequent nouns are by default consolidated into a single word, with adjectives appended. An example is 'kortetermijngeheugen', translating to 'short term memory' in English, formed by combining 'korte' (short), 'termijn' (term), and 'geheugen' (memory). While the majority of Dutch compound words typically comprise two or three constituents, the syntax governing their composition permits the formation of atypical words constructed from numerous parts. These seemingly infinite constructions could pose a difficulty when training a computational model, especially considering that Dutch - as previously mentioned - is relatively resource-poor compared to English.

A different issue is the distinction in the Dutch language between the gender of nouns associated with definite articles, a characteristic absent in English. The Dutch definite articles 'de' and 'het' both translate to the definite article 'the' in English. While 'de' is used for masculine and feminine nouns, 'het' is assigned to neutral nouns. Within the scope of Dutch Lexical Simplification, valid substitutes may not necessarily have the same gender as the complex word. Therefore, Dutch Lexical Simplification systems should be capable of changing the definite article associated with the complex word into the definite article of the substitute. This criterion should also apply to the annotations upon which the evaluations of these systems are based. An example is the Dutch word 'discrepantie', derived from the Latin word 'discrepantia', also used in English (discrepancy). Simplifying this female noun into the neutral noun 'verschil' (in English: 'difference', its direct hypernym[11] in English WordNet) should involve the possibility to change a definite article from 'de' to 'het' along with 'verschil'.

As denoted earlier in this section, the above Dutch language characteristics are limited to a selection. Research on the Dutch morphology in relation to Lexical Simplification may benefit the Dutch version of EDIA's readability analyzer Papyrus.

[11] `http://wordnetweb.princeton.edu/perl/webwn?o2=&o0=1&o8=1&o1=1&o7=&o5=&o9=&o6=&o3=&o4=&s=difference&i=1&h=1000000000#c`, last accessed on 2023-08-14.

Chapter 7

Conclusions

This work explored various methodologies regarding English Lexical Simplification, adhering to the requirements for the TSAR-2022 Shared Task on Multilingual Lexical Simplification. It focused on the consecutive stages of generating, selecting, and ranking substitutes for given complex words. Throughout the course of these stages, 62 models based on the Masked Language Model (MLM) technology were initially developed. Five were evaluated on the test set, based on predefined criteria. They used both the context of the original and the masked sentence, thereby generating more semantically accurate substitutes. Two models outperformed TSAR-LSBert, a recent benchmark. This was achieved with RoBERTa's base variant for Substitute Generation and BERTScore computations with RoBERTa's large variant for Substitute Selection. The model had no additional simplicity ranking method implemented, implying a need for further studies about the meaning of 'simplicity' in this context.

Understanding how different reading audiences perceive simplicity is essential for annotators and system designers in Lexical Simplification tasks. Carefully tailoring their respective task instructions to audience needs is recommended. The instructions for system designers should include the principal evaluation metric(s). For this purpose, a new evaluation metric is suggested, evaluating systems on their capability to predict multiple valid substitutes while minimizing individual annotator bias.

The inferior performance of a model using CEFR levels to rank its substitutes might be caused by the lack of a systematic relationship between frequency-based and CEFR-level-based gold label rankings. Future research is required to confirm this.

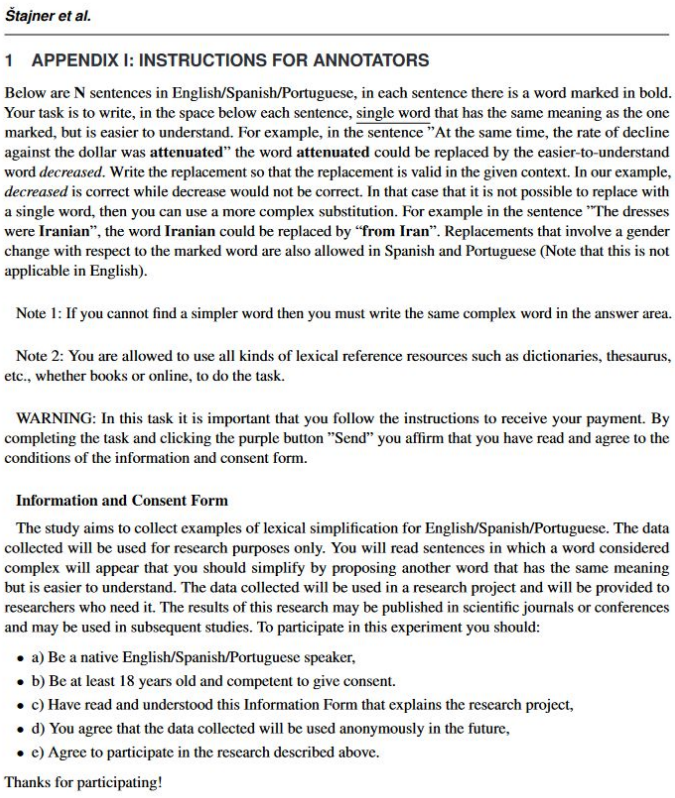
Post-evaluation experiments resulted in a notable performance boost with eight newly tested models, three of which surpassed in their rankings only by two GPT-based models. The models leveraged WordNet's semantic structure to assess whether substitutes served as hypernyms for the complex word, an innovative approach in Lexical Simplification. The promising results underscore the potential of hybrid methods that combine MLMs with supervised high-quality linguistic resources. Future work could further harness WordNet, possibly in combination with Basic Level Categories.

This study's findings were applied to potentially augment EDIA's readability analyzer Papyrus for English and Dutch, although the Dutch version may encounter obstacles due to resource scarcity and the characteristics of the Dutch morphology.

As this thesis comes to a close, I revisit my principal objective: to aid the ongoing pursuit of improving text comprehension. As every advancement in this field relies on shared efforts, this study on Lexical Simplification will bring us closer to alignment with the United Nations' goal of enabling everyone to access and understand information.

Appendix A

Annotation Guidelines



Štajner et al.

1 APPENDIX I: INSTRUCTIONS FOR ANNOTATORS

Below are **N** sentences in English/Spanish/Portuguese, in each sentence there is a word marked in bold. Your task is to write, in the space below each sentence, single word that has the same meaning as the one marked, but is easier to understand. For example, in the sentence "At the same time, the rate of decline against the dollar was **attenuated**" the word **attenuated** could be replaced by the easier-to-understand word *decreased*. Write the replacement so that the replacement is valid in the given context. In our example, *decreased* is correct while decrease would not be correct. In that case that it is not possible to replace with a single word, then you can use a more complex substitution. For example in the sentence "The dresses were **Iranian**", the word **Iranian** could be replaced by "**from Iran**". Replacements that involve a gender change with respect to the marked word are also allowed in Spanish and Portuguese (Note that this is not applicable in English).

Note 1: If you cannot find a simpler word then you must write the same complex word in the answer area.

Note 2: You are allowed to use all kinds of lexical reference resources such as dictionaries, thesaurus, etc., whether books or online, to do the task.

WARNING: In this task it is important that you follow the instructions to receive your payment. By completing the task and clicking the purple button "Send" you affirm that you have read and agree to the conditions of the information and consent form.

Information and Consent Form

The study aims to collect examples of lexical simplification for English/Spanish/Portuguese. The data collected will be used for research purposes only. You will read sentences in which a word considered complex will appear that you should simplify by proposing another word that has the same meaning but is easier to understand. The data collected will be used in a research project and will be provided to researchers who need it. The results of this research may be published in scientific journals or conferences and may be used in subsequent studies. To participate in this experiment you should:

- a) Be a native English/Spanish/Portuguese speaker,
- b) Be at least 18 years old and competent to give consent.
- c) Have read and understood this Information Form that explains the research project,
- d) You agree that the data collected will be used anonymously in the future,
- e) Agree to participate in the research described above.

Thanks for participating!

Figure A.1: Annotation Guidelines for English track of TSAR-2022 Shared Task on Multilingual Lexical Simplification, taken from Stajner et al. (2022)

Bibliography

D. Aleksandrova and O. Brochu Dufour. RCML at TSAR-2022 shared task: Lexical simplification with modular substitution candidate ranking. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 259–263, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.29.

A. P. Aprosio, S. Menini, S. Tonelli, L. Ducceschi, and L. Herzog. Towards personalised simplification based on l2 learners' native language. In *Italian Conference on Computational Linguistics*, 2018.

N. Arefyev, B. Sheludko, A. Podolskiy, and A. Panchenko. Always keep your target in mind: Studying semantics and improving performance of neural lexical substitution. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1242–1255, Barcelona, Spain (Online), Dec. 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.107. URL https://aclanthology.org/2020.coling-main.107.

D. Aumiller and M. Gertz. UniHD at TSAR-2022 shared task: Is compute all we need for lexical simplification? In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 251–258, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.28.

P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. ISSN 2307-387X.

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. *CoRR*, abs/2005.14165, 2020. URL https://arxiv.org/abs/2005.14165.

M. Brysbaert and B. New. Moving beyond kucera and francis: A critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for american english. *Behavior research methods*, 41:977–90, 11 2009. doi: 10.3758/BRM.41.4.977.

E. Chersoni and Y.-Y. Hsu. PolyU-CBS at TSAR-2022 shared task: A simple, rank-based method for complex word substitution in two steps. In *Proceedings of the*

Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022), pages 225–230, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.24.

K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1xMH1BtvB.

M. Davies. The 385+ million word corpus of contemporary american english (1990–2008+): Design, architecture, and linguistic insights. *International Journal of Corpus Linguistics*, 14(2):159–190, 2009. ISSN 1384-6655. doi: https://doi.org/10.1075/ijcl.14.2.02dav. URL https://www.jbe-platform.com/content/journals/10.1075/ijcl.14.2.02dav.

W. de Vries, A. van Cranenburgh, A. Bisazza, T. Caselli, G. van Noord, and M. Nissim. Bertje: A dutch bert model, 2019.

P. Delobelle, T. Winters, and B. Berendt. Robbert: a dutch roberta-based language model. *CoRR*, abs/2001.06286, 2020. URL https://arxiv.org/abs/2001.06286.

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)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

L. Dürlich and T. François. EFLLex: A graded lexical resource for learners of English as a foreign language. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1140.

C. Fellbaum. A semantic network of english verbs. *WordNet: An electronic lexical database*, 3:153–178, 1998.

D. Ferrés, H. Saggion, and X. Gómez Guinovart. An adaptable lexical simplification architecture for major Ibero-Romance languages. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 40–47, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5406. URL https://aclanthology.org/W17-5406.

J. Ganitkevitch, B. Van Durme, and C. Callison-Burch. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764, Atlanta, Georgia, June 2013. Association for Computational Linguistics. URL https://aclanthology.org/N13-1092.

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), pages 63–68, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-2011. URL https://aclanthology.org/P15-2011.

S. Gooding and E. Kochmar. Recursive context-aware lexical simplification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4853–4863, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1491. URL https://aclanthology.org/D19-1491.

S. Havens and A. Stal. Use bert to fill in the blanks, 2019. URL https://github.com/Qordobacode/fitbert.

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)*, pages 458–463, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-2075. URL https://aclanthology.org/P14-2075.

N. Katyal and P. K. Rajpoot. CENTAL at TSAR-2022 shared task: Lexical simplification using multi-level and modular approach. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 239–242, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.25.

J. Lee and C. Y. Yeung. Personalizing lexical simplification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 224–232, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://aclanthology.org/C18-1019.

X. Li, D. Wiechmann, Y. Qiao, and E. Kerz. MANTIS at TSAR-2022 shared task: Improved unsupervised lexical simplification with pretrained encoders. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 243–250, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.27.

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. *ArXiv*, abs/1907.11692, 2019.

C. Mills, F. Bond, and G.-A. Levow. Automatic identification of basic-level categories. In *Proceedings of the 9th Global Wordnet Conference*, pages 298–305, Nanyang Technological University (NTU), Singapore, Jan. 2018. Global Wordnet Association. URL https://aclanthology.org/2018.gwc-1.35.

K. North, A. Dmonte, T. Ranasinghe, and M. Zampieri. GMU-WLV at TSAR-2022 shared task: Evaluating lexical simplification models. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 264–270, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.30.

OECD. *Skills Matter, Additional Results from the Survey of Adult Skills*. 2019. URL https://www.oecd.org/skills/piaac/publications/Skills_Matter_Additonal_Results_from_the_Survey_of_Adult_Skills_ENG.pdf.

C. Orasan, R. Evans, and R. Mitkov. *Intelligent Text Processing to Help Readers with Autism*. Cham: Springer International Publishing, 2018.

G. Paetzold and L. Specia. LEXenstein: A framework for lexical simplification. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 85–90, Beijing, China, July 2015. Association for Computational Linguistics and The Asian Federation of Natural Language Processing. doi: 10.3115/v1/P15-4015. URL https://aclanthology.org/P15-4015.

G. Paetzold and L. Specia. Unsupervised lexical simplification for non-native speakers. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), Mar. 2016a. doi: 10.1609/aaai.v30i1.9885. URL https://ojs.aaai.org/index.php/AAAI/article/view/9885.

G. Paetzold and L. Specia. SemEval 2016 task 11: Complex word identification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 560–569, San Diego, California, June 2016b. Association for Computational Linguistics. doi: 10.18653/v1/S16-1085. URL https://aclanthology.org/S16-1085.

G. Paetzold and L. Specia. A survey on lexical simplification. *Journal of Artificial Intelligence Research*, 60:549–593, 11 2017a. doi: 10.1613/jair.5526.

G. Paetzold and L. Specia. Lexical simplification with neural ranking. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 34–40, Valencia, Spain, Apr. 2017b. Association for Computational Linguistics. URL https://aclanthology.org/E17-2006.

E. Pavlick and C. Callison-Burch. Simple PPDB: A paraphrase database for simplification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 143–148, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-2024. URL https://aclanthology.org/P16-2024.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL https://aclanthology.org/N18-1202.

M. Postma, E. van Miltenburg, R. Segers, A. Schoen, and P. Vossen. Open Dutch WordNet. In *Proceedings of the 8th Global WordNet Conference (GWC)*, pages 302–310, Bucharest, Romania, 27–30 Jan. 2016. Global Wordnet Association. URL https://aclanthology.org/2016.gwc-1.43.

J. Qiang, Y. Li, Y. Zhu, Y. Yuan, Y. Shi, and X. Wu. Lsbert: Lexical simplification based on bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3064–3076, 2021. doi: 10.1109/TASLP.2021.3111589.

E. Rosch, C. B. Mervis, W. D. Gray, D. M. Johnson, and P. Boyes-Braem. Basic objects in natural categories. *Cognitive Psychology*, 8(3):382–439, 1976. ISSN 0010-0285. doi: https://doi.org/10.1016/0010-0285(76)90013-X. URL https://www.sciencedirect.com/science/article/pii/001002857690013X.

H. Saggion, S. Štajner, D. Ferrés, K. C. Sheang, M. Shardlow, K. North, and M. Zampieri. Findings of the tsar-2022 shared task on multilingual lexical simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 271–283, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.31.

S. Seneviratne, E. Daskalaki, A. Lenskiy, and H. Suominen. CILex: An investigation of context information for lexical substitution methods. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4124–4135, Gyeongju, Republic of Korea, Oct. 2022a. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.362.

S. Seneviratne, E. Daskalaki, and H. Suominen. CILS at TSAR-2022 shared task: Investigating the applicability of lexical substitution methods for lexical simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 207–212, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022b. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.21.

I. Shatz. Refining and modifying the efcamdat: Lessons from creating a new corpus from an existing large-scale english learner language database. *International Journal of Learner Corpus Research*, 6(2):220–236, 2020. ISSN 2215-1478. doi: https://doi.org/10.1075/ijlcr.20009.sha. URL https://www.jbe-platform.com/content/journals/10.1075/ijlcr.20009.sha.

S. Stajner. Automatic text simplification for social good: Progress and challenges. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2637–2652, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.233. URL https://aclanthology.org/2021.findings-acl.233.

S. Stajner, D. Ferrés, M. Shardlow, K. North, M. Zampieri, and H. Saggion. Lexical simplification benchmarks for english, portuguese, and spanish. *Frontiers in Artificial Intelligence*, 5, 2022. ISSN 2624-8212. doi: 10.3389/frai.2022.991242. URL https://www.frontiersin.org/articles/10.3389/frai.2022.991242.

Y. Tono. CEFR-J Wordlist Version 1.3. Tokyo University of Foreign Studies, 2016. URL http://www.cefr-j.org/download.html.

Y. Tono. CEFR-J Wordlist Version 1.5. Tokyo University of Foreign Studies, 2020. URL https://github.com/openlanguageprofiles/olp-en-cefrj/blob/master/cefrj-vocabulary-profile-1.5.csv.

S. Uchida, S. Takada, and Y. Arase. CEFR-based lexical simplification dataset. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1514.

L. Vásquez-Rodríguez, N. Nguyen, M. Shardlow, and S. Ananiadou. UoM&MMU at TSAR-2022 shared task: Prompt learning for lexical simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 218–224, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.23.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

P. Whistely, S. Mathias, and G. Poornima. PresiUniv at TSAR-2022 shared task: Generation and ranking of simplification substitutes of complex words in multiple languages. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 213–217, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.22.

R. Wilkens, D. Alfter, R. Cardon, I. Gribomont, A. Bibal, W. Patrick, M.-C. De marneffe, and T. François. CENTAL at TSAR-2022 shared task: How does context impact BERT-generated substitutions for lexical simplification? In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 231–238, Abu Dhabi, United Arab Emirates (Virtual), Dec. 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.25.

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*, 3:283–297, 2015. doi: 10.1162/tacl_a_00139. URL https://aclanthology.org/Q15-1021.

Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Neural Information Processing Systems*, 2019.

S. M. Yimam, S. Štajner, M. Riedl, and C. Biemann. CWIG3G2 - complex word identification task across three text genres and two user groups. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 401–407, Taipei, Taiwan, Nov. 2017. Asian Federation of Natural Language Processing. URL https://aclanthology.org/I17-2068.

S. M. Yimam, C. Biemann, S. Malmasi, G. Paetzold, L. Specia, S. Štajner, A. Tack, and M. Zampieri. A report on the complex word identification shared task 2018. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 66–78, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-0507. URL https://aclanthology.org/W18-0507.

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. URL https://openreview.net/forum?id=SkeHuCVFDr.

Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books, 2015.