# Sentence Simplification with Pre-trained BART

Xinyi Hu, Jonathan Shaw, Pengyuan Zhang, Zihao Zhang

## Abstract

In this project, we examine language modeling for text simplification. The goal of sentence simplification is to simply grammar and structure to improve the readability of one of more sentences without losing too much underlying information. We adapted a pre-trained BART model as our strong baseline. In addition, we experimented with two extension techniques: lexical simplification and control token. These two modifications to the input sentences have several benefits, such as controlling the length of predictions, to the downstream fine-tuning task. Finally, we concluded that the fine-tuned sequence-to-sequence BART model with two extensions achieved the best performance when compared to strong baseline model and model with only one extension. The BLEU score of our final model is 0.58 and the SARI score is 51.56.

## 1 Introduction

The task of text simplification includes modifying the content and structure of a text so that the resulting text is easier to read and comprehend. Meanwhile, the main idea, as well as the original meaning can be preserved in the text derived from text simplification. A simplified version of text could benefit several groups of people, such as low literacy readers (Watanabe et al., 2009), non-native English speakers (Paetzold, 2016), children (De Belder and Moens, 2010), and people with diseases or disabilities (Devlin and Tait, 1998; Carroll et al., 1998; Rello et al., 2013b; Evans et al., 2014). In addition, text simplification is closely related to several other NLP tasks, including parsing, summarization, information extraction, semantic role labeling, and machine translation. It is claimed and confirmed by several researches that the improvement in text simplification could be easily adapted to these domains.

Sentence simplification is an important subtask of text simplification. Here is one example:

Complex: Originally, the college was to be specifically for boys from Eton College.

Simple: Originally, the college was just for boys from Eton College.

This example is quite simple conversion from complex to simple sentence, both for human and machine comprehension. In the simplified version, the phrase "to be specifically" is replaced with the word "just", which is appropriate both grammatically and semantically. Here is another example:

Complex: ~~King's has a venue known as~~ the Cellar Bar, a small room in the basement of the college, which regularly acts as a music venue

Simple: The Cellar Bar is a small room in the basement of the college, which is used for music

This example is more complicated and requires more attentions. First, the subject of the simplified version changes from "King's" to "The Cellar Bar", which fundamentally alters the sentence structure. In addition, the complex sentence uses a dependent clause to explain what is the Cellar Bar. In contrast, the simplified version expresses the definition more straightforward by giving up the information that the Cellar Bar is a venue and belongs to King's. Finally, both sentences use a relative clause but that in the simplified version is semantically clearer.

More formally, the goal of sentence simplification is to modify a sentence so that the resulting sentence is easier to read and understand. Several rewriting transformations, including replacement, reordering, and splitting, are used to achieve the goal. While executing the transformations are challenging, researchers usually encounter more difficult problems of maintaining the grammatical correctness, preserving the main idea, and generating a simpler output. Modern sentence simplification approaches attempted to learn these transformations using parallel corpora of aligned original-simplified sentences. This results in general simplification models that could be used for any specific type of audience, depending on the data used during training.

On one hand, our motivations are from the facts that nearly 1 in 12 U.S. children aged 3-17 have developmental language disorder. In addition, adults also can sometimes struggle to understand compound sentences. The medical treatment is expensive while limited and more importantly, it requires early intervention to maximize the curative effect. Therefore, with a sentence simplification tool that can modify the content and structure of sentences, people with reading disorders and limited access to medical intervention will feel much easier to still grasp the main point of complex text. On the other hand, we found that sentence simplification is a text generation problem that shares similar idea with Neural MT, which was introduced in class in details. We believe that BART, a pre-trained seq2seq model, could be adapted to sentence simplification domain by only a few epochs of fine-tuning. Finally, one of the evaluation metrics for sentence simplification called BLEU was also introduced in class while we add another one called SARI, which will be explained in **3.2 Evaluation Metrics**.

# 2 Literature Review

## 2.1 Sentence Simplification with Deep Reinforcement Learning

In 2017, Zhang and Lapata proposed a new method for sentence simplification called Deep Reinforcement Sentence Simplification (DRESS). They adopted a traditional encoder-decoder structure inspired by machine translation and combined with a deep reinforcement learning framework, which explores the space of possible simplifications while learning to optimize a reward function that encourages outputs which are simple, fluent and preserve the meaning of the input.

The encoder-decoder baseline is simple attention-based sequence to sequence model with an attention module. But a seq2seq model is too simple that it copies too well from the input when simplifying. So the authors introduced a reinforcement learning mechanism, regarding the seq2seq model as an agent receiving a reward which is a weighted sum of three component: simplicity, relevance and fluency. The simplicity module encourages the model to make changes and make the sentence simple, yielding a weighted sum of SARI and reverse SARI. The relevance module ensures that the generated sentences preserve the meaning of the source, yielding a cosine similarity between the source vector and target vector. The fluency module encourages the model's well-formedness, yielding a normalized sentence probability assigned by an LSTM model. With a training goal to maximize the reward, the model can learn to produce reasonable simplifications with a pre-trained encoder-decoder baseline. Furthermore, the authors proposed learning lexical simplification separately and linearly integrate with the reinforcement learning model, which should yield better results.

The authors conducted experiments on Newsela, WikiSmall, WikiLarge dataset. The results showed that DRESS gave a promising BLEU, FKGL and SARI, while DRESS with explicit lexical simplification (DRESS-LS) significantly improved BLEU with a slightly higher FKGL and SARI in Newsela dataset. In human evaluations, DRESS-LS gives the best fluency and overall performance.

## 2.2 Controllable Sentence Simplification

In 2019, Martin et al. proposed to use a Seq2Seq model to tackle sentence simplification. The dataset used is WikiLarge, similar to previous papers on sentence simplification. What is special about their model, ACCESS, is that it is trained using special parametrization, which allows the users to specify conditions such as length, amount of paraphrasing, lexical complexity, and syntactic complexity. To control these conditions, the author uses learning-based method instead of decoder-based method to have finer control without degrading performances. Each condition is represented by an explicit control token calculated based on the training source and the target; the control token is pre-pended to the source sentence and entered into the Seq2Seq model.

In the end, a combination of specific length, amount of paraphrasing, and lexical complexity lead to the best result with 41.29 SARI, significant improvement from previous score of 40.45. The single best token was WordRank, the token representing lexical complexity. For our paper, I believe SARI to be a good evaluating metric, and if possible, the addition of WordRank as a control token can also be implemented.

## 2.3 Complexity-Weighted Loss and Diverse Reranking for Sentence Simplification

In 2019, Kriz et al. proposed to use two techniques to alleviate the issue that the seq2seq models tend to copy directly from the original sentence. The first technique is to incorporate content word complexities into the loss function during training. A linear regression model was trained with data that were originally from a collection of 1,840 news articles written by professional editors at 5 reading levels (Xu et al., 2015). Then, the cross entropy loss function was improved to up-weight simple words while down-weighting more complex words. To accomplish this goal, the words were sent to the linear regression model to get their corresponding complexities. The complexities were then reversed and multiplied with original probability vector in the point-wise fashion to get the normalized probability vector, which could be converted back to logits for loss calculation. The second technique is to generate a large set of diverse candidate simplifications at test time and rerank them to promote fluency, adequacy and simplicity. The fluency $f_i$ was calculated based on a 5-gram language model trained on English Gigaword v.5 (Parker et al., 2011) using KenLM (Heafield, 2011). The adequacy $a_i$ was calculate with the cosine similarity. The simplicity $s_i$ was calculated by a sentence complexity prediction model that predicted the overall complexity of each generated sentence. Finally, each individual score was normalized to be in [0, 1] and the final score was calculated as $score_i = \beta_f f_i + \beta_a a_i + \beta_s s_i$ where the weights ($\beta$) were tuned on the validation data.

The evaluation included both automatic and human evaluation. The automatic evaluation results showed significant improvement by comparing the simplification derived using models with and without the specific techniques. In addition, when comparing the proposed models to baselines and state-of-the-art models, the proposed S2S-Cluster-FA model achieved the highest score of 37.22 on the SARI dataset. On the other hand, the human evaluation was performed since the the author believed that sentence structure complexity should also be taken into account. The authors concluded that their best models substantially outperform the Hybrid and DMASS systems.

# 3 Experimental Design

## 3.1 Data

The dataset contains 167,689 aligned complex-simple sentence pairs. The complex sentences were parsed from main English Wikipedia while the simple sentences were parsed from simple English Wikipedia that only has basic words. The dataset, Wikipedia Data Sets, Version 2.0, was collected by David Kauchak in 2011 and published in 2013. The data.tar.gz has three documents: train.txt, dev.txt, and test.txt. Each line of the text file contains one complex sentence and one corresponding simple sentence. These two sentences are split by a tab character in order to separate from space characters in each sentence. Note that Simple English Wikipedia is a site targeted at people with learning/reading difficulties, children, and learners of the English language. The dataset encompasses both lexical and syntax simplification of the English language and provides us with a wide coverage with respect to the topics of the sentences.

Later, given the limited computing resources for fine-tuning, 50,000 complex-simple sentence pairs were randomly sampled and split into training, development, and test dataset by a 98/1/1 ratio. The sampled data is then processed to meet the requirements of downstream tasks. The data preprocessing includes removing non-alphanumeric characters and filtering the sentences that have more than 103 words. The statistics of the training dataset is shown in Table 1.

| Avg Inp Length | Avg Out Length | Unique Token | Total Token |
|---|---|---|---|
| 132.39 | 115.79 | 201845 | 5764167 |

Table 1: Statistics of the Training Dataset

Finally, the sentences were sent to Complex Word Identification (CWI) model to identify complex words, generate simple word candidates, and replace the complex words with its simplest simple word candidates. The simplicity ratio token and length ratio token of each sentence were calculated, rounded to 0.05 precision and appended to the front of the sentences. Please refer to **4.2 Extensions** for a detailed explanation of the techniques mentioned above.

## 3.2 Evaluation Metric

Two evaluation metrics, including BiLingual Evaluation Understudy (BLEU) and System output Against References and against the Input sentence (SARI), were experimented to measure the goodness of the simplified sentence.

BLEU is a precision-oriented metric that relies on the proportion of n-gram matches between a system's output and reference(s). BLEU is mathematically defined as following:

$$BLEU = \min(1, \exp(1 - \frac{ref - len}{out - len}))(\prod_{n=1}^{4} p_n)^{\frac{1}{4}}$$

where $ref$ is reference, $len$ is length, $out$ is output, and with

$$p_n = \frac{\sum_{S \in C} \sum_{ngram \in S} Count_{matched}(ngram)}{\sum_{S \in C} \sum_{ngram \in S} Count(ngram)}$$

where $S$ is sentence and $C$ is corpus. Although BLEU does favor models that do not modify the source sentence, it has been shown to correlate well with human judgements of sentence simplification. In order to compensate for no recall measure in BLEU, brevity penalty is used to penalize short sentences. Brevity penalty is mathematically defined as following:

$$BP = \begin{cases} 1 & \text{if} \quad c > r \\ e^{1-r/c} & \text{if} \quad c \leq r \end{cases}$$

where $c$ is the length of the corpus of hypothesis translations and $r$ is the effective reference corpus length.

SARI is a metric used for evaluating automatic text simplification systems. The metric compares the predicted simplified sentences against the reference and the source sentences. It explicitly measures the goodness of words that are added, deleted and kept by the system. An average F-score of each operation is calculated. This metric remains the main metric for evaluating sentence simplification. The mathematical expression of SARI's calculation is shown below:

$$SARI = \frac{F_{add} + F_{keep} + P_{del}}{3}$$

where

$$P_{operation} = \frac{1}{k} \sum_{n=[1,\ldots,k]} p_{operation}(n)$$

$$R_{operation} = \frac{1}{k} \sum_{n=[1,\ldots,k]} r_{operation}(n)$$

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

$$operation \in [del, keep, add]$$

### 3.3 Simple Baseline

A lexicon-based lexical simplification module was adopted as the simple baseline of sentence simplification. Similar to majority class baseline for classification problem, the lexicon-based lexical simplification only modify the sentence on the word level. With a benchmark complex-simple words lexicon in a Python dictionary, complex words were first identified in original sentences and then substituted with corresponding simple words. The lexicon was found via Google and contains approximately 600 commonly used complex-simple word pairs.

| BLEU | Precisions | Brevity Penalty | SARI |
|---|---|---|---|
| 0.59 | [0.74, 0.62, 0.55, 0.50] | 1.00 | 54.85 |

Table 2: Test Results of the Simple Baseline Model

As shown in the Table 2, the simple baseline model yielded surprisingly-positive results, both for BLEU and SARI. Given the fact that the simple baseline model only substitute complex words that were included in the dictionary, the results make no sense. After a few more experiments and careful examinations, there are a few reasons that caused the unreasonably good results of the simple baseline model. First, for BLEU score, the precisions of n-gram are all very high because there are a lot of overlaps between the complex and simple sentences and the output sentences happen to be the original sentences with only few modifications. Second, the brevity penalty is designed to penalize the predictions that are shorter than the target sentences. However, since the length of input sentences were not changed by word substitution, the predictions were almost always longer than the target sentences, yielding a brevity penalty of 1. Third, the SARI score was largely determined by the target sentence, which is also known as reference. Consider the following example:

- source: "About 95 species are currently accepted ."
- prediction: "About 95 species are currently accepted ."
- references: ["About 95 species are currently known .","About 95 species are now accepted .","95 species are now accepted ."]

The resulting SARI score is 55.21, which is much higher than a lot of best-performed models in the research papers. As shown in the example, the source and prediction are exactly the same. The references indeed are simplified versions of the the source. Similarly, by only basic word substitution, the simple baseline model took advantages of the well-covered references and yielded remarkable results, which are not trustworthy.

# 4 Experimental Results

## 4.1 Published Baseline

Based on several papers that claimed state-of-the-art model, a sequence-to-sequence model was selected to establish the strong baseline. Although the encoder-decoder Long Short Term Memory model with attention mechanism achieved great success in traditional sequence-to-sequence, previous studies showed that the encoder-decoder LSTM tends to generate outputs that replicate the input. Therefore, it is difficult to achieve sentence simplification with LSTM.

After researching on the most recent advancement in NLP and transformers, a pre-trained Bidirectional and Auto-Regressive Transformers (BART) base model by Facebook AI was used to establish the strong baseline. BART is a denoising autoencoder built with a sequence-to-sequence model that is applicable to a very wide range of end tasks, such as text summarization and potentially sentence simplification with careful fine-tuning (Lewis et al., 2019). BART, as a member of transformer family, is obviously better than the LSTM-based sequence-to-sequence model. When compared to other transformer models, such as BERT and GPT, BART takes their advantages and complements their disadvantages by using the bidirectional encoder as in BERT and the left-to-right decoder as in GPT. The pre-trained tasks include token masking, token deletion, text infilling, sentence permutation, and document rotation. Then, the BART model was fine-tuned on 4,000 and evaluated on 500 complex-simple English Wikipedia pairs that were randomly sampled from the train and development dataset.

| BLEU | Precisions | Brevity Penalty | SARI |
|---|---|---|---|
| 0.30 | [0.79, 0.66, 0.60, 0.55] | 0.47 | 17.32 |
| 0.31 | [0.79, 0.67, 0.60, 0.55] | 0.47 | 17.46 |
| 0.30 | [0.79, 0.67, 0.60, 0.55] | 0.47 | 17.41 |

Table 3: Evaluation Results of the Strong Baseline Model

As shown in the Table 3, each line represents the evaluation result of an epoch. The results from each epoch were not very different. In addition, the brevity penalty is 0.47, indicating that the length of predictions are shorter than that of references. As shown in the Table 4, the model yielded similar results on the test set.

| BLEU | Precisions | Brevity Penalty | SARI |
|---|---|---|---|
| 0.28 | [0.76, 0.64, 0.58, 0.51] | 0.46 | 17.01 |

Table 4: Test Results of the Strong Baseline Model

## 4.2 Extension 1: Lexical Simplification

Lexical simplification usually involves 3 steps: identify complex words in a given sentence, generate candidates, and select the best candidates. There are many categories of strategies for each step. For complex word identification (CWI), the advanced approaches appear with threshold-based strategy, aiming at searching for a threshold over a given simplicity metric. Then, there is lexicon-based strategy, consisting of using a lexicon of complex words to identify simplifiable candidates. Modern strategies usually used machine learning-assisted methods. For candidates generation, there are linguistic database querying and automatic candidates generation, the latter usually requiring assistance from machine learning.

Although lexical simplification is a different task from sentence simplification, we believe that they orient towards the same goal and performing lexical simplification before sentence simplification will yield a better result for the final output. Sentence simplification could also bear an underlying goal of simplifying words. Thus, we decided to perform lexical simplification as an extension.

For identifying complex words, we first create a model that can detect or identify possible complex word, which is called CWI model. We adopted a Bi-LSTM structure for this part and trained on University of Hamburg's complex word identification dataset. After the CWI model, we have used a pre-trained BERT to generate candidates over the words which the CWI model identified as complex so that the candidates are based not only in the synonyms but also in the context. Here, we take advantage of BERT's most powerful feature to predict the missing word. We concatenated the original sentence with the masked sentence by masking the complex words identified so that BERT can generate most probable words to place in that blank. Then, for each candidate generated, we computed their Zipf value and select the word with the smallest value as substitution.

| BLEU | Precisions | Brevity Penalty | SARI |
|---|---|---|---|
| 0.55 | [0.78, 0.63, 0.55, 0.48] | 0.92 | 51.32 |
| 0.59 | [0.76, 0.62, 0.54, 0.48] | 1.00 | 52.18 |
| 0.59 | [0.76, 0.62, 0.54, 0.48] | 1.00 | 51.79 |

Table 5: Evaluation Results of the Extension 1 Model

| BLEU | Precisions | Brevity Penalty | SARI |
|---|---|---|---|
| 0.54 | [0.69, 0.57, 0.49, 0.44] | 1.00 | 50.07 |

Table 6: Test Results of the Extension 1 Model

The Table 3 shows the results of the strong baseline and the Table 5 shows the results of the Extension 1 model that includes both lexical simplification with BERT and sentence simplification with BART. When comparing the results from both tables, it is obvious that the BLEU scores of the all three training epochs in Table 5 are higher than that in Table 3. The main reason is that the brevity penalty of the Extension 1 model is much higher than that of the baseline model, meaning that the simplified sentences have similar or longer length as the reference (target) sentences. The high brevity penalty indicates that the Extension 1 model is more robust than the strong baseline model since the Extension 1 model could simplify sentences without losing too much information. However, the n-gram precisions of the Extension 1 model slightly decreased as comparing to those of the baseline model. First, the simple words generated might not be chosen by the reference sentence. Second, it is possible that the simple words generated by lexical simplification model require different sentence structure than the original words. Therefore, the n-gram precisions of the Extension 1 model are lower than the strong baseline model. In addition, the amount of decrease in precision increases as the number of grams increases. Finally, the SARI score of the Extension 1 model greatly improves when compared to that of the strong baseline model, which might be due to the same reason that the predicted sentences are not shorter than the target sentences.

As shown in the Table 6, the test results of the Extension 1 model are slightly worse than the evaluation results while much better than the test results of the strong baseline model.

### 4.3 Extension 2: Control Token

For our extension 2, we decided to build a control token to generate the best output. Previous studies have shown that by having more control over output sentence, we can improve the results of simplified sentences. In previous study, Controllable Sentence Simplification uses a combination of length, lexical complexity, and paraphrasing to create a golden model. For our model, we wanted to build upon extension 1 because with extension 1's complex word substitution, we already have good control over lexical complexity, we expect to see better performance once we add length and paraphrasing control. In order to do so, we calculate the token measurement between original sentence and simplified sentence. For length token we measure the character length ratio; for paraphrasing we use Levenshtein Similarity of the two sentences. In order to train our model to better recognize the token meaning, we would want more data for each unique token. One way to do that without increasing the training data size and train time is by rounding our token number to the nearest 0.05. After doing so, we have a good sample size for each token. For our test set, we experimented with several token levels. If we keep length token as 1, BLEU score will be highest due to the sentence having the same length, but SARI doesn't perform that well. For Levenshtein Similarity we found 0.85 to be best performing number. Our results for test set shows a good improvement from 0.28 to 0.58 BLEU score, compared to the output statistics of the strong baseline. In addition, the second extension has outperformed on both metrics (BLEU and SARI scores) compared to extension 1.

| BLEU | Precisions | Brevity Penalty | SARI |
|---|---|---|---|
| 0.59 | [0.74, 0.62, 0.55, 0.49] | 1.00 | 52.49 |
| 0.60 | [0.76, 0.63, 0.56, 0.50] | 1.00 | 53.05 |
| 0.59 | [0.75, 0.62, 0.55, 0.49] | 1.00 | 52.87 |

Table 7: Evaluation Results of the Extension 2 Model

| BLEU | Precisions | Brevity Penalty | SARI |
|---|---|---|---|
| 0.58 | [0.75, 0.62, 0.53, 0.47] | 1.00 | 51.56 |

Table 8: Test Results of the Extension 2 Model

### 4.4 Error Analysis

Original: green cove springs is the county seat of government for clay county, florida.

Reference: green cove springs is a city of florida in the united states. it is the county seat of clay county.

Prediction: the new, eight-sided structure was opened in 1956.

This sentence is one of the rare simplification where the simplified reference is composed of two separate simple sentences. Our model prediction does reduce the sentence length by about 0.9 by removing of "government" since "government" contains the most letters. Our model does not understand that ", florida" means a city of Florida and cannot decompose the complex meaning into two sentences. However, I do not consider this a big problem since our prediction sentence does make sense and convey the same meaning.

Original: the new, eight-sided structure was opened in 1956.

Reference: the new tower opened in 1956.

Prediction: the new, eight-sided structure was opened in 1956.

Here we have a much more simple sentence, the model does not recognize any complex words to replace and it fails to identify an eight-sided structure as a tower. It did not make any changes to the original sentence. One reason could be in our training, we have a lot of dataset where the sentence from simple Wikipedia is the same as original Wikipedia. In the future what we could do is clean out more of these data to avoid the model making such predictions.

## 5 Conclusions

In this project, we successfully built two extensions (control token and lexical simplification) to achieve reduction of text complexity. Comparing the performance of these two extensions to that of many state-of-the-art text simplification tools, we notice that although some concurrent works have yielded higher BLEU scores (>0.7) than ours (∼0.6), we are able to achieve a higher SARI score (∼50) than their models (∼40).
The gap between this peer comparison of the two metrics can be attributed to the fact that our model heavily focuses on word substitution and does well on that end. On the other hand, it's worth mentioning that as we've discussed before, some text simplification tools were able to achieve a high SARI score, but with a quick sanity check, we can discover that those tools are simply returning the exactly-same sentence as the "simplified" version. This discovery proves that the metric SARI itself might contain some intrinsic flaws and thus improvements remain viable to our current model.

## 6 Acknowledgement

We would like to thank Yifei Li, our TA, for providing a lot of helpful suggestions, especially the one that guided us to use BART instead of LSTM as the strong baseline model.

## 7 File Location

The project-related files could be found here: https://drive.google.com/drive/folders/1W5CTY-mEG2eFRgxChr0o0R9lguBH_7do?usp=sharing

## References

[1] Belder, J.D., & Moens, M. (2010). Text simplification for children. *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*

[2] Carroll, J., Minnen, G., Canning, Y., Devlin, S., & Tait J. (1998). Practical simplification of English newspaper text to assist aphasic readers. *Proceedings of AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology.*

[3] Devlin, S. & Tait J. (1998). The use of a psycholinguistic database in the simplification of text for aphasic readers. *Linguistic Databases.*

[4] Evans, R.J., Orasan, C., & Dornescu, I. (2014). An evaluation of syntactic simplification rules for people with autism. *PITR@EACL.*

[5] Heafield, K. (2011). KenLM: Faster and Smaller Language Model Queries. *WMT@EMNLP.*

[6] Kauchak, D. (2013). Improving Text Simplification Language Modeling Using Unsimplified Text Data. *Annual Meeting of the Association for Computational Linguistics.*

[7] Kriz, R., Sedoc, J., Apidianaki, M., Zheng, C., Kumar, G.S., Miltsakaki, E., & Callison-Burch, C. (2019). Complexity-Weighted Loss and Diverse Reranking for Sentence Simplification. *North American Chapter of the Association for Computational Linguistics.*

[8] Martin, L., Sagot, B., Villemonte de la Clergerie, E., & Bordes, A. (2019). Controllable Sentence Simplification. *ArXiv, abs/1910.02677.*

[9] Paetzold, Gustavo Henrique. (2016). Lexical Simplification for Non-Native English Speakers. Ph.D. thesis, University of Sheffield, Sheffield, UK.

[10] Parker, R., Graff, D., Kong, J., Chen, K., & Maeda, K. (2011). English Gigaword Fifth Edition LDC2011T07. *Philadelphia: Linguistic Data Consortium.*

[11] Rello, L., Bayarri, C., Gorriz, A., Baeza-Yates, R., Gupta, S., Kanvinde, G., Saggion, H., Bott, S., Carlini, R., & Topac, V. (2013). DysWebxia 2.0!: more accessible text for people with dyslexia. *International Cross-Disciplinary Conference on Web Accessibility.*

[12] Watanabe, W. M., Junior, A. C., Uzêda, V. R., Fortes, R. P., Pardo, T. A., & Aluísio, S. M. (2009). Facilita: Reading assistance for low-literacy readers. *Proceedings of the 27th ACM International Conference on Design of Communication - SIGDOC '09.*

[13] Xu, W., Callison-Burch, C., & Napoles, C. (2015). Problems in Current Text Simplification Research: New Data Can Help. *Transactions of the Association for Computational Linguistics, 3*, 283-297.

[14] Zhang, X., & Lapata, M. (2017). Sentence Simplification with Deep Reinforcement Learning. *ArXiv, abs/1703.10931.*