Document-level Text Simplification : A Two Stage Plan-Guided Approach

Atharv Ramesh Nair
EE20BTECH11006

Prashanth Sriram S
CS20BTECH11039

Rahul V
AI20BTECH11030

***Abstract*—Document-level text simplification involves rewriting documents containing a group of sentences into fewer or more sentences such that it retains the original meaning of the document and is easier to understand. We propose a Plan-Guided SIMSUM approach where we generate operations (copy, rephrase, split, delete) using the planning model and prepend them to individual sentences which are given as input to the SIMSUM model. We utilized Wiki-auto, D-Wikipedia, and PLABA document-level text simplification datasets to evaluate our model against individual SIMSUM and Plan-Guided simplification models. Our proposed model achieved SARI(43.56), D-SARI(38.52) scores on the Wiki-auto showing significant improvements over the individual models.**

I. Introduction

Text simplification involves simplifying a document such that it's understandable by people from different reading levels while still retaining the overall content of the document. In the case of sentence level text simplification, each sentence is simplified one by one, hence the total number of sentences in the document remains the same. Whereas in the case of document level text simplification, documents containing a group of sentences are rewritten into fewer or more sentences. Common operations performed in document-level text simplification include sentence splitting, joining, deletion, reordering, addition, and rephrasing.

The document-level text simplification task can be defined as follows - Given a complex article $C = \{S_1, S_2, ..., S_n\}$ consisting of n sentences as input, we want to output a simple article $F = \{T_1, T_2, ..., T_n\}$ consisting of m sentences. Where - F retains the original meaning of C. F is more easy to understand than C. m need not be equal to n, F can have more or equal or lesser number of sentences than C.

Blinova et al. [1] proposed a two-stage approach for document-level text simplification - summarization followed by simplification. They use two different transformer models (BART [2] or T5 [3]). Generation is guided using Keyword prompting, and embedding similarity is used as part of the loss function in an attempt to enhance performance.

Cripwell et al. [4] proposed a plan-guided approach for document-level text simplification which involves predicting the operation (copy, rephrase, split, delete) that needs to be performed on each sentence. The label prediction is assisted by using the context of surrounding sentences in the document.

Our proposed model combines both the above approaches. Instead of keyword prompting [1], we generate operations (copy, rephrase, split, delete) using the planning model [4] and prepend them to individual sentences which are given as input to the SIMSUM model [1]. Our proposed model achieves a D-SARI score of 38.52, which outperforms SIMSUM model (33.22), the Plan-Guided model (24.27) and the baseline BART model (24.32) on our custom downsized version of Wiki-auto Dataset. The rest of the report is organized as follows: In Section II, we introduce various datasets used in literature and our custom dataset. In Section III and Section IV, we briefly go over the SIMSUM and Plan-Guided models. In Section V, we explain our proposed approach. In Section VI, we go over the implementation details. In Section VII and Section VIII, we discuss the metrics used and results obtained on different datasets respectively.

II. Datasets

Most of the commonly used text simplification datasets - WikiLarge, Turkcorpus, Newsela are made specifically for the sentence-level simplification task and connot be used for our task.

Blinova et al. [1] have used two datasets - D-Wikipedia and Wiki-Doc for their SIMSUM model. D-Wikipedia dataset [5] and Wiki-Doc dataset [6] have been built using the articles from the English Wikipedia and its corresponding Simple English Wikipedia article. These datasets were filtered and re-aligned before training on the SIMSUM model.

Cripwell et al. [4] have used two datasets - Wiki-auto and Newsela-auto for their Plan-guided simplification model. Wiki-auto dataset [4] and Newsela-auto dataset [7] were prepared from the WikiLarge and Newsela datasets respectively by getting alignments of the documents with its simplification at both sentence and paragraph levels.

In Wiki-auto, complex sentences were annotated by heuristically assigning a simplification operation label (copy, rephrase, split, delete) using pairs of complex and simplified sentences (c_i, s_j).

1) Delete: c_i is aligned to single s_j
2) Copy: c_i is aligned to single s_j with Levenshtein similarity above 0.92
3) Rephrase: c_i is aligned to single s_j with Levenshtein similarity below 0.92
4) Split: c_i is aligned to more than one s_j

In Wiki-auto, many complex documents were much larger than simplified documents. Hence, the complex documents were clipped after the last aligned paragraph. Many simplified documents were much smaller in size compared to their complex documents. Hence, those documents with more than 50% of labels as deleted were removed. Articled with more than 1024 tokens were removed to fit into the BART model.

We couldn't use the D-Wikipedia and Wiki-Doc datasets for training as there are no aligned sentence pairs for each document, so we decided to use Wiki-auto for training of all models for a fair comparison. Further, keeping in mind the computational constraints, we utilized a reduced version of the Wiki-auto (R-Wiki-auto) for training. It was prepared by randomly sampling from the Wiki-auto dataset. R-Wiki-auto contains 12k documents for training, 1k documents for validation, 1k documents for testing.

We used PLABA (Plain Language Adaptation of Biomedical Abstracts) [8] for testing our model out of domain. It contains document-level text adaptation of answers (from PubMed search results) to common consumer health questions (from MedlinePlus queries).

III. SIMSUM

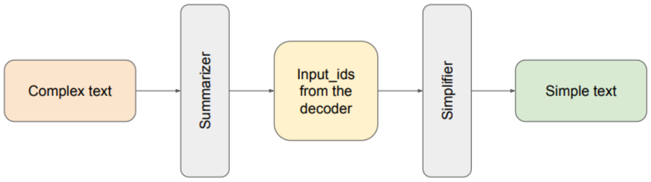

Fig. 1. SIMSUM Framework

SIMSUM [1] is a two-stage process 1 for document-level text simplification consisting of a summarizer transformer followed by a simplification transformer, trained in an end-to-end fashion. This model structure is helpful for this task because in text simplification we have to retain the original meaning of the document (summarizer takes care of this) and also make it easy to understand (simplifier takes care of this). The authors explored using both Pretrained BART [2] and T5 [3] models as backbones for each stage.

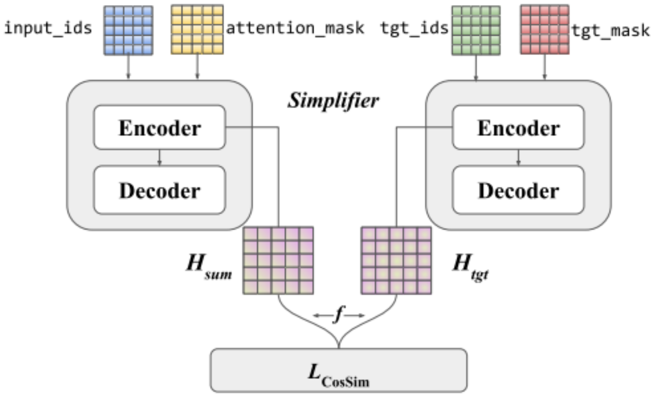

Fig. 2. Embedding similarity computation process

Model	D-Wikipedia	Wiki-Doc
Without prompting (Vanilla)	39.54	40.32
3 kw_score div=0.5	39.69	41.68
3 kw_score div=0.7	39.65	**41.96**
3 kw_sep div=0.7	38.85	39.58
4 kw_score div=0.7	39.97	41.85
3 kw_score div=0.9	39.65	41.94
4 kw_score div=0.9	**40.07**	41.89

TABLE I
D-SARI SCORES ON D-WIKIPEDIA AND WIKI-DOC BY SIMSUM (T5-BACKBONE) WITH KEYWORD PROMPTS. DIV DENOTES THE PARAMETER OF THE DIVERSITY OF THE EXTRACTED KEYWORDS IN KEYBERT. 3 KW_SCORE MEANS 3 KEYWORDS IN KW_SCORE STRATEGY

Model	D-Wikipedia	Wiki-Doc
$\lambda = 0$ (Vanilla)	39.54	**40.32**
$\lambda = 0.001$	38.51	40.03
$\lambda = 0.01$	38.27	40.15
$\lambda = 0.1$	39.39	39.90
$\lambda = 0.5$	**39.85**	36.25
$\lambda = 1.0$	38.38	31.86

TABLE II
RESULTS ON D-WIKIPEDIA AND WIKI-DOC BY SIMSUM (T5-BACKBONE) WITH EMBEDDING SIMILARITY LOSS. λ DENOTES THE HYPER-PARAMETER THAT CONTROLS THE CONTRIBUTION OF THE ADDITIONAL LOSS TERM

KeyBERT [9] was used to fetch main keywords from the complex document to force the model to focus on those keywords. Two different approaches were used for keyword prompting. The first one is kw_score where keywords along with their similarity score was prepended to the input text. The second one is kw_sep where keywords and EOS (End of sentence) tokens were prepended to the input text.

[1] proposed a new loss function consisting of L_1 (the original cross-entropy loss) and L_{CosSim} (the embedding similarity loss 2). This additional embedding loss term forces the model to generate outputs more similar to the reference texts. Hyper parameter λ is used to control the contribution of this loss term. Embedding similarity is the cosine distance between the output embeddings and the reference embeddings during training. Reference embeddings are obtained by feeding the reference text to the simplifier as input and taking the embedding of the last hidden state of the encoder. Output embeddings are obtained by taking the summarizer's encoding representation and then transforming it to the simplification space.

Limitations

The two-stage approach significantly helps in performance. The Summarizer helps significantly reduce the size of the output which results in lower D-SARI Scores and also better scores in human evaluation. From the ablation study presented in [1], we can infer that the two additional factors, i.e., KeyBert and Embedding Similarity loss doesn't lead to a significant improvement in automatic evaluation results across Datasets. Using keyword extraction, the best improvement was only 1.5% and 0.5% in D-SARI score on D-Wikipedia and Wiki-Doc respectively. Similarly, even using the additional embedding similarity loss didn't produce significant improvements. Moreover, these improvements are heavily dependent on the hyperparameters, for which the optimal value varies with the dataset.

IV. Plan-Guided Simplification

In Plan-Guided simplification [4], for each sentence in the input text, an operation (copy, rephrase, split, delete) is predicted using a classifier by taking the current sentence as the input and its surrounding sentences as the context. The predicted operation is appended to the complex sentence and is fed as input to BART which performs the simplification.

Given an input document $C = \{c_1, c_2, ..., c_n\}$ consisting of n complex sentences c_i as input, the planner predicts a sequence of n simplification operations $\hat{P} = \{\hat{o}_1, \hat{o}_2, ..., \hat{o}_n\}$ where $\hat{o}_i \in \{copy, rephrase, split, delete\}$. Different operations have conflicting requirements - splitting is mostly context independent whereas deletion and to some extent copy and rephrase are all context dependent.

The authors use Dynamic context representation 4 which is obtained by taking token level embeddings of the current sentence, sentence level embeddings of the preceding simplified sentences (During training, ground truth simplifications are used and during inference simplifications generated during the previous time step are used), succeeding complex sentences using pretrained Sentence-BERT model [10].

The planning model 3 consists of a classifier with cross-attention over the context and two types of positional embeddings. The classifier is built upon the ROBERTa classifier architecture by inserting a cross-attention between the self-attention block and the feed-forward layer of every transformer block, which enables the model to condition upon the context sentences in the document.

Document positional embedding indices are the document quintile (1-5) in which the sentence is present. Context positional embedding is the relative distance of the given sentence from the input sentence. These positional embeddings are generated by embedding layer and is useful in encoding information about the document and the relative positions of the context.

V. Plan-Guided SIMSUM

The control tokens used in the Plan-Guided simplification could be more useful when compared to the prepended keywords used in SIMSUM [1] which didn't show any consistent improvement. We propose a fusion of the two models. We first predict the operations to be performed on each sentence of the document by using the approach discussed in Section IV. We then prepend the corresponding operation token before each sentence in the document and feed the entire document to the two-stage SIMSUM model [1]. An example of such a document is shown in Fig 5

VI. Implementation

We have used Pytorch Lightning [11] and used code from [1] and [4] for this project. We trained three models (BART, SIMSUM, PG-SIMSUM) on R-Wiki-auto dataset and used the available Plan-Guided model pretrained weights (trained on the entire Wiki-auto) for evaluating it. We closely followed the procedure used in [1] for training. All models were trained for 10 epochs using the AdamW Optimizer [12] with a learning rate of 5×10^{-5} and using a cosine scheduler (warmup epochs = 5) with a batch size of 16.

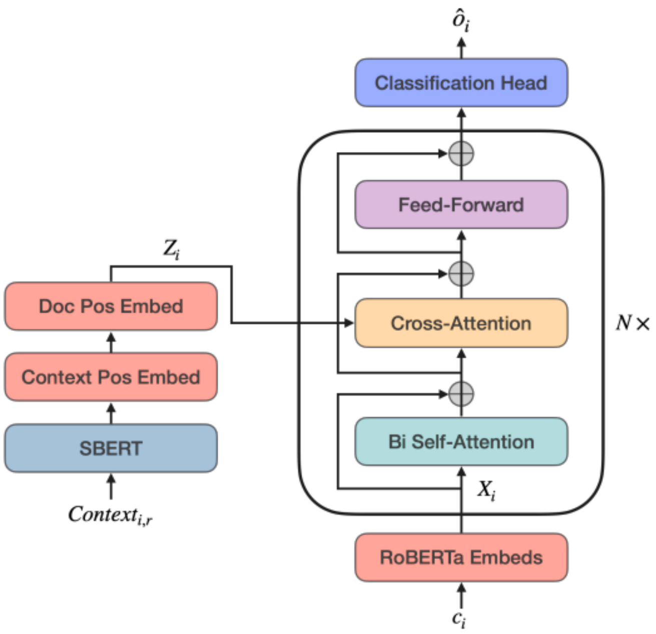

Fig. 3. Contextual classifier model architecture

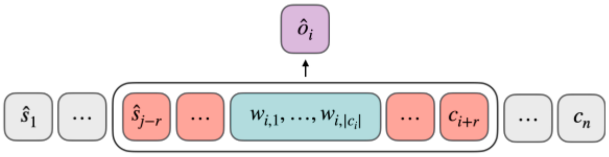

Fig. 4. Dynamic contextual classifier

For the Plan-Guided model, we first predicted the control tokens (operations) for the entire dataset using the pretrained planner model and saved the sentences prepended with control tokens. This was fed as input to the SIMSUM model (entire document at a time). The Vanilla SIMSUM and Plan-Guided SIMSUM are identical except for the control tokens. For evaluation, we tested our models on the custom R-Wiki-auto

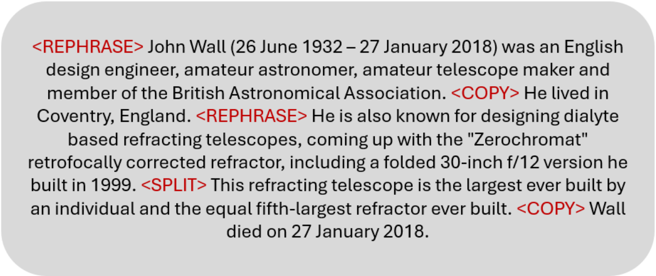
<REPHRASE> John Wall (26 June 1932 – 27 January 2018) was an English design engineer, amateur astronomer, amateur telescope maker and member of the British Astronomical Association. <COPY> He lived in Coventry, England. <REPHRASE> He is also known for designing dialyte based refracting telescopes, coming up with the "Zerochromat" retrofocally corrected refractor, including a folded 30-inch f/12 version he built in 1999. <SPLIT> This refracting telescope is the largest ever built by an individual and the equal fifth-largest refractor ever built. <COPY> Wall died on 27 January 2018.

Fig. 5. Example Sentence

test set, D-Wikipedia and PLABA.

We only used Cross Entropy Loss for training We didn't use the Embedding Similarity Loss from SIMSUM as the performance improvement was not significant.

VII. Evaluation metrics

We have utilized SARI, D-SARI text simplification metrics for evaluating the performance of our proposed model. We have not used FKGL and BLEU metrics as they are not suitable for evaluating text simplification.

1) SARI: SARI [13] measures goodness of words that are added, kept, deleted in the output sentence compared to the input and the reference sentences. It is the most popular metric used in text simplification tasks.
2) D-SARI: D-SARI [5] is a modified version of the SARI to be more applicable for the document-level simplification task. It adds additional penalties when the number of output sentences differs greatly from the reference sentences, for duplicated sentences, etc.

VIII. Results

BART, SIMSUM, Plan-Guided, PG-SIMSUM models were tested on the R-Wiki-auto dataset, and the results are tabulated in Table III. We can clearly see that PG-SIMSUM model outperformed all the other models in both SARI and D-SARI metrics.

Model\Metric	SARI	D-SARI
BART	38.84	24.32
SIMSUM	35.07	32.47
Plan-Guided	25.57	24.27
PG-SIMSUM	**43.56**	**38.52**

TABLE III
Results on R-Wiki-auto

Results on PLABA dataset are tabulated in Table IV. BART model achieved the best SARI score and Plan-guided model achieved the best D-SARI score.

Model\Metric	SARI	D-SARI
BART	**35.09**	28.19
SIMSUM	31.54	25.14
Plan-Guided	30.42	**28.35**
PG-SIMSUM	32.52	25.95

TABLE IV
Results on PLABA

Results on D-Wikipedia dataset are tabulated in Table V. BART model achieved the best SARI score and SIMSUM model achieved the best D-SARI score.

Model\Metric	SARI	D-SARI
BART	**44.81**	28.35
SIMSUM	43.69	**31.73**
PG-SIMSUM	41.66	31.51

TABLE V
Results on D-Wikipedia

Poorer results on datasets other than the R-Wiki-Auto dataset have been explained in detail in Section X.

IX. Sample Outputs

Example from R-Wiki-auto

Complex: Giovanni Berlinguer Cavaliere di Gran Croce OMRI (] ; 9 July 1924 â€“ 6 April 2015) was an Italian politician, humanist and professor of social medicine. He was born in Sassari, Sardinia, the son of Mario Berlinguer. A physician and professor of public health, he worked first in social medicine at the University of Sassari (1969â€“1974) and then in occupational health at the University ”La Sapienza” Rome (1975â€“1999). Like his brother Enrico, Giovanni Berlinguer was a major figure in the Italian Communist Party (PCI) and was elected to the Chamber of Deputies in 1972, 1976 and 1979 and to the Senate in 1983 and 1987. He ran for secretary of the Democrats of the Left (DS) in 2001, and was defeated by Piero Fassino 61.8% to 34.1%. From 2004 to 2009 he was a Member of the European Parliament representing the DS and sits with the Party of European Socialists group. At the convening of the Parliament on 20 July 2004 he was found to be the oldest member, and as such presided over the Parliament during the election of Josep Borrell Fontelles in 2004 and Hans-Gert PoÌˆttering in 2007 as President of the Parliament. He died in Rome at the age of 90.

Simple: Giovanni Berlinguer (]), Cavaliere di Gran Croce OMRI, (9 July 1924 â€“ 6 April 2015) was an Italian politician and Professor of Social Medicine. He was born in Sassari, Sardinia, the son of Mario Berlinguer. He ran for the secretarship of the Democrats of the Left (DS) in 2001, being defeated by Piero Fassino 61.8% to 34.1%. From 2004 to 2009 he was a Member of the European Parliament representing the DS and sits with the Party of European Socialists group. He died in Rome at the age of 90.

BART: Giovanni Berlinguer (]; 9 July 1924 â€“ 6 April 2015) was an Italian politician, humanist and professor of social medicine. He was born in Sassari, Sardinia.He was elected to the Chamber of Deputies in 1972, 1976 and 1979 and to the Senate in 1983 and 1987, and served as a Member of the European Parliament from 2004 to 2009 representing the Democrats of the Left (DS). Berringuer died in Rome at the age of 90.

SIMSUM: Giovanni Berlinguer (25 January 1930 â€“ 24 January 2019) was an Italian politician. He was a major figure in the Italian Communist Party. Â He was elected to the Chamber of Deputies in 1972, 1976 and 1979, representing the San Lorenzo region in the state of Lazio from 1972 to 1979, and to the Senate from 1979 to 1981 and again from 1981 to 1982. Ã— He was re-elected in 1982 to the second term, but was defeated by Piero Fassino 61.8% to 34.1% in the 2001 general election.

Plan-Guided: Berlinguer Cavaliere di Gran Crocerebri (9 July 1924 â€“ 6 April 2015) was an Italian politician and doctor. He was born in Sardinia, Italy, the son of Mario Berlinguer, the country's leader. malaria was a doctor and professor of public health. Like his brother Enrico, Giovanni Berlinguer was a major figure in the Italian Communist Party (PCI).From 2004 to 2009 he was a member of the European Parliament. The member of Parliament at the time was found to be the oldest member. He served as the president of the Parliament during the 2004 election of Josep Borrell Fontelles, a member of the government, and Hans-Gert Po Tillesttering in 2007 as president. He died in Rome, Italy, on Tuesday, at the age of 90.

PG-SIMSUM: Giovanni Berlinguer Cavaliere di Gran Croce OMRI (9 July 1924 – 6 April 2015) was an Italian politician, humanist and professor of social medicine. He was a major figure in the Italian Communist Party (PCI) and was elected to the Chamber of Deputies in 1972, 1976 and 1979.

We can observe that the output size in PG-SIMSUM and SIMSUM is least when compared to the other models

X. Conclusion and Limitations

We observe that Plan-Guided SIMSUM model is able to perform much better than the previous models on R-Wiki-auto dataset. We tested the models on out-of-domain data (PLABA). We can observe that the baseline BART is able to perform the best in this case. It probably means that BART model has a better summarization capacity when compared to SIMSUM or Plan-Guided models.

One big limitation of the Plan-Guided approach is the requirement of sentence-aligned datasets for training. Even while testing, not having correct sentence alignments can be an issue. We tried splitting datasets like Wiki-Doc and Wiki-auto into sentences using "." (full stops). But "." is a commonly occurring character in many of the simplification datasets. Some of its usages are valid (like U.S.A), but most of these occur due to the noisy nature of the data, primarily because they are scraped from Wiki-simple. So, we can't split documents to sentences by relying on "." characters since this may lead to mismatch between simple and complex pairs. Even if there is no mismatch i.e., both the complex and simple sentences have the same erroneous "." character, the pair will probably be irrelevant. We observed a significant amount of such pairs while looking at the D-Wikipedia and Wiki-Doc datasets. This can really hurt the training process and even affect the SARI/D-SARI scores.

Training on larger sentence-aligned datasets like the full Wiki-auto or Newsela [7] can be helpful. We also need to come up with ways of effectively splitting document-level datasets to corresponding sentences so the model can be more effective on datasets like D-Wikipedia and Wiki-Doc.

References

[1] S. Blinova, X. Zhou, M. Jaggi, C. Eickhoff, and S. A. Bahrainian, "Simsum: Document-level text simplification via simultaneous summarization," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9927–9944, 2023.

[2] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

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

[4] L. Cripwell, J. Legrand, and C. Gardent, "Document-level planning for text simplification," in *17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 993–1006, Association for Computational Linguistics, 2023.

[5] R. Sun, H. Jin, and X. Wan, "Document-level text simplification: Dataset, criteria and baseline," *arXiv preprint arXiv:2110.05071*, 2021.

[6] D. Kauchak, "Improving text simplification language modeling using unsimplified text data," in *Proceedings of the 51st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 1537–1546, 2013.

[7] C. Jiang, M. Maddela, W. Lan, Y. Zhong, and W. Xu, "Neural crf model for sentence alignment in text simplification," *arXiv preprint arXiv:2005.02324*, 2020.

[8] K. Attal, B. Ondov, and D. Demner-Fushman, "A dataset for plain language adaptation of biomedical abstracts," *Scientific Data*, vol. 10, no. 1, p. 8, 2023.

[9] M. Grootendorst, "Keybert: Minimal keyword extraction with bert," 2020.

[10] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[11] W. Falcon and The PyTorch Lightning team, "PyTorch Lightning," Mar. 2019.

[12] I. Loshchilov and F. Hutter, "Fixing weight decay regularization in adam," *CoRR*, vol. abs/1711.05101, 2017.

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