Beyond Sentence-level Text Simplification
Reproducibility Study of Context-Aware Document Simplification

Jan Bakker, Jaap Kamps
University of Amsterdam
Amsterdam, The Netherlands
jan.bakker@student.uva.nl, kamps@uva.nl

Abstract

Previous research on automatic text simplification has focused on almost exclusively on sentence-level inputs. However, the simplification of full documents cannot be tackled by naively simplifying each sentence in isolation, as this approach fails to preserve the discourse structure of the document. Recent Context-Aware Document Simplification approaches explore various models whose input goes beyond the sentence-level. These model achieve state-of-the-art performance on the Newsela-auto dataset, which requires a difficult to obtain license to use. We replicate these experiments on an open-source dataset, namely Wiki-auto, and share all training details to make future reproductions easy. Our results validate the claim that models guided by a document-level plan outperform their standard counterparts. However, they do not support the claim that simplification models perform better when they have access to a local document context. We also find that planning models do not generalize well to out-of-domain settings.

Lay Summary: *We have access to unprecedented amounts of information, yet the most authoritative sources may exceed a user's language proficiency level. Text simplification technology can change the writing style while preserving the main content. Recent paragraph-level and document-level text simplification approaches outcompete traditional sentence-level approaches, and increase the understandability of complex texts.*

Keywords: Generative Text Simplification, Machine Learning for Natural Language Processing, Reproducibility Study.

1. Introduction

To date, most research on automatic text simplification has focused on sentence-level inputs. However, the simplification of full documents cannot be tackled by naively simplifying each sentence in isolation, as this approach fails to preserve the discourse structure of the document. Cripwell et al. (2023b) proposed to guide the simplification of each sentence by a document-level plan specifying how it should be simplified—should it be copied, deleted, split or rewritten? Their planning model leverages both the context of each sentence and its internal structure in order to predict a simplification operation. Although this approach was able to outperform the baseline end-to-end systems, it is still limited in that the simplification model has no direct access to the context of each sentence.

In their follow-up paper, Cripwell et al. (2023a) explored various systems that use a local document context within the simplification process itself, either by working at the paragraph level or attending over an additional input representation. In doing so, they achieved state-of-the-art performance on the Newsela-auto dataset, even when not relying on plan-guidance. Figure 1 shows a Wiki-auto example input and the output of one of the sentence-level and paragraph-level text simplification models.

In this paper, we aim to replicate their experiments on another dataset, namely Wiki-auto, in order to assess the generalizability of their findings. Furthermore, we investigate the ability of the models trained on Newsela-auto to adapt to new domains by evaluating them on Wiki-auto. The rest of this paper is structured in the following way. Section 2 discusses the exact scope of our reproducibilty study. Section 3 details the experimental data, models, and setup. Section 4 presents the planning and simplication results on Wiki-auto, both under within-domain and out-of-domain conditions. We end the paper with discussion and conclusions in Section 5. An appendix provides additional evaluation measures and further examples of output of the various models.

2. Scope of Reproducibility

This section discusses the exact scope of our reproducibilty study.

We identify two main claims made by Cripwell et al. (2023a) about document-level simplification which we aim to verify:

1. Considering all metrics, text-only models that take as input either a sentence ($\text{BART}_{\text{sent}}$) or a whole document ($\text{BART}_{\text{doc}}$, LED_{doc}) underperform compared to models that have access to a local document context ($\text{BART}_{\text{para}}$, LED_{para}, ConBART).

2. Plan-guided models outperform their standard counterpart on all metrics.

Proceedings of the Workshop on DeTermIt! Evaluating Text Difficulty in a Multilingual Context @ LREC-COLING 2024, pages 27–38

Complex document

Silvano "Nano" Campeggi (] ; January 23, 1923 – August 29, 2018) was an Italian artist who designed and produced the artwork for the posters of many classic Hollywood films. His iconic images are associated with the golden era of Hollywood and Campeggi is now generally regarded as the most important graphic artist and poster designer in the history of American cinema.

In the following decades, Campeggi designed and produced the poster and advertising graphics for over 3000 films, working not only under contract with the MGM studios, but also with Warner Brothers, Paramount, Universal, Columbia Pictures, United Artists, RKO, Twentieth-Century Fox and several other movie studios. Sixty- four of the films he illustrated won Oscars, including "Casablanca", "Ben-Hur", "Singin' in the Rain", "An American in Paris", "West Side Story", "Exodus", "Breakfast at Tiffany's", and "Gigi".

Campeggi died on 29 August 2018, at the age of 95.

Simplification plan-guided sentence-level BART model

Silvano "Nano" Campeggi (]; January 23, 1923 – August 29, 2018) was an Italian artist. He designed and produced the artwork for the posters of many classic Hollywood movies. His iconic images are associated with the golden era of Hollywood and Campeggi is generally regarded as the most important graphic artist and poster designer in the history of American cinema.

Sixty-four of the films he illustrated won Oscars, including "Casablanca", "Ben-Hur", "Singin' in the Rain", "An American in Paris", "West Side Story", "Exodus", "Breakfast at Tiffany's", and "Gigi".

Campeggi died on 29 August 2018 in Milan at the age of 95.

Simplification plan-guided paragraph-level BART model

Silvano "Nano" Campeggi (]; January 23, 1923 – August 29, 2018) was an Italian artist. He designed and produced the artwork for the posters of many classic Hollywood movies. His iconic images are associated with the golden era of Hollywood.

Campeggi illustrated over 3000 movies, including "Casablanca", "Ben-Hur", "Singin' in the Rain", "An American in Paris", "West Side Story", "Exodus", "Breakfast at Tiffany's", and "Gigi".

Campeggi died on 29 August 2018, at the age of 95.

Figure 1: Wiki-auto example of plan-guided text simplification at the sentence- and paragraph-level.

These claims are made in the Results and Discussion section of the original paper, under the subsections *Context Awareness Matters* and *The Utility of Planning*.

While the authors of the original paper only performed their simplification experiments on Newsela-auto, we replicate their experiments on Wiki-auto.[1] In a sense, our paper adds a missing table to Cripwell et al. (2023a), as the earlier Cripwell et al. (2023b) evaluated their planning models on both datasets. They found the accuracy on Newsela-auto to be significantly higher, which they attributed to Wiki-auto being an inferior simplification corpus. Indeed, the pairs of complex-simple documents in Wiki-auto were automatically collated and aligned, while the Newsela dataset contains news articles that were manually rewritten at different levels of simplification (Xu et al., 2015). However, the Newsela dataset requires a license to use, making it difficult to fully reproduce the results obtained by the original authors. Furthermore, replicating their experiments on another dataset allows us to assess whether the aforementioned claims generalize to new domains. Lastly, by evaluating their pretrained models on Wiki-auto, we are able to gain insight into the out-of-domain performance of these models.

[1] Replication according to the ACM definition: different team, different experimental setup. Also replication according to the NeurIPS definition: same code and analysis, but different data.

3. Methodology

This section details our methodology: first, the experimental data; second, the experimenal models; third, the experimental setup; and fourth, the computational requirements of our experiments.

The authors of the original paper made their code, open-source datasets and several pretrained models available on GitHub.[2] Because their code is of high quality, running it allows us to use the exact same model architectures, training and evaluation scripts for our replication study. We describe the data, models and our experimental setup in the following subsections.

3.1. Data

WikiLarge (Zhang and Lapata, 2017) is a dataset of complex-simple document pairs that were automatically collated from English Wikipedia and Simple English Wikipedia. Wiki-auto (Jiang et al., 2020) was derived from WikiLarge by aligning the simple output document with the complex input document at both the sentence and paragraph level. For all experiments, we utilize the preprocessed version of Wiki-auto from Cripwell et al. (2023b). In this version, each complex document consists of only the aligned paragraphs, and each simple document consists of only the aligned sentences within the aligned paragraphs. Moreover, each complex sentence is annotated with a simplification operation - delete, copy, rewrite or split - based on the simple sentences to which it is aligned. For example, if a complex sentence is aligned to multiple simple sentences, it is assigned the split operation. Documents with lots of deletion are removed from dataset; we refer to the original paper for more details on the preprocessing procedure.

Since the authors made their Wiki-auto datasets publicly available, we did not have to preprocess the data ourselves. However, as these datasets were only used for training and evaluating the planning models, they do not contain information on which sentences belong to the same paragraph. Meanwhile, fine-tuning certain simplification models also requires paragraph pairs. Therefore, we constructed a preprocessed paragraph-level

[2] https://github.com/liamcripwell/plan_simp

Data	Copy	Rephrase	Split	Delete
Wiki-auto	20.64	39.01	11.18	29.17
Newsela-auto	26.06	35.49	21.75	16.69

Table 1: Operation class distributions of Wiki-auto and Newsela-auto in percentages.

dataset by combining the information from the original Wiki-auto data with the datasets shared by the authors.

To illustrate the difference between the preprocessed Wiki-auto and Newsela-auto datasets, we highlight some characteristics also reported by the original authors. First, the number of document pairs is significantly higher for Wiki-auto (85,123) than for Newsela-auto (18,319). Second, the average number of sentences per complex document is much smaller for Wiki-auto (5.4) than for Newsela-auto (38.6). Third, percentage-wise, the Wiki-auto dataset contains more rephrase and delete operations, and less copy and split operations than the Newsela-auto dataset. The exact percentages are shown in Table 1.

3.2. Planning models

Cripwell et al. (2023b) experimented with several planning models, whose task is to predict a simplification operation - delete, copy, rewrite or split - for each sentence in a complex document. For example, their RoBERTa-based *classifier* simply takes a tokenized sentence as input and outputs a prediction score for each operation class. Their *contextual classifier* additionally attends over a high-level representation of the document context. This is a sequence of vector encodings for the sentences surrounding the input sentence, combined with custom positional embeddings indicating their relative distance to it.

On both Wiki-auto and Newsela-auto, the contextual classifier achieved the highest accuracy. Specifically, the best-performing variants used dynamic context, weight initialization and a context window radius of 13 sentences. During inference, using dynamic context means that the left context consists of previously simplified sentences, rather than complex ones. During training, the ground truth simplifications are used. Weight initialization means that the RoBERTa layers of the contextual classifier are initialised with weights from the context-independent classifier. For Newsela-auto, the most accurate variant also included document positional embeddings into the context, indicating the document quintile (1-5) that a given sentence falls into. This variant was used for plan-guidance by Cripwell et al. (2023a). Similarly, in this work, we fine-tune both planners - with and without document positional embeddings - on Wiki-auto, and utilize the variant with the highest accuracy to guide our simplification models.

3.3. Simplification models

We train all document simplification models from the original paper on Wiki-auto. That is, we fine-tune them on pairs of complex inputs and simple outputs. The original authors distinguished three model categories, each of which we briefly describe here.

3.3.1. Text-only

Text-only models take only a text sequence as input. They are trained by fine-tuning BART and a Longformer encoder-decoder to perform simplification on documents (BART_{doc}, LED_{doc}), paragraphs ($\text{BART}_{\text{para}}$, LED_{para}), and sentences ($\text{BART}_{\text{sent}}$). The sentence- and paragraph-level models are iteratively applied over a document in order to simplify it.

3.3.2. Context-aware

ConBART is a modification of the BART architecture, that takes both a sentence and a high-level representation of its document context as input. This context representation is constructed using the same strategy as for the planning models, with a context window radius of 13 sentences and a dynamic context mechanism. ConBART is iteratively applied over the sentences in a document in order to simplify it.

3.3.3. Plan-Guided

Each of the proposed models can be modified to take a simplification operation as control-token at the beginning of each text input. During training, the ground-truth operations are used as control-tokens. At inference time, the operations are generated by a planning model. The resulting systems are referred to as $\hat{O} \rightarrow h$, where h is the simplification model. If the ground-truth operations are used during inference, the resulting systems are referred to as $O \rightarrow h$. Furthermore, to align with the original paper, we rename $\hat{O} \rightarrow \text{BART}_{\text{sent}}$ to PG_{Dyn} and $O \rightarrow \text{BART}_{\text{sent}}$ to $\text{PG}_{\text{Oracle}}$.

3.4. Experimental setup

We use the code provided by the original authors for our experiments. It is complete, readable and runs without errors. Furthermore, it is well-documented, including instructions on how to leverage the pretrained models. The exact arguments used to train each planning and simplification model are not documented. Still, we are largely able to recover them

from careful inspection of the code and the training details outlined in the original paper. We use these arguments to train our models on Wiki-auto, and share them on GitHub[3] to make reproduction easy. We also provide our code for constructing the preprocessed paragraph-level dataset.

3.4.1. Training details

Despite being able to recover most arguments, we have to make a few assumptions about the training procedure. First of all, the authors mention training their simplification models until convergence, without defining convergence. We implement early stopping and train until the first epoch at which the validation loss does not improve. Then we select the model checkpoint from the epoch before. The authors also do not specify when to stop training the planning models. We decide to train them for 10 epochs, and select the checkpoint with the lowest validation macro F1-score. Moreover, there are some inconsistencies between the training details reported by Cripwell et al. (2023b) and Cripwell et al. (2023a). Both papers report different learning rates for their simplification models, and whereas the first paper mentions enforcing a minimum output length for BART_{doc}, the second does not. However, both papers report the same results for those models that they have in common. We use the training details specified in the second paper, since this is the one that we aim to replicate.

3.4.2. Inference

Following the original authors, we perform inference using beam search with a beam size of 5 and a maximum length of 1024 tokens. Furthermore, for our out-of-domain experiments, we utilize all models that were pretrained on Newsela-auto and made available by the authors. These include one planning model, which is the best variant of the contextual classifier, and four simplification models, namely LED_{para} and the plan-guided modifications of $\text{BART}_{\text{sent}}$, ConBART and LED_{para}. Because Wiki-auto does not have multiple simplification levels, we manually specify a target reading level of 3 (the second simplest) for our experiments.

3.4.3. Evaluation metrics

We evaluate each model using the same evaluation scripts and metrics as the original authors. Thus, we evaluate the planning models using the F1-score for each operation class, as well as the micro and macro averages. To evaluate the simplification models, we leverage BARTScore (Yuan et al., 2021) and SMART (Amplayo et al., 2022) as analogs for meaning preservation and fluency. Furthermore, we assess readability using the Flesch-Kincaid grade level (FKGL) (Kincaid et al., 1975), and simplicity using SARI (Xu et al., 2016).

[3] https://github.com/JanB100/doc_simp

Planning model	Training time
Classifier	62
Dyn. context	97
+ docpos	102

Table 2: Training time per planning model in minutes. **Dyn. Context** is the contextual classifier with r = 13, dynamic context and weights initialised using the classifier weights.

Simplification model	Training time
BART_{doc}	72
$\text{BART}_{\text{sent}}$	111
$\text{BART}_{\text{para}}$	54
LED_{doc}	146
LED_{para}	136
ConBART	109

Table 3: Training time per simplification model in minutes.

3.5. Computational requirements

We run all training and inference processes on two NVIDIA A100 GPUs with 40 GB memory. In line with the original paper, we use a batch size of 32 to train the planning models on Wiki-auto. The time needed to train each planning model for 10 epochs on 2 GPUs is shown in Table 2. Note that because of weight initalization, one can only train the contextual classifier after the context-independent classifier has been trained.

The original authors used a batch size of 16 to train their simplification models on Newsela-auto. However, using the same batch size to train on Wiki-auto results in memory issues. Therefore, we leverage a batch size of 8 and accumulate the gradients over 2 batches. The time needed to train each simplification model without plan-guidance on 2 GPUs is shown in Table 3. The training times with plan-guidance are approximately equal. We refer to the original paper for statistics on inference times and parameter counts.

4. Results and Discussion

This section presents in results of our experiments on Wiki-auto. First, the planning results. Second, the text simplication results. Third, the effectiveness under out-of-domain conditions.

Model	Copy	Rephrase	Split	Delete	Micro	**Macro**
Classifier	40.0 (42.1)	53.0 (52.9)	42.3 (42.6)	48.9 (49.0)	48.2 (48.4)	46.0 (46.7)
Dyn. context	**45.7** (44.8)	56.0 (57.9)	**42.9** (42.4)	**57.1** (54.8)	**52.8** (52.8)	**50.5** (50.0)
+ docpos	44.2 (43.7)	**58.6** (55.4)	39.8 (43.6)	52.1 (56.7)	52.4 (52.3)	48.7 (49.9)

Table 4: Reproduced (and original) Planning Accuracy (class and average F1-scores) on Wiki-auto. **Dyn. Context** is the contextual classifier with r = 13, dynamic context and weights initialised using the classifier weights.

System	**BARTScore ↑**			**SMART ↑**			**FKGL ↓**	**SARI ↑**	**Length**	
	P ($r \rightarrow h$)	R ($h \rightarrow r$)	F1	P	R	F1			Tok.	Sent.
Input	-2.48	-1.65	-2.06	55.9	64.1	59.3	9.64	16.7	155.3	5.5
Reference	-0.61	-0.61	-0.61	100	100	100	6.59	97.2	97.1	4.5
$BART_{doc}$	-2.04	-2.09	-2.07	**62.9**	53.9	57.2	9.66	45.2	96.6	2.3
$BART_{sent}$	-2.11	-1.91	-2.01	58.1	62.8	59.7	6.95	43.1	111.5	5.2
$BART_{para}$	-2.01	-1.90	-1.96	62.0	62.6	61.6	7.69	43.7	107.6	4.5
LED_{doc}	-2.21	-1.61	**-1.91**	60.7	**68.3**	**63.7**	8.42	34.3	145.7	5.5
LED_{para}	-2.26	**-1.60**	-1.93	60.1	68.0	63.3	8.73	31.1	151.0	5.6
ConBART	-2.19	-1.81	-2.00	58.5	64.9	60.9	7.54	39.4	128.6	5.4
PG_{Dyn}	**-1.85**	-2.05	-1.95	61.3	59.9	59.9	6.46	**48.6**	90.2	4.4
$\hat{O} \rightarrow$ ConBART	-1.86	-2.03	-1.95	61.5	60.1	60.1	6.54	48.4	92.5	4.4
$\hat{O} \rightarrow BART_{para}$	-1.86	-2.04	-1.95	60.7	59.8	59.6	6.40	48.4	93.3	4.5
$\hat{O} \rightarrow LED_{para}$	-1.87	-1.94	**-1.91**	62.5	61.7	61.4	7.11	47.2	102.6	4.5
PG_{Oracle}	**-1.57**	-1.72	-1.65	67.5	67.7	67.5	6.39	**56.4**	89.6	4.5
$O \rightarrow$ ConBART	-1.59	-1.70	-1.65	67.7	67.8	67.7	6.48	56.1	91.9	4.5
$O \rightarrow BART_{para}$	-1.58	-1.73	-1.66	67.0	67.1	67.0	6.28	56.1	91.1	4.5
$O \rightarrow LED_{para}$	-1.62	**-1.63**	**-1.62**	**69.0**	**69.1**	**69.0**	7.04	55.0	100.9	4.5

Table 5: **Results of document simplification systems on Wiki-auto.** For BARTScore, *h* is the hypothesis and *r* is the reference.

4.1. Planning results

Table 4 summarizes the results of training and evaluating our planning models on Wiki-auto. The planning accuracies of our models are close to those originally reported in Cripwell et al. (2023b, Table 2), indicating a succesful reproduction. In particular, the improvement of the contextual classifiers over the context-free classifier is the biggest for the delete operation, and the smallest for the split operation. This confirms the intuition of the original authors that deletion is mostly context dependent, while splitting is mostly context independent. However, all F1-scores are relatively low. As indicated by the authors, this is likely a result of Wiki-auto being an inferior simplification corpus. In line with the original results, we find the macro F1-score of the contextual classifier on Wiki-auto to be optimal when not using document positional embeddings. We hypothesize that the small document lengths (as shown in Section 3.1) make these embeddings redundant, and utilize the contextual classifier without document positional embeddings for our plan-guided simplification systems.

4.2. Simplification results

Table 5 shows the results of training and evaluating our document simplification systems on Wiki-auto. It corresponds to the Newsela-auto results in Cripwell et al. (2023a, Table 3). We leverage these results to assess the main claims made by the original authors:

1. Considering all metrics, text-only models that take as input either a sentence ($BART_{sent}$) or a whole document ($BART_{doc}$, LED_{doc}) underperform compared to models that have access to a local document context ($BART_{para}$, LED_{para}, ConBART).

2. Plan-guided models outperform their standard counterpart on all metrics.

The first claim is concerned with all models that are not guided by a simplification plan. Considering only those models, we find that $BART_{sent}$

Complex document

Silvano "Nano" Campeggi (] ; January 23, 1923 – August 29, 2018) was an Italian artist who designed and produced the artwork for the posters of many classic Hollywood films. His iconic images are associated with the golden era of Hollywood and Campeggi is now generally regarded as the most important graphic artist and poster designer in the history of American cinema.

In the following decades, Campeggi designed and produced the poster and advertising graphics for over 3000 films, working not only under contract with the MGM studios, but also with Warner Brothers, Paramount, Universal, Columbia Pictures, United Artists, RKO, Twentieth-Century Fox and several other movie studios. Sixty- four of the films he illustrated won Oscars, including "Casablanca", "Ben-Hur", "Singin' in the Rain", "An American in Paris", "West Side Story", "Exodus", "Breakfast at Tiffany's", and "Gigi".

Campeggi died on 29 August 2018, at the age of 95.

Simple document

Silvano "Nano" Campeggi (]; January 23, 1923 – August 29, 2018) was an Italian artist. He designed and produced the artwork for the posters of many classic Hollywood movies.

Campeggi was known for his poster designs for "Casablanca", "Singin' in the Rain", and "Breakfast at Tiffany's".

Campeggi died on August 29, 2018 in Florence from respiratory failure at the age of 95.

Simplification by pretrained $\hat{O} \rightarrow$ LED$_{para}$

Silvano "Nano" Campeggi (; born January 23, 1923 – August 29, 2018) was an Italian artist. He designed and produced the artwork for the posters of many Hollywood movies. Campeggi is now known as the most important picture artist and poster designer in the history of American movies.

Over the following many years, Campeggi designed and produced the poster and advertising pictures for many movies. Sixty-four of the films he illustrated won awards. They included "Casablanca," "Ben-Hur", "Singin' in the Rain," "An American in Paris" and "Gigi."

Campeggi died on 29 August 2018, when he was 95.

Simplification by BART$_{para}$

Silvano ""Nano"" Campeggi (]; January 23, 1923 – August 29, 2018) was an Italian artist. He designed and produced the artwork for the posters of many classic Hollywood movies.

Campeggi designed and produced the poster and advertising graphics for over 3000 movies, working not only under contract with the MGM studios, but also with Warner Brothers, Paramount, Universal, Columbia Pictures, United Artists, RKO, Twentieth Century Fox and several other movie studios. Sixty-four of the movies he illustrated won Oscars, including "Casablanca", "Ben-Hur", "Singin' in the Rain", "An American in Paris", "West Side Story", "Exodus", "Breakfast at Tiffany's", and "Gigi".

Campeggi died on 29 August 2018, at the age of 95

Simplification by PG$_{Dyn}$

Silvano "Nano" Campeggi (]; January 23, 1923 – August 29, 2018) was an Italian artist. He designed and produced the artwork for the posters of many classic Hollywood movies. His iconic images are associated with the golden era of Hollywood and Campeggi is generally regarded as the most important graphic artist and poster designer in the history of American cinema.

Sixty-four of the films he illustrated won Oscars, including "Casablanca", "Ben-Hur", "Singin' in the Rain", "An American in Paris", "West Side Story", "Exodus", "Breakfast at Tiffany's", and "Gigi".

Campeggi died on 29 August 2018 in Milan at the age of 95.

Simplification by $\hat{O} \rightarrow$ BART$_{para}$

Silvano "Nano" Campeggi (]; January 23, 1923 – August 29, 2018) was an Italian artist. He designed and produced the artwork for the posters of many classic Hollywood movies. His iconic images are associated with the golden era of Hollywood.

Campeggi illustrated over 3000 movies, including "Casablanca", "Ben-Hur", "Singin' in the Rain", "An American in Paris", "West Side Story", "Exodus", "Breakfast at Tiffany's", and "Gigi".

Campeggi died on 29 August 2018, at the age of 95.

Figure 2: A complex-simple document pair from Wiki-auto, along with the corresponding outputs of three document simplification systems trained on Wiki-auto and one system pretrained on Newsela-auto.

and BART$_{para}$ perform best overall. While LED$_{doc}$ achieves the highest BARTScore and SMART F1-scores, its outputs are much longer than the references. Furthermore, whereas BART$_{doc}$ obtains the highest SARI scores, its outputs are not more readable than the inputs according to FKGL. This is largely a result of the sentences being relatively long, which SARI does not account for since it is a token-based metric. Thus, BART$_{sent}$ and BART$_{para}$ perform best overall and therefore the claim does not hold; BART$_{sent}$ even outperforms its contextual modification (ConBART) in terms of SARI. This suggests that having access to a local document context is more advantegeous for models performing simplification on Newsela-auto than for models performing simplification on Wiki-auto.

Regarding the second claim, we find that plan-guided models significantly outperform their standard counterparts in terms of SARI and FKGL. Although this is not necessarily true for SMART and BARTScore, the differences in F1-scores are small. Thus, we find that the claim largely holds. The underlying intuition is that document simplification is a highly complex task, and therefore decomposing it into two easier tasks, namely planning and generation, makes the full task simpler. Our results demonstrate that this is true even when the accuracy on the planning subtask is relatively low, and that using an oracle plan further increases performance across every metric.

Furthermore, we observe that the outputs of the text-only LED models are approximately as long as the inputs, and therewith much longer than the references and the outputs of all other models. We also find that this problem can be overcome by using a planning model in combination with the simplification model. However, our $\hat{O} \rightarrow$ LED$_{para}$ system does not outperform $\hat{O} \rightarrow$ BART$_{para}$, as was the case in the original paper. This is because the Longformer architecture was designed to process long text sequences, and the input paragraphs and documents in Newsela-auto are substantially longer than those in Wiki-auto.

In any case, it is important to realize that automatic evaluation metrics have their limitations. Specifically, when considering all metrics, we found that sentence-level models do not underperform compared to models that have access to a local document context (Claim 1). Nevertheless, it is conceivable that the latter class of models performs

Model	Copy	Rephrase	Split	Delete	Micro	**Macro**
Dyn. context + docpos	21.3	45.6	25.1	23.8	33.5	29.0

Table 6: Planning Accuracy (class and average F1-scores) on Wiki-auto for a model trained on Newsela-auto.

System	**BARTScore ↑**			**SMART ↑**			**FKGL ↓**	**SARI ↑**	**Length**	
	P ($r \to h$)	R ($h \to r$)	F1	P	R	F1			Tok.	Sent.
Input	-2.48	-1.65	-2.06	55.9	64.1	59.3	9.64	16.7	155.3	5.5
Reference	-0.61	-0.61	-0.61	100	100	100	6.59	97.2	97.1	4.5
LED_{para}	-2.68	-2.60	-2.64	39.4	45.5	41.4	4.55	35.5	91.4	5.7
PG_{Dyn}	-2.84	-2.81	-2.82	38.4	43.6	40.1	4.69	35.6	96.3	6.2
$\hat{\text{O}} \to$ ConBART	-2.89	-2.86	-2.88	37.7	42.6	39.3	4.55	35.6	93.6	6.2
$\hat{\text{O}} \to \text{LED}_{\text{para}}$	**-2.52**	**-2.52**	**-2.52**	**41.8**	**47.3**	**43.7**	4.87	**36.6**	98.4	5.9
$\text{PG}_{\text{Oracle}}$	-2.21	-2.47	-2.34	50.8	51.5	**51.1**	5.47	**44.9**	79.7	4.5
$\text{O} \to$ ConBART	-2.27	-2.52	-2.40	49.7	50.3	49.9	5.29	44.6	77.8	4.6
$\text{O} \to \text{LED}_{\text{para}}$	**-2.03**	**-2.30**	**-2.17**	**50.9**	**51.9**	**51.1**	5.32	43.7	82.2	4.7

Table 7: **Results on Wiki-auto for document simplification systems trained on Newsela-auto.** For BARTScore, *h* is the hypothesis and *r* is the reference.

better according to human judgements, because intuitively they should be better able to preserve the discourse structure of the document.

Figure 2 shows an example of a complex document from Wiki-auto, along with the simple document to which it is aligned and the corresponding outputs of four simplification systems. First of all, note that the simple document is no direct simplification of the complex document, as the last paragraph contains additional information. This is a result of the complex-simple document pairs in Wiki-auto being automatically collated. Second, note that the last sentence of the simplification created by PG_{Dyn} contains a factual error. This demonstrates that these systems are prone to hallucination, and therefore they should only be used in practice when their outputs are checked by humans. Most importantly, the right part of Figure 2 illustrates the effects of plan-guidance and access to a local document context onto the output. For example, we observe that $\text{BART}_{\text{para}}$ and $\hat{\text{O}} \to \text{BART}_{\text{para}}$ leave out different sentences, which shows that leveraging a document-level plan can make a difference even when the simplification model already operates at the paragraph-level. Conversely, we also observe that $\hat{\text{O}} \to \text{BART}_{\text{para}}$ merges multiple sentences in the second paragraph, while PG_{Dyn} is unable to do so. This reveals the ability of the simplification model to take advantage of operating at the paragraph-level, even when it is guided by a document-level plan.

4.3. Out-of-domain results

Table 6 shows the accuracy of the planning model, which was pretrained on Newsela-auto, when it is evaluated on Wiki-auto. The macro F1-score is close to that of a random classifier (25.0), indicating a poor out-of-domain performance. In particular, what the planner has learned about when to copy, split or delete a sentence does not at all generalize to Wiki-auto. Only for the rephrase operation does the acquired knowledge partially generalize, and 39.01% of the sentences in Wiki-auto fall into this class (Table 1). However, the class F1-score is still significantly lower than that of the same model trained on in-domain data (Table 4).

Table 7 displays the results of the full document simplification systems, which were pretrained on Newsela-auto, when they are evaluated on Wiki-auto. In terms of SARI, we find that LED_{para} performs better than its standard counterpart trained on in-domain data (Table 5). We interpret this as a certain capacity of generalization. Furthermore, we notice that the plan-guided models do not obtain significantly better results than LED_{para}. This is unsurprising given the poor out-of-domain performance of the planning model. However, we also find that leveraging the planner does not harm performance. Using oracle plans significantly increases performance, which demonstrates that plan-guidance can still be helpful when using simplfication models in an out-of-domain setting.

Compared to the simplification models trained on Wiki-auto, the models trained on Newsela-auto achieve significantly lower FKGL scores, indicating

that their outputs are easier to read. BARTScore, SMART and SARI compare these outputs to the references. As the references come from Wiki-auto, it is rather predictable that the best models trained on Wiki-auto achieve significantly better scores than the models trained on Newsela-auto. Even so, these results demonstrate that the models trained on Newsela-auto and Wiki-auto perform different types of transformations.

The difference between the in-domain and out-of-domain results can best be illustrated using an example. The lower left part of Figure 2 shows the output of the $\hat{\text{O}} \rightarrow \text{LED}_{\text{para}}$ system pretrained on Newsela-auto, given an input from Wiki-auto. In contrast to the other systems, $\hat{\text{O}} \rightarrow \text{LED}_{\text{para}}$ simplifies "graphic" to "picture", and "at the age of" to "when he was". Similar observations can be made upon inspection of more examples. This is because the system was essentially pretrained to rewrite news articles to a lower grade level, and this is not the same as rewriting English Wikipedia articles to Simple English Wikipedia articles. Yet, despite being less similar to the references, the outputs of the pretrained systems on Wiki-auto are in general fluent and easy to understand.

5. Conclusion

This section summarizes the main conclusions from our replication study of the paper Context-Aware Document Simplification (Cripwell et al., 2023a). The original paper evaluates a variety of document simplification systems on the Newsela-auto dataset, which requires a license to use. We leverage the code of the original authors to replicate their experiments on an open-source dataset, namely Wiki-auto, and share the exact arguments that we use to make reproduction easy. The accuracies of our planning models are close to those originally reported by the authors. Furthermore, we verify the claim that models guided by a document-level plan outperform their standard counterparts. We cannot verify the claim that models with access to a local document context perform better than those operating at the sentence- or document-level. Lastly, we evaluate the pretrained models shared by the original authors on Wiki-auto, and find that the planning model does not generalize well, while the simplification models partially generalize.

6. Ethics and Limitations

The motivation of this paper is the unavailability of the Newsela dataset used in (Cripwell et al., 2023a). The used Wiki-auto data (Zhang and Lapata, 2017) is freely available, hence offers an easy starting point for investigating document-level text simplification models and approaches. However, the alignment is of less quality than the unavailable Newsela data, and there is a need for a new open-access data set based on direct document-level text simplifications.

Our experiments are restricted to English and Encyclopedic data and we welcome research on text simplification in other languages and document genres.

7. Acknowledgements

Experiments in this paper were carried out on the National Supercomputer Snellius, supported by SURF and the HPC Board of the University of Amsterdam. Jan Bakker is partly supported by a conference grant of the master AI program at the University of Amsterdam. Jaap Kamps is funded in part by the Netherlands Organization for Scientific Research (NWO CI # CISC.CC.016, NWO NWA # 1518.22.105), and the University of Amsterdam (AI4FinTech program). Views expressed in this paper are not necessarily shared or endorsed by those funding the research.

8. Bibliographical References

Reinald Kim Amplayo, Peter J. Liu, Yao Zhao, and Shashi Narayan. 2022. Smart: Sentences as basic units for text evaluation.

Liam Cripwell, Joël Legrand, and Claire Gardent. 2023a. Context-aware document simplification. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13190–13206, Toronto, Canada. Association for Computational Linguistics.

Liam Cripwell, Joël Legrand, and Claire Gardent. 2023b. Document-level planning for text simplification. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 993–1006, Dubrovnik, Croatia. Association for Computational Linguistics.

Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. 2020. Neural CRF model for sentence alignment in text simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7943–7960, Online. Association for Computational Linguistics.

Peter Kincaid, Robert P. Fishburne, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems*, volume 34, pages 27263–27277. Curran Associates, Inc.

Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594, Copenhagen, Denmark. Association for Computational Linguistics.

9. Language Resource References

Jan Bakker. 2024a. *Document Simplification on Wiki-Auto (Data and Code)*. University of Amsterdam. Document Simplification Project, Document Simplification, 1.0. PID https://github.com/JanB100/doc_simp.

Jan Bakker. 2024b. *Document Simplification on Wiki-Auto (Models)*. University of Amsterdam. Document Simplification Project, Document Simplification, 1.0. PID https://huggingface.co/janbakker.

A. Data, code, and trained models

We share all our data, code, and pretrained models on GitHub (Jan Bakker, 2024a, https://github.com/JanB100/doc_simp) and HuggingFace (Jan Bakker, 2024b, https://huggingface.co/janbakker) building on the earlier code-base[4] and Wiki-auto corpus.[5] As Wiki-Auto is freely available, this offers an easy starting point for any researcher wanting to explore paragraph-level or document-level text simplification.

B. Additional Evaluation Results

Table 8 shows extra evaluation results for the document simplification systems trained and evaluated on Wiki-auto (complementing Table 5).

Table 9 shows extra evaluation results on Wiki-auto for the document simplification systems trained on Newsela-auto (complementing Table 7).

C. Example Simplifications

In addition to Figure 2, Table 10 and Table 11 show the outputs of four document simplification systems on two more examples from Wiki-auto.

[4] https://github.com/liamcripwell/plan_simp
[5] https://github.com/chaojiang06/wiki-auto

System	**BARTScore** Faith. ($s \rightarrow h$)	**BLEU** ↑	**ROUGE-L** ↑	**SARI** ↑	add	keep	delete
Input	-0.60	34.4	59.3	16.7	0.0	50.2	0.0
Reference	-1.65	100	100	97.2	96.1	97.2	98.5
$BART_{doc}$	-1.05	36.8	61.2	45.2	16.6	55.8	63.2
$BART_{sent}$	-0.92	39.9	63.8	43.1	17.8	56.1	55.5
$BART_{para}$	-0.90	41.2	64.9	43.7	17.5	57.8	55.7
LED_{doc}	-0.78	**42.7**	64.5	34.3	17.1	57.1	28.6
LED_{para}	-0.74	41.6	63.7	31.1	14.8	56.0	22.6
ConBART	-0.84	39.7	63.4	39.4	16.5	55.3	46.3
PG_{Dyn}	-1.02	39.9	64.1	**48.6**	**19.2**	58.8	**67.8**
$\hat{O} \rightarrow$ ConBART	-1.02	39.9	64.7	48.4	19.0	58.8	67.5
$\hat{O} \rightarrow BART_{para}$	-0.96	41.5	64.7	47.2	19.1	**59.5**	62.9
$\hat{O} \rightarrow LED_{para}$	-0.96	41.5	**65.3**	47.2	19.1	**59.5**	62.9
PG_{Oracle}	-1.02	51.3	**73.7**	**56.4**	23.2	**68.7**	**77.2**
$O \rightarrow$ ConBART	-1.02	51.3	73.6	56.1	22.9	68.5	76.9
$O \rightarrow BART_{para}$	-1.07	50.8	73.4	56.1	**23.5**	68.4	76.4
$O \rightarrow LED_{para}$	-0.96	**52.4**	**73.7**	55.0	23.4	**68.7**	72.9

Table 8: **Extra results of document simplification systems on Wiki-auto.** For BARTScore, *s* is the source and *h* is the hypothesis.

System	**BARTScore** Faith. ($s \rightarrow h$)	**BLEU** ↑	**ROUGE-L** ↑	**SARI** ↑	add	keep	delete
Input	-0.60	34.4	59.3	16.7	0.0	50.2	0.0
Reference	-1.65	100	100	97.2	96.1	97.2	98.5
LED_{para}	-1.62	22.4	49.5	35.5	5.1	42.3	59.1
PG_{Dyn}	-1.78	20.1	48.6	35.6	4.5	40.7	61.7
$\hat{O} \rightarrow$ ConBART	-1.89	19.4	48.0	35.6	4.2	40.1	**62.6**
$\hat{O} \rightarrow LED_{para}$	-1.44	**23.5**	**52.0**	**36.6**	**5.2**	**44.4**	60.3
PG_{Oracle}	-1.65	**31.0**	60.1	**44.9**	6.5	**54.6**	73.7
$O \rightarrow$ ConBART	-1.76	29.9	59.1	44.6	6.1	53.6	**74.1**
$O \rightarrow LED_{para}$	-1.30	**31.0**	**60.4**	43.7	**6.9**	54.1	70.2

Table 9: **Extra results on Wiki-auto for document simplification systems trained on Newsela-auto.** For BARTScore, *s* is the source and *h* is the hypothesis.

System	Output
Complex	*Ralph Steven Greco (May 25, 1942 – March 31, 2019) was the Johnson and Johnson Distinguished Professor, Emeritus of Surgery at Stanford University School of Medicine. He was a leader of the resident Well Being in surgery movement and surgical training program leader.* *Greco married to Irene L. Wapnir, M.D., professor of surgery at Stanford. Together they had 3 children. He died on March 31, 2019 at the age of 76.* *SARI = 22.6, FKGL = 2.51*
Simple	*Ralph Steven Greco (May 25, 1942 – March 31, 2019) was an American surgeon and sculptor. He was the Johnson and Johnson Distinguished Professor, Emeritus of Surgery at Stanford University School of Medicine. He was well-known for his advocacy for the well-being of surgery residents.* *He was married to Irene L. Wapnir. The couple had three children. Greco died from prostate cancer on March 31, 2019 in Stanford, California.* *SARI = 100.0, FKGL = 3.81*
$\text{BART}_{\text{para}}$	*Ralph Steven Greco (May 25, 1942 – March 31, 2019) was an American surgeon. Greco was the Johnson and Johnson Distinguished Professor, Emeritus of Surgery at Stanford University School of Medicine. He was a leader of the resident Well Being in surgery movement and surgical training program leader.* *Greco died on March 31, 2019 in Stanford, California at the age of 76.* *SARI = 55.4, FKGL = 7.21*
PG_{Dyn}	*Ralph Steven Greco (May 25, 1942 – March 31, 2019) was an American surgeon. He was the Johnson and Johnson Distinguished Professor, Emeritus of Surgery at Stanford University School of Medicine. He was a leader of the resident Well Being in surgery movement and surgical training program leader.* *Greco married Irene L. Wapnir, M.D., professor of surgery at Stanford University. Together they had 3 children. He died on March 31, 2019 at the age of 76.* *SARI = 32.6, FKGL = 2.91*
$\hat{\text{O}} \rightarrow \text{BART}_{\text{para}}$	*Ralph Steven "Ralph" Greco (May 25, 1942 – March 31, 2019) was an American surgeon. He was the Johnson and Johnson Distinguished Professor, Emeritus of Surgery at Stanford University School of Medicine. Greco died in Palo Alto, California from complications of a stroke on March 31 at the age of 90. He is a leader of the resident Well Being in surgery movement and surgical training program leader.* *Greco married to Irene L. Wapnir, M.D., professor of surgery at Stanford. Together they had 3 children. Greco died on March 31, 2019 at the age of 76.* *SARI = 34.2 , FKGL = 3.11*
Pretrained	*Ralph Steven Greco was a top doctor in California. Greco worked at the Stanford University School of Medicine from 1942 to 1954. He was also a leader in the movement and training program.* *Greco married to Irene L. Wapnir, a professor of surgery at Stanford. Together they had 3 children. He died on March 31, 2019, at the age of 76.* *SARI = 35.7, FKGL = 2.82*

Table 10: A complex-simple document pair from Wiki-auto, along with the outputs of three simplification systems trained on Wiki-auto and the $\hat{\text{O}} \rightarrow \text{LED}_{\text{para}}$ system pretrained on Newsela-auto.

System	Output
Complex	*In organic chemistry and biochemistry, a side chain is a chemical group that is attached to a core part of the molecule called the "main chain" or backbone. The side chain is a hydrocarbon branching element of a molecule that is attached to a larger hydrocarbon backbone. It is one factor in determining a molecule's properties and reactivity. A side chain is also known as a pendant chain, but a pendant group (side group) has a different definition.* *The placeholder R is often used as a generic placeholder for alkyl (saturated hydrocarbon) group side chains in chemical structure diagrams. To indicate other non-carbon groups in structure diagrams, X, Y, or Z are often used.* *SARI = 14.9, FKGL = 10.40*
Simple	*A side chain in organic chemistry is a chemical group. It is attached to the molecule's "main chain" or backbone. A side chain is also known as a pendant chain.* *Alkyl (saturated hydrocarbon) group side chains are represented by the placeholder R. Other non-carbon groups can be represented by X, Y, or Z.* *SARI = 100.0, FKGL = 4.52*
$\text{BART}_{\text{para}}$	*In organic chemistry and biochemistry, a side chain is a chemical group that is attached to a core part of the molecule called the "main chain" or backbone. The side chain, also known as a pendant chain, is one factor in determining a molecule's properties and reactivity. A pendant group (side group) has a different definition.* *The placeholder R is often used as a generic placeholder for alkyl group side chains in chemical structure diagrams. To indicate other non-carbon groups in structure diagrams, X, Y, or Z are often used.* *SARI = 27.7, FKGL = 9.34*
PG_{Dyn}	*In organic chemistry and biochemistry, a side chain is a chemical group that is attached to a core part of the molecule. The side chain is a hydrocarbon branching element of a molecule that is attached to a larger hydrocarbon backbone.* *The placeholder R is often used as a generic placeholder for alkyl group side chains in chemical structure diagrams.* *SARI = 28.5, FKGL = 11.9*
$\hat{\text{O}} \rightarrow \text{BART}_{\text{para}}$	*In organic chemistry and biochemistry, a side chain is a chemical group that is attached to a core part of the molecule called the "main chain" or backbone. The "side chain" is a hydrocarbon branching element of a molecule that is connected to a larger hydrocarbon backbone. A pendant group (side group) has a different definition.* *The placeholder R is often used as a generic placeholder for alkyl group side chains in chemical structure diagrams.* *SARI = 29.1, FKGL = 10.2*
Pretrained	*In organic chemistry, a side chain is a chemical group. It is attached to a part of the molecule called the "main chain" or backbone. The side chain is a group of atoms that are attached to a larger part of the molecule. It is one of the things that makes a molecule different. A side chain is also known as a pendant chain. But a pendant group (side group) has a different definition.* *SARI = 43.9, FKGL = 4.86*

Table 11: A complex-simple document pair from Wiki-auto, along with the outputs of three simplification systems trained on Wiki-auto and the $\hat{\text{O}} \rightarrow \text{LED}_{\text{para}}$ system pretrained on Newsela-auto.