Controllable Sentence Simplification via Operation Classification

Liam Cripwell
Université de Lorraine
CNRS/LORIA
liam.cripwell@loria.fr

Joël Legrand
Université de Lorraine
Centrale Supélec
CNRS/LORIA
joel.legrand@inria.fr

Claire Gardent
CNRS/LORIA
Université de Lorraine
claire.gardent@loria.fr

Abstract

Different types of transformations have been used to model sentence simplification ranging from mainly local operations such as phrasal or lexical rewriting, deletion and re-ordering to the more global affecting the whole input sentence such as sentence rephrasing, copying and splitting. In this paper, we propose a novel approach to sentence simplification which encompasses four global operations: whether to rephrase or copy and whether to split based on syntactic or discourse structure. We create a novel dataset that can be used to train highly accurate classification systems for these four operations. We propose a controllable-simplification model that tailors simplifications to these operations and show that it outperforms both end-to-end, non-controllable approaches and previous controllable approaches.

1 Introduction

Sentence simplification is a text generation task where a sentence is transformed into a simpler version of itself while preserving its core meaning. Transformations can involve several different rewrite operations such as word substitutions (lexical paraphrasing), structural modifications (e.g. sentence splitting or syntactic paraphrasing), and deletion.

Sentence simplification has been shown to aid reader comprehension (Mason, 1978; Williams et al., 2003; Kajiwara et al., 2013) and be a useful preprocessing step for downstream NLP tasks such as relation extraction (Miwa et al., 2010; Niklaus et al., 2016) and machine translation (Chandrasekar et al., 1996; Mishra et al., 2014; Li and Nenkova, 2015; Mishra et al., 2014; Štajner and Popovic, 2016).

Modern systems are data-driven, learning to perform transformations from parallel corpora of complex-simple $\langle C, S \rangle$ pairs. Although many different approaches have been attempted in the past, including statistical machine translation (SMT)-based methods, nearly all systems proposed in recent years follow a neural sequence-to-sequence approach. As these systems are trained in an end-to-end manner they are able to perform lexical and syntactic operations in combination and produce outputs with very high fluency.

However, given the black-box nature of these end-to-end systems, they are forced to rely on imperfect training corpora to implicitly learn rewrite operations, many of which occur infrequently (Jiang et al., 2020). As a result, neural end-to-end systems have been found to be overly conservative, often making no changes to the original text or being limited to the paraphrasing of short word sequences (Alva-Manchego et al., 2017; Maddela et al., 2021). In addition, these systems provide limited capacity for controllability and are unable to express alternative variants of the simplified text (Alva-Manchego et al., 2017; Cripwell et al., 2021).

In response, attempts have been made to produce controllable simplification systems that can constrain either the shape (length, amount of paraphrasing, lexical and syntactic complexity) of the output (Martin et al., 2020) or the type of transformation to be applied (e.g., copy, split, merge, rewrite, etc.) (Scarton and Specia, 2018; Dong et al., 2019; Scarton et al., 2020; Garbacea et al., 2021; Maddela et al., 2021).

In this work we propose a novel approach to sentence simplification which encompasses four global operations: whether to copy the input sentence (no simplification needed), rephrase it, split it based on syntax, or split it based on discourse structure. We create a novel dataset that can be used to train highly accurate classification systems for these four operations and propose a controllable-simplification model that tailors simplification to them. We compare our model with various alternatives and previous work, using both quantitative metrics and human evaluation, and show that our

Findings of the Association for Computational Linguistics: NAACL 2022, pages 2091 - 2103

model outperforms them. We also provide a qualitative analysis of the differences between the best models.

2 Related Work

2.1 Controllable Simplification

Scarton and Specia (2018), Nishihara et al. (2019) and Scarton et al. (2020) focus on tailoring outputs to specific reader groups based on the Newsela corpus (Xu et al., 2015), a popular simplification dataset which provides versions of news articles written for audiences of different reading levels. These works propose systems that adjust their simplifications to match one of these reading levels.

Martin et al. (2020) introduce a wider array of control attributes concerning grammatical features of the desired text such as compression level, amount of paraphrasing, and lexical and syntactic complexity.

Most recently, Maddela et al. (2021) propose a system that first uses a rule-based component (Niklaus et al., 2019) to generate candidates that have undergone splitting and deletion, before ranking them and sending the top n to a neural paraphasing model. Tunable settings in both components provide control over how much of the input is changed and whether to favour deletion or splitting. Their system received higher fluency and simplicity scores from human annotators compared to existing works.

However, at inference time, these methods all require the model to be explicitly informed about which reader level to cater to or which specific grammatical features or rewrite operations to prioritise. In constrast, we develop an approach that can not only be tuned manually, but can also operate in an end-to-end manner by inferring tunable parameters from the input.

2.2 Operation Classification

Alva-Manchego et al. (2017) and Dong et al. (2019) consider sentence simplification as a sequence-labeling problem, proposing systems that predict rewrite operations at the token-level before realising them downstream. Alva-Manchego et al. (2017) showed gains over previous approaches in terms of simplicity, but at the cost of fluency and meaning preservation. Dong et al. (2019) appears to resolve this trade off by introducing an enhanced interpreter that better constructs the resulting text.

Several existing works have attempted to use a classifer to determine which rewrite operation should be performed on an input at the sentence-level. Applying a sentence-level binary classifier as an initial step to predict whether simplification should be performed has been found to yield improved SARI results, reducing conservatism and spurious transformations (Scarton et al., 2020; Garbacea et al., 2021).

Multi-class systems have been explored with limited results. Scarton and Specia (2018) and Scarton et al. (2020) predict one of 4 operations (identical, elaboration, split, and merge) and feed this into an end-to-end model alongside the C as either a control token or one-hot vector. While Scarton and Specia (2018) fail to produce an accurate classifier or show any improvement over baselines, Scarton et al. (2020) show some gains in SARI when using predicted operation labels. However, their best classifier only yields an accuracy of 70%.

In the multi-class setting, models tend to struggle to accurately predict identity cases. We believe this is partially due to the training data used. All existing works use Cs from identical $\langle C, S \rangle$ pairs as training examples for this class, either alone or alongside standard Ss. The assumption here is that these pairs are identical because the C is already simplified. We will show that it is much more likely these items are unsimplified noise from the distribution of Cs and that excluding them from training data can dramatically improve accuracy.

We extend upon these sentence-level classification approaches by redefining the set of operations, creating comprehensive training and test data, and ultimately producing a classifier with much higher accuracy. We show that a pipeline approach which first predicts a rewrite operation outperforms existing end-to-end and controllable systems.

3 Operation Classification

We consider 4 operation types: *identity*, *rephrase*, *syntax-split*, and *discourse-split*. The *identity* and *rephrase* classes are equivalent to *identical* and *elaboration* from Scarton et al. (2020). In contrast, we split the *split* class into two distinct groups to capture further nuances of sentence splitting, as was explored in Cripwell et al. (2021).

Syntax-split indicates that a split should be performed based on a syntactic construct, whereas *discourse-split* indicates that a split should be performed based on a discourse relation. Examples of

these can be found in Appendix A.

As we focus on single sentence simplification, we exclude the *merge* class used in (Scarton and Specia, 2018; Scarton et al., 2020).

3.1 Training Data

We construct training data for a simplification operation classifier by combining subsets of existing English datasets. We consider simplification datasets Wiki-auto, Newsela-auto [1] (Jiang et al., 2020), and MUSS (Martin et al., 2021) as well as dedicated splitting datasets WikiSplit (Botha et al., 2018) and D-CCNews (Cripwell et al., 2021).

Wiki-auto and Newsela-auto are automatically aligned $\langle C, S \rangle$ pairs extracted from Wikipedia and Newsela, respectively. MUSS contains 2.7M pairs mined from Common Crawl web data which are estimated paraphrases based on embedding distance. WikiSplit contains 1M split pairs mined from Wikipedia edit history, while D-CCNews contains discourse-split pairs mined from the CCNews corpus (Nagel, 2016). D-CCNews has two subsets: D-CCNews-C which contains single Cs, and D-CCNews-S which contains pairs of organic Ss and synthetic Cs. We include samples from both subsets. Table 1 provides a breakdown of the inclusions from each source.

We heuristically assign silver operation labels to sentences from these datasets as follows:

identity: Ss from the Wiki-auto and Newsela-auto *rephrase* and *syntax-split* sets. We can be fairly confident that Ss from known simplification datasets are sufficiently simplified.

rephrase: Cs from MUSS, Wiki-auto and Newsela-auto where there is no split in the output S and Levenshtein similarity between the C and S is less than 1 standard deviation above the mean (< 0.92). This is to exclude near-identical pairs.

syntax-split: Cs from WikiSplit, MUSS, Wiki-auto and Newsela-auto whose S exhibits a split and does not contain an identifiable discourse marker.

discourse-split: Cs from all datasets whose S contains a split and a discourse adverbial. [2]

We call the resulting dataset **IRSD**$_4^C$. [3] We also consider a 3-class subset which excludes the *identity* class (**IRSD**$_3^C$) to explore how results change when models are trained to always simplify.

3.2 Test Data

We use two datasets for evaluation. A random sample of 1% of the training data is set aside as a large (34K examples) silver test set. We also create a smaller gold test set by randomly sampling 100 items from each of the 4 classes in our silver test set and presenting them to 3 annotators instructed to select the most appropriate operation with which to simplify the text. Further details of the annotation process are in Appendix B.

We approved all annotations that received a majority label agreement and manually adjudicated cases where all annotators disagreed (11%). The mean Cohen's Kappa agreement score between annotators is 0.246, illustrating the difficulty of this task. In many cases, several operations could feasibly apply, and so assigning a single correct label is not always a perfect solution. Appendix C lists some examples of this.

3.3 Classification Model

We fine-tune pretrained RoBERTa models (Liu et al., 2019) with classification heads on IRSD$_4^C$ and IRSD$_3^C$. [4] Further training details are provided in Appendix D.

Results on Silver Test Data. Results can be seen in Figure 1. Accuracy on the silver test set (98%) is much higher than previous works: Scarton and Specia (2018) and Scarton et al. (2020) achieve mean accuracies of 51% and 70% for a similar 4-class task. Garbacea et al. (2021), who only train a binary (*simp*, *no-simp*) classifier achieve 81% accuracy.

Notably, the accuracy for the *identity* class is much higher than the 59% achieved by Scarton et al. (2020). This is perhaps in part due to our exclusion of Cs from identical $\langle C, S \rangle$ pairs in the *identity* training subset. We explored this hypothesis by using a test set containing Cs from identical pairs alongside the existing *identity* examples.

Figure 2 shows that doing so reduces performance on the *identity* class dramatically; the model only classifies 9.8% of these Cs as *identity* and 82.4% of them as *rephrase*. This suggests that

[1] We specifically use the aligned pairs used for simplification experiments in Jiang et al. (2020), which excludes identical pairs and those of readability levels 0-1, 1-2, and 2-3.

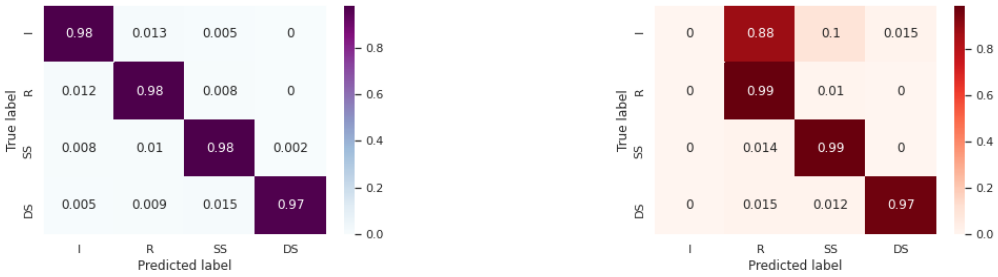
[2] D-CCNews is down-sampled to keep classes similar in size.

[3] Our data and code is available at https://github.com/liamcripwell/control_simp. Newsela data is excluded, subject to their terms of use, but can be provided upon request after receiving a licence.

[4] We use the pretrained *roberta-base* model available at https://huggingface.co/roberta-base.

Class	Source						Total
	WikiSplit	MUSS	Wiki-auto	Newsela-auto	D-CCNews-C	D-CCNews-S	
Identity (0)	-	-	513,436	338,798	-	-	**852,234**
Rephrase (1)	-	461,702	366,382	171,508	-	-	**999,592**
Syntax-Split (2)	633,900	53,008	68,357	88,669	-	-	**843,934**
Discourse-Split (3)	269,666	1,002	5,277	2,060	250,062	249,958	**778,025**
Total	**903,566**	**515,712**	**953,452**	**601,035**	**250,062**	**249,958**	**3,473,785**

Table 1: Data source distributions for each operation class in IRSD_4^C.



(a) 4-class (b) 3class

Figure 1: Normalised confusion matrix of (a) the four-class classifier and (b) the three-class classifier, evaluated on the silver-label test set.

these examples are from a distribution more similar to the *rephrase* examples and are possibly complex sentences themselves that have not been fully simplified in the source data. This observation validates our decision to exclude them.

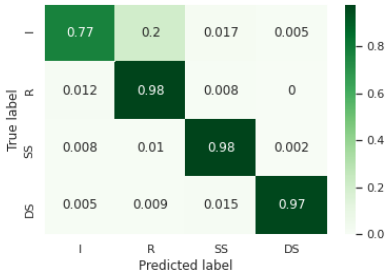


Figure 2: Normalised confusion matrix for the 4-class operation classifier, evaluated on the silver-label test set containing Cs from identical $\langle C, S\rangle$ pairs in the *identity* class.

Results on Gold Test Data. As shown in Figure 3, classification accuracy on the gold test set is considerably lower than on the silver data. *Identity* examples are often predicted as *rephrase*; *syntax-split* often as *discourse-split*; and *rephrase* examples regularly receive predictions across all four classes.

However, this aligns with our observations with respect to manual labelling difficulties. Often it is not immediately clear whether a particular example should be ignored or slightly rephrased. Similarly, it often seems plausible for either type of split to be performed. *Rephrase* is the broadest of the four classes, and so cases where any one of the other three classes could also apply should be expected.

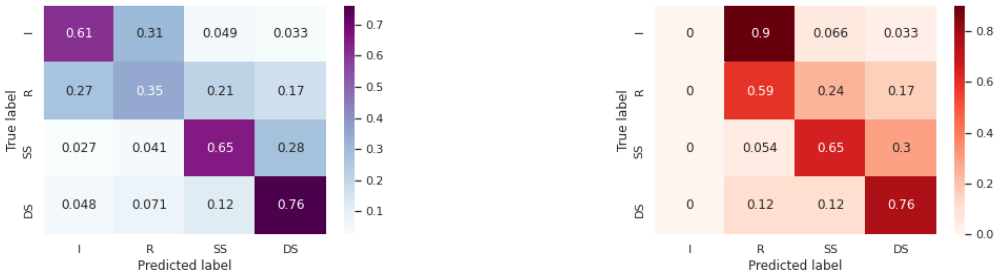
Despite being lower than on the silver examples, we believe these results show a strong signal of performance, with common mistakes being analogous to difficulties encountered by human annotators.

4 Sentence Simplification

4.1 Data

Training Data. For the sentence simplification task we use a modified version of IRSD^C which additionally includes target simplifications, i.e. $\langle C, o, S\rangle$ triples. We refer to this as $\textbf{IRSD}_4^S$ and its 3-class subset as $\textbf{IRSD}_3^S$.

For the *identity* class, we take all inputs from $\textbf{IRSD}_4^S$ labelled as *identity* and map them to themselves. For *rephrase* and *syntax-split*, we take the *rephrase* and *syntax-split* inputs and map them to their simplifications in the source datasets. We do the same for *discourse-split*, but, as D-CCNews-C instances do not contain simplifications, we replace them with additional $\langle C, S\rangle$ pairs from D-CCNews-S.



(a) 4-class (b) 3-class

Figure 3: Normalised confusion matrix of (a) the four-class classifier and (b) the three-class classifier, evaluated on the human-annotated test set.

Test Data. We first train and test our systems on $\text{IRSD}^{S}_{3/4}$ and Newsela-auto.

Next, in order to compare with past works, we perform evaluation on the Newsela-auto test set introduced by Maddela et al. (2021). It contains 24,035 *rephrases*, 9,208 *syntax-splits*, and 148 *discourse-splits*. We refer to this as **Newsela-M** and also include results on the subset with split in their reference S (Newsela-M (Split)). We use this test set so we can leverage pre-existing system outputs from past works for comparison.

Additionally, we evaluate on the ASSET corpus (Alva-Manchego et al., 2020) which is a much smaller test set (359 examples) containing 10 human-written references per input. All test examples have at least one *rephrase* reference, 248 have at least one *syntax-split* reference, 12 have at least one *discourse-split* reference, and 0 have an *identity* reference.

4.2 Models

Existing Systems We consider a number of past works for comparison: (i) **Hybrid** (Narayan and Gardent, 2014), an older system with a probabilistic splitting component combined with an MT-based lexical paraphraser; (ii) **BERT**, pretrained encoder-decoder transformer (BERT_{base}) finetuned on simplification which achieved state-of-the-art performance (Jiang et al., 2020); (iii) **EditNTS** (Dong et al., 2019), a recent model using operation prediction; and (iv) **MadExP** (Maddela et al., 2021), current state-of-the-art controllable system. [5] We exclude other systems which require conditioning on specific reading levels.

Baseline End-to-End Model. We include end-to-end baselines that are trained to perform $C \rightarrow S$ with no additional information. These are used to gauge whether our controllable models are competitive with a black-box approach. We use the BART architecture (Lewis et al., 2020) and fine-tune a pretrained model with a language-modelling head on $\langle C, S \rangle$ pairs from 4 distinct datasets: IRSD^{S}_{4} (**BART**$_4$), IRSD^{S}_{3} (**BART**$_3$), Wiki-auto (**BART**$_W$) and Newsela-auto (**BART**$_N$). [6]

Controllable Model. Next, we train an end-to-end generative model to perform $\langle C, o \rangle \rightarrow S$, where o is an operation label. The o is used as a control token prepended to the input sequence for C. We use the same BART architecture as our end-to-end baselines.

From this model, we construct several systems: (i) an oracle baseline (**Ctrl**$_{Oracle}$) taking the silver operation label and performing generation as an end-to-end task; (ii) a pipeline system using a classifier to predict o before running the generative model.

We refer to different configurations as **Ctrl**$_{i,j}$, where i is the number of classes the classifier is trained on and j is the number of classes the generator is trained on. E.g. $\text{Ctrl}_{3,4}$ uses a classifier trained on IRSD^{C}_{3} and a generator trained on IRSD^{S}_{4}. [7] We expect that using the 4-class classifier will result in more conservative outputs. Using the 3-class generator could allow more model capacity to focus on simplification. Conversely, the extra training data used by the 4-class generator could improve general performance.

[5] We use system outputs from versions of all of these models that have been trained on Newsela-Auto.

[6] We use the pretrained *facebook/bart-base* model available at https://huggingface.co/facebook/bart-base.

[7] For $\text{Ctrl}_{4,3}$ any inputs classified as *ignore* are returned without being passed to the generator.

Model	IRSD_4^S				IRSD_3^S		Newsela-auto			
	P_{BERT}	SARI	R_{Split}	P_{Split}	P_{BERT}	SARI	P_{BERT}	SARI	R_{Split}	P_{Split}
Input	0.83	27.4	0.00	0.00	0.77	25.4	0.53	15.9	0.00	0.00
Reference	0.99	80.1	1.00	1.00	0.99	95.3	0.99	94.1	1.00	1.00
End-to-End Models										
BART_W	0.81	35.0	0.18	0.85	0.76	34.7	0.54	24.6	0.05	**0.64**
BART_N	0.77	38.9	0.64	0.81	0.74	42.0	**0.56**	**35.9**	0.46	0.59
BART_3	0.85	50.6	0.82	0.94	0.81	54.9	0.55	27.3	0.27	0.59
BART_4	0.86	51.2	0.85	0.93	0.82	55.7	**0.56**	26.9	0.21	0.62
Controllable Models with predicted control-tokens										
$\text{Ctrl}_{3,3}$	0.83	50.6	**0.99**	0.93	0.82	58.5	0.54	33.6	0.48	0.54
$\text{Ctrl}_{3,4}$	0.84	51.2	**0.99**	0.93	**0.83**	59.4	0.55	**35.9**	**0.49**	0.54
$\text{Ctrl}_{4,3}$	0.86	52.9	**0.99**	**0.98**	**0.83**	59.5	0.55	30.7	0.45	0.56
$\text{Ctrl}_{4,4}$	**0.87**	**55.1**	**0.99**	**0.98**	**0.83**	**60.4**	**0.56**	32.4	0.45	0.56
Controllable Models with Oracle control-tokens										
Ctrl_{Oracle}	**0.87**	**55.5**	**1.00**	**1.00**	**0.83**	**60.7**	**0.57**	**38.3**	**0.99**	**1.00**

Table 2: Automatic sentence simplification results on the IRSD_4^S, IRSD_3^S and Newsela-auto test sets.

5 Experimental Setup

5.1 Automatic Evaluation

The most common evaluation metrics used in text simplification are BLEU and SARI, with SARI being viewed as the more effective at describing simplicity. Both focus primarily on lexical similarities between the reference and system output without consideration for structural simplification.

A recent meta-analysis (Alva-Manchego et al., 2021) of automated text simplification evaluation shows that the precision-based BERTScore (P_{BERT}) (Zhang et al., 2019) is most highly correlated with human judgements. As P_{BERT} is very effective at identifying low quality simplifications, the authors recommend using it as a primary test of quality before referring to other metrics like SARI.

We report P_{BERT} and SARI as our primary metrics [8] and also use the split recall (R_{Split}) to evaluate how often the model performs splitting in known cases. We value recall over precision as it gives a better indication of whether a model regularly performs splits, but have also included the precision (P_{Split}) for clarity.

5.2 Human Evaluation

We perform a human evaluation of simplification systems by having 3 annotators evaluate outputs. In order to consider a range of structurally diverse examples we use our classifier to label the Newela-M test set with predicted operations and randomly select 25 from each of the 4 classes (further details in Appendix B). We presented the annotators with the input C from each $\langle C, S \rangle$ pair alongside the reference S and outputs from selected systems. Judgements are made with respect to 3 criteria: fluency, adequacy, and simplicity.

Fluency refers to the grammaticality of the output; adequacy measures meaning preservation with respect to the input; and simplicity measures the overall simplicity of the result. We followed standard practice by having these criteria judged on a 1-5 Likert scale and averaging the results. For simplicity, we advised workers that a high score can be given to an output identical to the input if there is little to no obvious changes that would make the sentence simpler.

We consider the following systems for comparison: EditNTS, MadExp, BART_N, BART_4, and $\text{Ctrl}_{4,4}$. This allows us to compare our systems to strong recent works and examine the effect of (i) using IRSD^S vs Newsela training data and (ii) using our controllable model vs an end-to-end approach.

6 Results and Discussion

Automatic evaluation results are shown in Table 2.

IRSD^S vs Other Data Models trained with IRSD^S ($\text{BART}_{3/4}$ and $\text{Ctrl}_{*,*}$) greatly outperform those trained on other datasets ($\text{BART}_{N/W}$) across every metric on the IRSD^S test sets. On the Newsela test set, IRSD^S models perform at least as well as BART_N. This is unsurprising as IRSD^S is much larger than Newsela and contains many of

[8] The EASSE python library (Alva-Manchego et al., 2019) is used for calculation.

Model	Training	Newsela-M			Newsela-M (Split)			ASSET	
		P_{BERT}	SARI	R_{Split}	P_{Split}	P_{BERT}	SARI	P_{BERT}	SARI
Hybrid	Newsela-auto	0.39	30.2	0.17	0.42	0.39	31.9	0.43	30.5
BERT	Newsela-auto	0.46	32.2	0.40	0.46	0.47	34.5	0.59	35.2
EditNTS	Newsela-auto	0.49	29.3	0.32	0.45	0.53	30.8	0.54	31.4
MadExP	Newsela-auto	0.43	**36.0**	0.41	0.48	0.43	37.4	0.59	36.2
BART_N	Newsela-auto	0.54	34.0	0.52	0.48	0.58	37.1	**0.64**	**36.4**
BART_3	IRSD_3^S	0.54	25.0	0.31	0.49	0.58	28.8	**0.64**	34.3
BART_4	IRSD_4^S	**0.55**	25.3	0.25	**0.51**	0.58	28.6	**0.64**	33.7
$\text{Ctrl}_{3,3}$	IRSD_3^S	0.54	33.4	**0.54**	0.43	0.58	35.9	**0.64**	34.1
$\text{Ctrl}_{3,4}$	IRSD_4^S	**0.55**	35.6	**0.54**	0.43	**0.59**	**37.8**	**0.64**	33.8
$\text{Ctrl}_{4,4}$	IRSD_4^S	0.54	30.4	0.51	0.45	**0.59**	34.5	**0.64**	33.5
Ctrl_{Oracle}	IRSD_4^S	**0.56**	**37.3**	**1.00**	**0.99**	**0.59**	**38.6**	-	-

Table 3: Comparison with existing systems and baselines. Oracle labels are acquired by applying the same heuristics used in the creation of IRSD^S. Note that the oracle labels for these test sets do not contain *identity* cases.

System	Fluency	Adequacy	Simplicity	Mean
Ref.	4.65^{**}	3.95^{**}	4.37^{*}	4.32
EditNTS	3.81^{**}	3.83^{**}	3.91^{**}	3.85
MadExP	3.74^{**}	3.52^{**}	3.97^{**}	3.75
BART_N	4.68	4.26^{**}	**4.38***	4.44
BART_4	4.71	**4.74**	4.14	4.53
$\text{Ctrl}_{4,4}$	**4.77**	**4.74**	4.20	**4.57**

Table 4: Human evaluation results for selected simplification systems and baselines. Ratings significantly different from $\text{Ctrl}_{4,4}$ are denoted with $*$ ($p < 0.05$) and $**$ ($p < 0.01$). Significance was determined with a Student's t-test.

the same examples. However, it shows the diversity of IRSD^S does not reduce Newsela-specific performance.

On Newsela test data, using the 3-class classifier ($\text{Ctrl}_{3,*}$) yields higher SARI and R_{Split} than the 4-class case. This is likely because *identity* is never predicted thereby encouraging less conservative simplification on a test set where most examples are simplified (Maddela et al. (2021) excludes all examples with high or low similarity between the input and the reference from the test set).

End-to-End vs Controllable Controllable systems outperform their end-to-end counterpart on all metrics and datasets. In particular, they show a large increase in R_{Split}, suggesting that explictly triggering splits via control tokens greatly improves a model's ability to correctly administer splits where needed. Using silver operation labels in Ctrl_{Oracle} shows universally higher scores than classifier-based pipelines, indicating that there is still room for improvement in terms of classification performance.

Existing Systems Comparative results with existing systems are summarised in Table 3. All of our systems achieve much higher P_{BERT} scores than any existing system. This suggests that merely using the BART architecture yields much more fluent outputs than other models. MadExP, which receives the highest SARI scores, actually receives much lower P_{BERT} than almost any other system, indicating that although it may be simplifying the text well, outputs might be less fluent.

$\text{Ctrl}_{3,4}$ achieves the highest scores on Newsela-M, except for being slightly beaten by MadExP on SARI. It is expected to be better than $\text{Ctrl}_{4,4}$ here as Newsela-M does not include any identical $\langle C, S \rangle$ pairs and therefore penalises copying inputs. Ctrl_{Oracle} universally outperforms other systems.

On ASSET, BART_N actually achieves the highest SARI, closely followed by MadExP. We hypothesise that models trained on IRSD^S might achieve lower SARI as the training data includes examples from splitting datasets that do not exhibit any further simplification, leading to more conservative lexical changes when splitting is performed.

Human Evaluation Human evaluation results are shown in Table 4. $\text{Ctrl}_{4,4}$ scores highest in fluency and overall. Both $\text{Ctrl}_{4,4}$ and BART_4 score highest in adequacy. BART_N scores highest in simplicity.

All of our systems are rated highly accross all criteria and receive better average scores than even the reference. This clearly highlights the capability of pretrained generative models like BART to produce highly fluent texts. Further, we see that using a classifier trained on our data to predict operation-specific control-tokens further enhances

	Output
(1) Complex	*An acidic goo from the stomach dissolves the snail's flesh so the worm can re-swallow both stomach and prey.*
Simple	*An acidic goo from the stomach dissolves the snail's flesh, and the flatworm can then re-swallow both stomach and snail.*
BART_N	*[del] The goo dissolves the snail's flesh so the worm can re-swallow both stomach and prey.*
BART_4	*An acidic goo from the stomach dissolves the snail's flesh so the worm can re-swallow both stomach and prey.*
$\text{Ctrl}_{4,4}$	*An acidic goo from the stomach dissolves the snail's flesh. **[split] This allows** the worm to re-swallow both stomach and prey.*
(2) Complex	*Archer's parents were Catholic in Protestant England, which became illegal.*
Simple	*Archer's parents were Catholic. Being Catholic was against the law in England at the time.*
BART_N	*Archer's parents were Catholic. They did not want the government to interfere with their religion.*
BART_4	*Archer's parents were Catholic in Protestant England. [del]*
$\text{Ctrl}_{4,4}$	*Archer's parents were Catholic in Protestant England, which became illegal.*
(3) Complex	*Many people around the world are now furious at an American dentist who paid to go lion hunting in Zimbabwe.*
Simple	*Palmer paid to go lion hunting in Zimbabwe, which is in Africa.*
BART_N	*People around the world are **mad** at a dentist. He was paid to go lion hunting in Africa.*
BART_4	*Many people around the world are now furious at an American dentist who paid to go lion hunting in Zimbabwe.*
$\text{Ctrl}_{4,4}$	*Many people around the world are now furious at an American dentist who paid to go lion hunting in Zimbabwe.*
(4) Complex	*The Senegalese capital, on the far western edge of Africa, looks nothing like the sleepy port city that Amadou Sy remembers from his youth.*
Simple	*Amadou Sy remembers Dakar as a quiet city.*
BART_N	*The Senegalese capital **[del]** does not look like the sleepy port city that Amadou Sy remembers from his youth.*
BART_4	*The Senegalese capital **[del]** looks nothing like the sleepy port city that Amadou Sy remembers from his youth.*
$\text{Ctrl}_{4,4}$	*The Senegalese capital is on the far western edge of Africa. [split] It looks nothing like the sleepy port city that Amadou Sy remembers from his youth.*

Table 5: Example system outputs illustrating commonly seen patterns. Blue/**bold** marks positive changes while red/underlined marks negative changes or errors.

performance in both fluency and simplicity. However, we also believe our training data limits simplicity at times due to examples from pure splitting datasets exhibiting no simplification but for a split.

We believe the relatively low adequacy rating given to the reference can partly be attributed to sentence alignment failures and cases where the S makes reference to terms mentioned earlier in their article that are not explicit in the C.

Qualitative Analysis We perform a qualitative analysis of system outputs from the human evaluation to get a better idea of differences between our models. Table 5 illustrates common patterns.

BART_N regularly produces the most simple output, but often over-simplifies to the point of removing important contextual information (e.g. item 1). It also sometimes fails to maintain the correct meaning of the input (e.g. items 2 and 3).

BART_4 often produces outputs very similar to $\text{Ctrl}_{4,4}$, but performs splitting much less regularly (e.g. items 1 and 4) which can uphold structural complexity.

$\text{Ctrl}_{4,4}$ outputs best retain the original meaning of the input. The benefit of having a classifier predict *identity* cases can be seen in item 2 where the other models end up rephrasing poorly or deleting important information. However, when performing splits, it fails to sufficiently rephrase, often keeping obviously complicated words (e.g. item 4).

7 Conclusion

In this work we present a new dataset for simplification operation classification and show that it can be used to produce classifiers of much higher accuracy than what has been proposed in existing studies. We show that a controllable system using such a classifier to predict control tokens outperfoms end-to-end baselines and existing systems on a range of datasets and receives extremely high ratings in fluency, adequacy and simplicity from human evaluators. However, this system does result in slightly lower simplicity ratings compared to reference texts and a Newsela baseline, suggesting that further improvements can be made to the system or dataset in order to achieve the best possible results across all criteria.

Acknowledgments

We thank the anonymous reviewers for their feedback. We gratefully acknowledge the support of the French National Research Agency (Gardent; award ANR-20-CHIA-0003, XNLG "Multilingual, Multi-Source Text Generation") and of Facebook AI Research (FAIR) Paris.

Experiments presented in this paper were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see https://www.grid5000.fr).

References

Fernando Alva-Manchego, Joachim Bingel, Gustavo Paetzold, Carolina Scarton, and Lucia Specia. 2017. Learning how to simplify from explicit labeling of complex-simplified text pairs. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 295–305, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020. ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4668–4679, Online. Association for Computational Linguistics.

Fernando Alva-Manchego, Louis Martin, Carolina Scarton, and Lucia Specia. 2019. EASSE: Easier automatic sentence simplification evaluation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 49–54, Hong Kong, China. Association for Computational Linguistics.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2021. The (Un)Suitability of Automatic Evaluation Metrics for Text Simplification. *Computational Linguistics*, pages 1–29.

Jan A. Botha, Manaal Faruqui, John Alex, Jason Baldridge, and Dipanjan Das. 2018. Learning to split and rephrase from Wikipedia edit history. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 732–737, Brussels, Belgium. Association for Computational Linguistics.

R. Chandrasekar, Christine Doran, and B. Srinivas. 1996. Motivations and methods for text simplification. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

Liam Cripwell, Joël Legrand, and Claire Gardent. 2021. Discourse-based sentence splitting. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 261–273, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yue Dong, Zichao Li, Mehdi Rezagholizadeh, and Jackie Chi Kit Cheung. 2019. EditNTS: An neural programmer-interpreter model for sentence simplification through explicit editing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3393–3402, Florence, Italy. Association for Computational Linguistics.

Cristina Garbacea, Mengtian Guo, Samuel Carton, and Qiaozhu Mei. 2021. Explainable prediction of text complexity: The missing preliminaries for text simplification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1086–1097, Online. 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.

Tomoyuki Kajiwara, Hiroshi Matsumoto, and Kazuhide Yamamoto. 2013. Selecting proper lexical paraphrase for children. In *Proceedings of the 25th Conference on Computational Linguistics and Speech Processing (ROCLING 2013)*, pages 59–73, Kaohsiung, Taiwan. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP).

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Junyi Jessy Li and Ani Nenkova. 2015. Detecting content-heavy sentences: A cross-language case study. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1271–1281, Lisbon, Portugal. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Mounica Maddela, Fernando Alva-Manchego, and Wei Xu. 2021. Controllable text simplification with explicit paraphrasing. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3536–3553, Online. Association for Computational Linguistics.

Louis Martin, Éric de la Clergerie, Benoît Sagot, and Antoine Bordes. 2020. Controllable sentence simplification. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4689–4698, Marseille, France. European Language Resources Association.

Louis Martin, Angela Fan, Éric de la Clergerie, Antoine Bordes, and Benoît Sagot. 2021. Muss: Multilingual unsupervised sentence simplification by mining paraphrases. *arXiv preprint arXiv:2005.00352*.

Jana M Mason. 1978. Facilitating reading comprehension through text structure manipulation. *Center for the Study of Reading Technical Report; no. 092*.

Kshitij Mishra, Ankush Soni, Rahul Sharma, and Dipti Sharma. 2014. Exploring the effects of sentence simplification on Hindi to English machine translation system. In *Proceedings of the Workshop on Automatic Text Simplification - Methods and Applications in the Multilingual Society (ATS-MA 2014)*, pages 21–29, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Makoto Miwa, Rune Sætre, Yusuke Miyao, and Jun'ichi Tsujii. 2010. Entity-focused sentence simplification for relation extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 788–796, Beijing, China. Coling 2010 Organizing Committee.

Sebastian Nagel. 2016. Cc-news.

Shashi Narayan and Claire Gardent. 2014. Hybrid simplification using deep semantics and machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 435–445, Baltimore, Maryland. Association for Computational Linguistics.

Christina Niklaus, Bernhard Bermeitinger, Siegfried Handschuh, and André Freitas. 2016. A sentence simplification system for improving relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 170–174, Osaka, Japan. The COLING 2016 Organizing Committee.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2019. Transforming complex sentences into a semantic hierarchy. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3415–3427, Florence, Italy. Association for Computational Linguistics.

Daiki Nishihara, Tomoyuki Kajiwara, and Yuki Arase. 2019. Controllable text simplification with lexical constraint loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 260–266, Florence, Italy. Association for Computational Linguistics.

C. Scarton, P. Madhyastha, and L. Specia. 2020. Deciding when, how and for whom to simplify. © 2020 The Author(s) and IOS Press. This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial Licence (http://creativecommons.org/licenses/by-nc/4.0/).

Carolina Scarton and Lucia Specia. 2018. Learning simplifications for specific target audiences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 712–718, Melbourne, Australia. Association for Computational Linguistics.

Sanja Štajner and Maja Popovic. 2016. Can text simplification help machine translation? In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 230–242.

Sandra Williams, Ehud Reiter, and Liesl Osman. 2003. Experiments with discourse-level choices and readability. In *Proceedings of the 9th European Workshop on Natural Language Generation (ENLG-2003) at EACL 2003*, Budapest, Hungary. Association for Computational Linguistics.

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.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with BERT. *CoRR*, abs/1904.09675.

A Splitting Types

In the paper we distinguish two types of sentence splits, based on the findings of Cripwell et al. (2021). A syntax-split is when the split is licensed by syntactic constructs such as relative clauses, VP or sentence coordinations, gerund or appositive constructions. On the other hand, discourse-splits are licensed by the presence of a discourse relation between two discourse units. Often discourse-splits require additional rewriting in order to fully preserve the discourse semantics which would otherwise be broken by a minimal split operation. Furthermore, they can usually be represented via many equivalent variants.

Table 6 shows an example of these differences. [9] The text in C1 contains a temporal discourse relation marked by ***and after this*** which is made explicit in the discourse-split output (S1a) by the adverbial ***Afterwards***. A possible variant exists (S1b) where an inverse adverbial connective (***Before this***) is used. The seconds tier of the table shows two syntax-split examples, where minimal rephrasing is required.

B Human Annotation

In order to compile the gold-label test set for the classification task we instructed 3 human annotators to assign the labels they considered most appropriate for 400 examples. These annotators were students enrolled in a local NLP master's program and were paid slightly above the minimum wage for their work. Nine items were identified as malformed and thus removed.

Annotations were completed through a web form interface (e.g. in Figure 4). For each of the 400 items, they were presented with the input sentence and required to select one of the four class labels. They were also given the option to flag examples as being malformed or incomprehensible (which we removed from the final set). Prior to their completion of the task, they were given a detailed description of each class along with a range of examples.

For the simplification output evaluation we instructed the same 3 annotators to give their judgements on outputs from 6 systems for 100 inputs randomly sampled from the silver-label test set. Again, this was done via a web form where each input text is provided followed by the outputs from each system (e.g. in Figure 5). Below this, the annotators select 1-5 for each of the outputs on the three quality criteria: fluency, adequacy, and simplicity.

Full text instructions for both human annotation tasks are provided in the supplementary materials.

C Difficult Labelling Examples

The main paper mentions cases where it is difficult to determine a single best rewrite operation. Table 7 shows some common examples of this.

D Training Details

During training of the RoBERTa classification models, we used a learning rate of $2e^{-5}$. The network has 12 hidden layers, a hidden size of 768, and was pretrained with the masked language modeling objective on 160GB of books and web content.

During training of the BART generative models, we used a learning rate of $3e^{-5}$. The network has 6 layers in each of the encoder and decoder, a hidden size of 768, and was pretrained to perform reconstruction of corrupted documents on a combination of books and Wikipedia data.

All of our finetuning experiments used a batch size of 32, performed dropout with a rate of 0.1 and early stopping as regularisation measures. All models were trained on a computing grid using 4 Nvidia RTX 2080 Ti GPUs (11GB memory) for an average of 24 hours. For each experiment we set aside 1% of the training set for validation.

For the generative models, at test time we generate output sequences by performing beam search with a beam size of 5 and restrict output to a maximum length of 128 tokens.

The use of the Newsela corpus is subject to a data sharing agreement from Newsela, Inc. This licence permits the data to be used for non-commercial research purposes.

E System Outputs

Table 8 contains example system outputs not included in the main paper which illustrate commonly seen patterns across systems.

[9] These examples are taken directly from Cripwell et al. (2021)

C1.	The Masovians were caught by surprise, since virtually without any defense the capital, Płock, fell **and after this** Mindaugas crossed the Vistula river and captured the fortress of Jazdów.
S1a.	The Masovians were caught by surprise, since virtually without any defense the capital, Płock, fell. **Afterwards**, Mindaugas crossed the Vistula river and captured the fortress of Jazdów.
S1b.	Mindaugas crossed the Vistula river and captured the fortress of Jazdów. **Before this**, the Masovians were caught by surprise, since virtually without any defense the capital, Płock, fell.
C2.	He settled in London, devoting himself chiefly to practical teaching.
S2.	He settled in London. He devoted himself chiefly to practical teaching.
C3.	It was a time to go back to nature, and the plastic flamingo quickly became the prototype of bad taste and anti-nature.
S3.	It was a time to go back to nature. The plastic flamingo quickly became the prototype of bad taste and anti-nature.

Table 6: An example of discourse- (1) vs. syntax-based (2) sentence splitting.

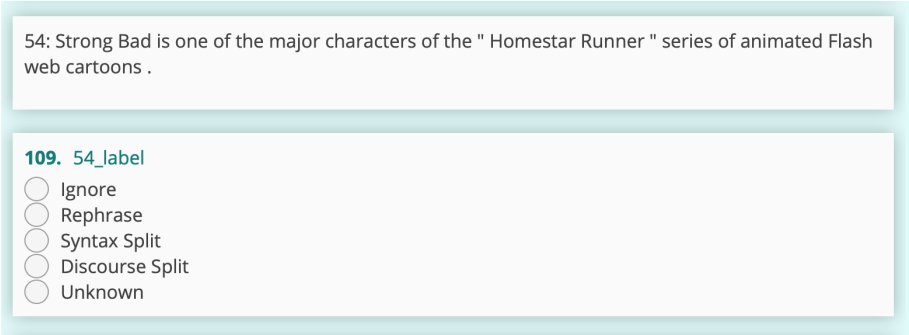


Figure 4: Section of annotation form used for gold-label classification test set creation.

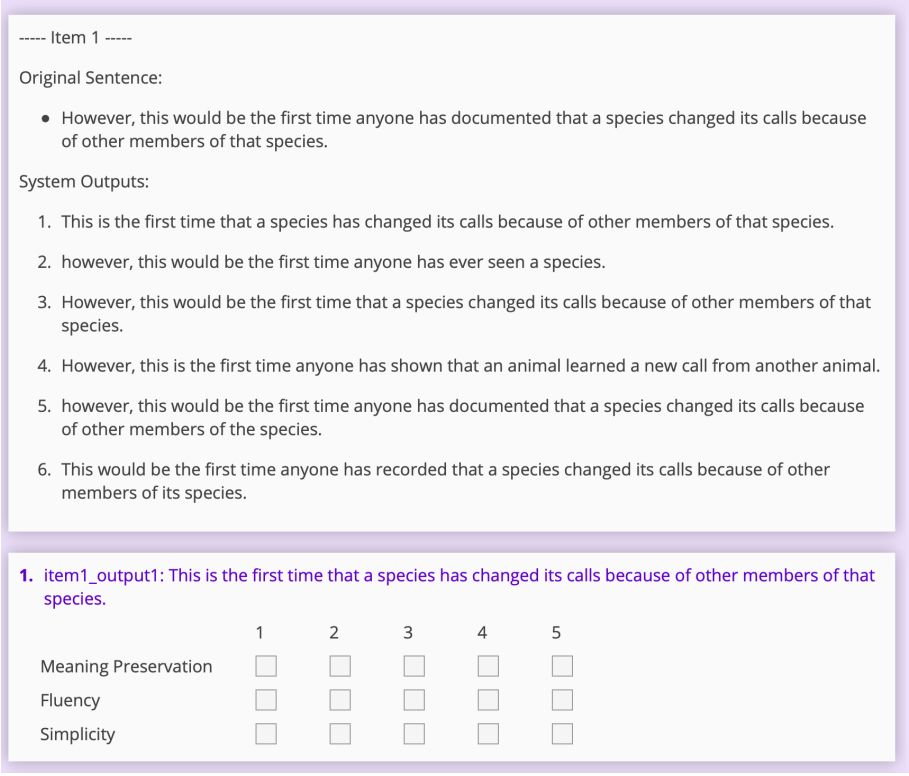


Figure 5: Section of annotation form used for human simplification evaluation.

C1.	He served as Mayor of The Hague from 2008 to 2017; he then took two acting positions in Drenthe and Amsterdam.
rephrase	He was Mayor of The Hague from 2008 to 2017 then took two acting positions in Drenthe and Amsterdam.
syntax-split	He served as Mayor of The Hague from 2008 to 2017. He then took two acting positions in Drenthe and Amsterdam.
discourse-split	He served as Mayor of The Hague from 2008 to 2017. Later, he took two acting positions in Drenthe and Amsterdam.
C2.	A bus stop is a designated place where buses stop for passengers to get on and off the bus.
identity	A bus stop is a designated place where buses stop for passengers to get on and off the bus.
rephrase	A bus stop is a place where buses stop for passengers.
C3.	He led Villa to victory in the inaugural League Cup in 1961 but was then sacked in 1964 on grounds of ill health.
syntax-split	He led Villa to victory in the inaugural League Cup in 1961. He was sacked in 1964 on grounds of ill health.
discourse-split	He led Villa to victory in the inaugural League Cup in 1961. However, he was sacked in 1964 on grounds of ill health.

Table 7: Some complex sentence examples where multiple rewrite operations are plausible.

	Output
Complex	*When that wasn't enough, Ysra, Sarah and three others who were also strong swimmers jumped into the water in order to give the boat more buoyancy.*
Simple	*Ysra, Sarah and three others who were also strong swimmers jumped out.*
BART_N	*When that wasn't enough, Ysra, Sarah and three others who were also strong swimmers jumped into the water.* *[del]*
BART_4	*When that wasn't enough, Ysra, Sarah and three others who were also strong swimmers jumped into the water in order to give the boat more buoyancy.*
$\text{Ctrl}_{4,4}$	*When that wasn't enough, Ysra, Sarah and three others who were also strong swimmers jumped into the water.* ***[split] They did this to*** *give the boat more buoyancy.*

Table 8: Additional system outputs illustrating commonly seen patterns. Blue/**bold** marks positive changes while red/underlined marks negative changes or errors.