Lexically Constrained Decoding with Edit Operation Prediction for Controllable Text Simplification

Tatsuya Zetsu[†], **Tomoyuki Kajiwara**[‡], **Yuki Arase**[†]
[†]Graduate School of Information Science and Technology, Osaka University, Japan
[‡]Graduate School of Science and Engineering, Ehime University, Japan
[†]`{zetsu.tatsuya, arase}@ist.osaka-u.ac.jp`
[‡]`kajiwara@cs.ehime-u.ac.jp`

Abstract

Controllable text simplification assists language learners by automatically rewriting complex sentences into simpler forms of a target level. However, existing methods tend to perform conservative edits that keep complex words intact. To address this problem, we employ lexically constrained decoding to encourage rewriting. Specifically, the proposed method predicts edit operations conditioned to a target level and creates positive/negative constraints for words that should/should not appear in an output sentence. The experimental results confirm that our method significantly outperforms previous methods and demonstrates a new state-of-the-art performance.

1 Introduction

Text simplification (Shardlow, 2014) paraphrases complex sentences into simpler forms. Controllable text simplification (Scarton and Specia, 2018; Nishihara et al., 2019; Agrawal et al., 2021) is a task in text simplification that aims to rewrite a sentence for an audience of a specific level. It is a crucial technique in assisting children and non-native speakers with language learning (Watanabe et al., 2009; Allen, 2009).

Text simplification can be performed based on three approaches: (1) translation-based, (2) edit-based, and (3) hybrid approaches. The translation-based approach, *e.g.*, (Nisioi et al., 2017; Zhang and Lapata, 2017; Kriz et al., 2019; Surya et al., 2019; Martin et al., 2022), formalizes text simplification as monolingual machine translation from complex to simple sentences. This approach can rewrite a sentence flexibly; however, it implicitly learns simplification operations through translation. The infrequent nature of simplification operations hinders a model from learning necessary operations, which makes the model conservative to maintain complex words intact (Zhao et al., 2018; Kajiwara, 2019). In contrast, the edit-based approach (Alva-Manchego et al., 2017; Dong et al., 2019; Kumar et al., 2020; Mallinson et al., 2020; Omelianchuk et al., 2021) rewrites an input by applying edit operations of `add` or `replace`, `keep`, and `delete` to words. This approach can address the conservativeness problem owing to explicit word-by-word edits. However, it lacks the flexibility to rewrite an entire sentence to drastically change its syntactic structure.

Finally, the hybrid approach takes advantages of the above two by applying lexical constraints to translation-based models. Nishihara et al. (2019) added weights to a loss function to bias a sequence-to-sequence (seq2seq) model to output certain words. Agrawal et al. (2021) biased a non-autoregressive simplification model by setting an initial state of decoding, considering the lexical complexity of a source sentence. The constraints in these studies were soft; in contrast, Kajiwara (2019) and Dehghan et al. (2022) applied a hard constraint using lexically constrained decoding to avoid outputting complex words. In spite of their success, these two methods lack flexibility in their constraints. They only use *negative* constraints to avoid outputting specified words. However, *positive* constraints, which encourage the output of specified words, are also valuable for text simplification.

In this study, we propose a hybrid method for controllable text simplification with flexible combinations of positive and negative constraints using NeuroLogic decoding (Lu et al., 2021). The proposed method predicts edit operations conditioned on a target level to generate positive and negative lexical constraints sensible to a target level. Experiments on Newsela (Xu et al., 2015) and Newsela-Auto (Jiang et al., 2020) reveal that the proposed method outperforms previous methods and achieves a new state-of-the-art performance. The codes and outputs of the proposed method will be released at `https://github.com/t-zetsu/ConstrainedTS`.

Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022), pages 147 - 153
December 8, 2022 ©2022 Association for Computational Linguistics

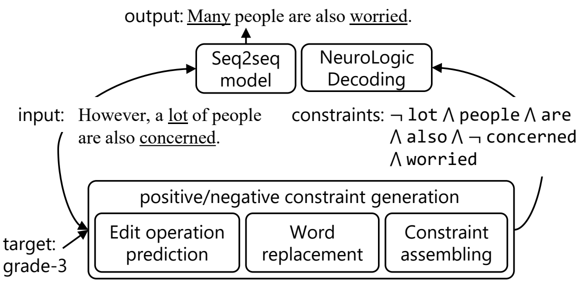
output: Many people are also worried.
Seq2seq model
NeuroLogic Decoding
input: However, a lot of people are also concerned.
constraints: ¬ lot ∧ people ∧ are ∧ also ∧ ¬ concerned ∧ worried
positive/negative constraint generation
Edit operation prediction
Word replacement
Constraint assembling
target: grade-3

Figure 1: Overview of the proposed method

2 Proposed Method

Figure 1 illustrates an overview of the proposed method, in which generated constraints are applied to a seq2seq model via NeuroLogic decoding.

2.1 Word Level Lexicon

We create word level lexicons to generate constraints sensible to a target level. We assign word levels based on their frequency in sentences of a certain level, assuming that higher-level words would frequently appear in higher-level sentences. The frequency of a word w in sentences of a level ℓ is as follows: $f(w, \ell) = \frac{n_\ell(w)}{\sum_{\hat{w} \in V_\ell} n_\ell(\hat{w})}$, where $n_\ell(w)$ denotes the number of occurrences of w in ℓ-level sentences, and V_ℓ denotes a set of unique words in those sentences. A word level k is determined as $k = \text{argmax}_\ell \, f(w, \ell)$. Finally, we collect all ℓ-level words as a lexicon D_ℓ for each level.

2.2 Constraint Generation

Constraints are generated in three steps. The proposed method first predicts all edit operations in an input conditioned on a target level. Following this, it identifies lexical paraphrases for replacing higher-level words. Finally, positive and negative constraints are assembled based on these edit operations, lexical paraphrases, and word level lexicons.

Edit Operation Prediction The proposed method uses a pre-trained language model to predict an edit operation among `replace`, `keep`, and `delete` for each word. These edit operations should depend on a target level. Therefore, the input sentence is tagged with a special token representing the target level, *e.g.*, "sentence <3>."

Manual annotation of these edit operations is costly. Thus, we synthesize a fine-tuning corpus using a state-of-the-art word alignment model (Lan et al., 2021). Specifically, we obtain word alignments between parallel sentences in a simplification corpus. Words with null-alignments are assigned `delete` labels. Among the aligned words, words aligned with identical counterparts are assigned `keep` labels, and the ones aligned to words with different surfaces are assigned `replace` labels. This pseudo-labelled corpus is used for fine-tuning.

target level: ℓ	`replace`	`keep`	`delete`
word level $\leq \ell$	—	P	—
word level $> \ell$	N, P	—	N

Table 1: Assembling positive (P) and negative (N) constraints relevant to controlling output levels

Replacement Word Identification The proposed method identifies a word $\hat{w}$ that should replace another word w whose predicted label is `replace`. Given the target level ℓ of simplification, it computes the semantic similarity between w and words in $\{D_k | k \leq \ell\}$ and identifies the replacement word $\hat{w}$ as the one with the highest similarity. For similarity estimation, we fine-tune a pre-trained language model.

Constraint Assembling Finally, we generate positive and negative constraints based on the predicted edit operations and the replacement words. We focus on the edit operations that are relevant to controlling output levels. Note that the predicted edit operations should be a mixture of various edits, including general lexical paraphrasing and omissions. Therefore, we use the word level lexicons to select operations relevant to controlling output levels as summarized in Table 1.

Specifically, words with the `delete` label transform into negative constraints if their levels are higher than the target level ℓ. Words with `keep` labels transform into positive constraints if their levels are lower than or equal to ℓ. Finally, words with `replace` labels transform into negative constraints and their replacement words transform into positive constraints if their levels are higher than ℓ.

The cases where the edit operations and the word level lexicons conflict, *i.e.*, words whose levels are lower than or equal to ℓ but predicted `replace` and `delete` operations, are expected to be independent for controlling output levels and correspond to general lexical paraphrasing and omissions. Therefore, we exclude these operations from the constraints and rely on the seq2seq model for their handling.

3 Experiments

3.1 Dataset

To evaluate the proposed method on the controllable text simplification task, we used Newsela and Newsela-Auto, which provide pairs of complex and simple sentences with K-12 grade levels. These are the only corpora providing fine-grained levels, which makes them standard datasets for evaluating controllable text simplification models. While Newsela-Auto preserves higher quality sentence alignments, we also experimented on Newsela for comprehensive comparison to previous studies. For Newsela, we used the data-split by Zhang and Lapata (2017) consisting of $94,208$ training, $1,129$ validation, and $1,077$ test sentences. For Newsela-Auto, we used the official split of $394,300$ training, $43,317$ validation, and $44,067$ test sentences.

3.2 Implementation Details

We implemented the proposed method using Pytorch[1] and Transformers (Wolf et al., 2020)[2]. All experiments were conducted on an NVIDIA A6000 GPU with a 48 GB memory. Appendix A presents details regarding the fine-tuning settings.

Edit Operation Prediction Model We fine-tuned pre-trained BERT (Devlin et al., 2019) models for an edit operation prediction using the pseudo-labelled corpora created using Newsela and Newsela-Auto, respectively. Table 2 depicts the precision, recall, and F1 of the operation prediction on the test sets. The results indicate that `replace` operations are difficult to predict owing to their infrequency; however, the results confirm that the proposed method improves text simplification even though the edit operation prediction is imperfect.

	Newsela			Newsela-Auto		
Edit Operation	P	R	F1	P	R	F1
`replace`	0.28	0.21	0.24	0.28	0.15	0.19
`keep`	0.58	0.57	0.57	0.58	0.57	0.58
`delete`	0.70	0.73	0.72	0.73	0.77	0.75

Table 2: Performance of edit operation prediction on the test sets of Newsela and Newsela-Auto

Lexical Similarity Estimation Model We fine-tuned a pre-trained RoBERTa (Liu et al., 2019) for a lexical similarity estimation using a corpus that provides human assessment of semantic similarities for 26.5k word pairs on a 5-point scale (Pavlick et al., 2015)[3]. Specifically, we concatenate a pair of words w and $\hat{w}$ with start and separator symbols as "<s>w</s></s>$\hat{w}$</s>" and input it in the model. The hidden output of the <s> symbol is then input into a linear layer to predict the similarity. Finally, we obtain a symmetric similarity score based on $(\text{sim}(w, \hat{w}) + \text{sim}(\hat{w}, w))/2$. We randomly split the corpus into 72% for training, 8% for validation, and 20% for testing. The fine-tuned model achieved a sufficiently high Pearson correlation coefficient of 0.86 on the test set. For a comparison, the correlation coefficient of cosine similarities computed using FastText (Bojanowski et al., 2017) was found to be 0.50.

Seq2seq Model As a seq2seq model to employ NeuroLogic decoding (Lu et al., 2021), we fine-tuned two pre-trained BART-Base (Lewis et al., 2020) models separately for Newsela and Newsela-Auto corpora. The batch size was 64, and the optimizer used was Adam (Kingma and Ba, 2015) with a learning rate of $1e-5$. The fine-tuning continued for 20 epochs, and a checkpoint with the highest SARI (Xu et al., 2016) score on the validation set was used as the final model.

3.3 Comparison

The proposed method is the hybrid of translation-based and edit-based approaches, hence, we compare it with existing methods in these categories. As translation-based methods, We compare our method to DRESS (Zhang and Lapata, 2017), which uses reinforcement learning for maximizing SARI score, as a conventional method. We also compare to MUSS (Martin et al., 2022), which also uses the pre-trained BART and hods the state-of-the-art measured on the Newsela corpus. From strong edit-based methods, we compare the proposed method to EditNTS (Dong et al., 2019) that explicitly learns edit operations using a neural programmer-interpreter model and the model proposed by Kumar et al. (2020) that conducts iterative edits of input sentences. As existing hybrid methods, we compare our method to the models proposed by Kajiwara (2019)[4] and Dehghan et al. (2022), both of which employ negative con-

[1] https://pytorch.org/

[2] https://huggingface.co/docs/transformers/

[3] http://www.seas.upenn.edu/~nlp/resources/ppdb-2.0-human-labels.tgz

[4] For a fair comparison, we employed a fine-tuned BART in (Kajiwara, 2019), which resulted in a higher SARI score.

Model	SARI	Add	Keep	Delete	FKGL	PCC	MSE	ACC	Len
Source	12.24	0.00	36.72	0.00	9.18	0.338	47.2	15.5	23.06
Reference	100.0	100.0	100.0	100.0	3.96	1.000	0.0	100.0	12.75
DRESS (Zhang and Lapata, 2017)†	38.03	2.43	42.20	69.47	4.97	0.388	13.0	24.3	14.37
MUSS (Martin et al., 2022)†	41.20	**6.02**	35.88	**81.70**	2.43	0.362	13.3	20.9	9.23
BART	38.54	3.64	40.59	71.40	4.63	0.350	13.6	26.2	11.26
EditNTS (Dong et al., 2019)⋆	37.05	1.23	36.55	73.37	**3.82**	0.266	16.1	21.4	13.25
(Kumar et al., 2020)†	38.37	1.01	36.51	77.58	2.95	0.334	12.6	25.5	9.61
(Kajiwara, 2019)	38.48 ⋆	4.55	**43.41**	67.47	5.01	0.417	12.2	**28.1**	14.27
(Dehghan et al., 2022)‡	40.01	3.06	36.53	80.43	3.20	–	–	–	11.72
Proposed	**42.65**	4.55	42.49	80.90	3.74	**0.420**	**11.1**	27.9	**12.01**
Proposed (Oracle)	54.73	10.98	66.07	87.14	4.07	0.591	8.0	37.3	12.47

Table 3: Results on the Newsela test set: † indicates that a score was recomputed with EASSE using outputs shared by the authors, ⋆ indicates that a model was trained in this study using the released implementation, and ‡ presents that a score was borrowed from the original papers with the same settings as this experiment.

Model	SARI	Add	Keep	Delete	FKGL	PCC	MSE	ACC	Len
Source	12.04	0.00	36.12	0.00	10.11	**0.393**	57.7	13.9	24.82
Reference	100.0	100.0	100.0	100.0	4.34	1.000	0.0	100.0	13.34
BART	39.66	4.16	39.17	75.65	**4.38**	0.342	16.4	**26.9**	10.33
EditNTS (Dong et al., 2019)	37.43	0.97	34.78	76.53	3.12	0.215	20.4	23.2	11.24
(Kajiwara, 2019)	38.30	**4.42**	40.51	69.96	5.03	0.371	16.0	26.8	**13.79**
Proposed	**43.09**	4.41	**42.74**	**82.13**	3.89	0.391	**15.1**	26.8	11.85
Proposed (Oracle)	51.75	7.45	61.14	86.66	4.64	0.611	9.9	34.5	12.90

Table 4: Results on the Newsela-Auto test set, where all models were trained and evaluated in this study.

straints.[5] In contrast, our method employs both positive and negative constraints on a translation-based model.

3.4 Evaluation Metrics

Following previous studies, we measured the SARI (with F1 scores of `Add`, `Keep`, and `Delete` operations) and FKGL using EASSE (Alva-Manchego et al., 2019), as well as the average output lengths (Len). Note that the FKGL and Len should be closer to those of references. Furthermore, to evaluate simplification controllablity, we measured Pearson's correlation coefficient (PCC), Mean Squared Error (MSE), and Accuracy (ACC) between FKGL scores of outputs and references (Agrawal et al., 2021). The Accuracy represents the percentage of outputs whose grades are within 1-grade difference from those of references.

Src	The rest would be preserved as open space.
Ref	The rest would be saved as open space.
BART	The rest would be preserved as open space.
Prop.	The rest would be kept as open space.
- PC	rest, be, kept, open, space
- NC	preserved

Table 5: Example outputs: "PC" and "NC" represent positive and negative constraints, respectively.

3.5 Results

The experimental results on the test sets of Newsela and Newsela-Auto are presented in Tables 3 and 4, respectively. The tables present the performance of representative translation-based (the second set of rows), edit-based (the third set of rows), and hybrid (the last set of rows) methods. The tables also present the performance of source and reference sentences (the first set of rows). "BART"

[5] Due to the heavy dependence on Google Translate to prepare a training corpus, we could not replicate (Agrawal et al., 2021) in this study.

Src	So Yan, a widow since her husband's death nearly a decade ago, spends every weekday at a modest community center near her home, where she plays mahjong and eats meals prepared by a volunteer staff.
Prop. (Grade 8)	She spends every weekday at a community center near her home.
- PC	husband, at, community, near, staff
- NC	widow, a
Prop. (Grade 5)	Yan's husband died almost 10 years ago.
- PC	–
- NC	widow, nearly, a, every, community, center, and, meals, prepared, by, staff
Prop. (Grade 2)	Yan is a widow.
- PC	–
- NC	widow, husband, nearly, a, ago, at, community, center, near, and, meals, prepared, by, staff

Table 6: Example outputs of controllable simplification; an input sentence of grade-12 was simplified to the grade-8, 5, and 2, respectively.

corresponds to the fine-tuned BART in this study, and "Proposed" represents the proposed method applying our lexical constraint on "BART."

The proposed method achieved the highest SARI and MSE scores with the highest and second-highest PCC scores on Newsela and Newsela-auto, respectively.[6] Furthermore, its output lengths are closest and second-closest to those of the references. A comparison with hybrid methods indicates the effectiveness of the flexible constraints of the proposed method, in spite of the imperfect nature of the edit operation prediction, as shown in Table 2. Among the previous methods, MUSS presents the highest SARI score, which fine-tunes BART using a large-scale data augmentation. The proposed method outperforms it using only the Newsela training set. Finally, a comparison of the `Add`, `Keep`, and `Delete` scores against BART confirms that our lexical constraint successfully improves all of these operations.

Oracle Performance The last rows in Tables 3 and 4 show the proposed method with oracle lexical constraints created using reference sentences as described in Section 2.2. The significantly higher SARI scores indicate that the proposed method can be further enhanced by improved constraint generation, in particular, by more precise edit operation prediction.

Example Outputs Table 5 presents example outputs where the input of grade-5 was simplified to grade-3. The proposed method successfully replaced *preserved* with *kept* owing to the lexical constraints. By contrast, BART ended up preserving it in the output. Table 6 shows example outputs where the input of grade-12 was simplified to the grade-8, 5 and 2, respectively. These outputs indicate that the proposed method can adjust sentence structures while considering lexical complexities according to the target levels.

4 Conclusion

We proposed a hybrid method for controllable text simplification that takes both advantages of translation- and edit-based methods using the flexible lexically constrained decoding. The experimental results showed that the proposed method conducts high-quality controllable text simplification on Newsela and Newsela-Auto. We expect that the proposed method also works for text simplification in general, *i.e.*, the binary transformation from complex to simple sentences. This investigation is left for our future work. We will also explore complex combinations of constraints allowed by NeuroLogic decoding in the future.

Acknowledgements

We sincerely thank Masato Yoshinaka for his contribution to the lexical similarity estimation model. We appreciate the anonymous reviewers for their insightful comments and suggestions to improve the paper. This work was supported by MHLW AC Program Grant Number JPMW21AC5001.

[6] The highest PCC score of source in Newsela is due to the positive correlation between grades of the source and reference sentences.

References

Sweta Agrawal, Weijia Xu, and Marine Carpuat. 2021. A non-autoregressive edit-based approach to controllable text simplification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3757–3769.

David Allen. 2009. A study of the role of relative clauses in the simplification of news texts for learners of english. *System*, 37(4):585–599.

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*, pages 295–305.

Fernando Alva-Manchego, Louis Martin, Carolina Scarton, and Lucia Specia. 2019. EASSE: Easier automatic sentence simplification evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association of Computational Linguistics (TACL)*, 5:135–146.

Mohammad Dehghan, Dhruv Kumar, and Lukasz Golab. 2022. GRS: Combining generation and revision in unsupervised sentence simplification. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 949–960.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4171–4186.

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 Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3393–3402.

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 Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7943–7960.

Tomoyuki Kajiwara. 2019. Negative lexically constrained decoding for paraphrase generation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6047–6052.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Reno Kriz, João Sedoc, Marianna Apidianaki, Carolina Zheng, Gaurav Kumar, Eleni Miltsakaki, and Chris Callison-Burch. 2019. Complexity-weighted loss and diverse reranking for sentence simplification. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 3137–3147.

Dhruv Kumar, Lili Mou, Lukasz Golab, and Olga Vechtomova. 2020. Iterative edit-based unsupervised sentence simplification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7918–7928.

Wuwei Lan, Chao Jiang, and Wei Xu. 2021. Neural semi-Markov CRF for monolingual word alignment. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6815–6828.

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 Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7871–7880.

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. *arXiv*, 1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Ximing Lu, Peter West, Rowan Zellers, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Neuro-Logic decoding: (un)supervised neural text generation with predicate logic constraints. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4288–4299.

Jonathan Mallinson, Aliaksei Severyn, Eric Malmi, and Guillermo Garrido. 2020. FELIX: Flexible Text Editing Through Tagging and Insertion. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1244–1255.

Louis Martin, Angela Fan, Éric de la Clergerie, Antoine Bordes, and Benoît Sagot. 2022. MUSS: Multilingual unsupervised sentence simplification by mining paraphrases. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, pages 1651–1664.

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

Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P. Dinu. 2017. Exploring neural text simplification models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 85–91.

Kostiantyn Omelianchuk, Vipul Raheja, and Oleksandr Skurzhanskyi. 2021. Text Simplification by Tagging. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 11–25.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the Joint Conference of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 425–430.

Carolina Scarton and Lucia Specia. 2018. Learning simplifications for specific target audiences. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 712–718.

Matthew Shardlow. 2014. A survey of automated text simplification. *International Journal of Advanced Computer Science and Applications (IJACSA), Special Issue on Natural Language Processing*, 4(1).

Sai Surya, Abhijit Mishra, Anirban Laha, Parag Jain, and Karthik Sankaranarayanan. 2019. Unsupervised neural text simplification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2058–2068.

Willian Massami Watanabe, Arnaldo Candido Junior, Vinícius Rodriguez Uzêda, Renata Pontin de Mattos Fortes, Thiago Alexandre Salgueiro Pardo, and Sandra Maria Aluísio. 2009. Facilita: Reading assistance for low-literacy readers. In *Proceedings of the ACM international conference on Design of communication*, pages 29–36.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 38–45.

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.

Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 584–594.

Sanqiang Zhao, Rui Meng, Daqing He, Andi Saptono, and Bambang Parmanto. 2018. Integrating transformer and paraphrase rules for sentence simplification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3164–3173.

A Details Regarding the Fine-Tuning Settings

Edit Operation Prediction Model We fine-tuned the pre-trained BERT-Base, uncased model for edit operation prediction. The batch size was 40, and the optimizer used was AdamW (Loshchilov and Hutter, 2019) with a learning rate of $1e-5$ with linear decay according to steps. We applied early stopping with the patience of 3 epochs to maximize the F1 score on the validation set.

Lexical Similarity Estimation Model We fine-tuned RoBERTa-Large for lexical similarity estimation. The batch size was 256, and the optimizer used was AdamW with a learning rate of $2e-5$ with linear decay according to steps. The training was terminated early with the patience of 7 epochs to minimize the mean squared error in the validation set.