Text Simplification with Reinforcement Learning using Supervised Rewards on Grammaticality, Meaning Preservation, and Simplicity

Akifumi Nakamachi[†], **Tomoyuki Kajiwara**[‡], **Yuki Arase**[†]
[†]Graduate School of Information Science and Technology, Osaka University
[‡]Institute for Datability Science, Osaka University
[†]{nakamachi.akifumi, arase}@ist.osaka-u.ac.jp
[‡]kajiwara@ids.osaka-u.ac.jp

Abstract

We optimize rewards of reinforcement learning in text simplification using metrics that are highly correlated with human-perspectives. To address problems of exposure bias and loss-evaluation mismatch, text-to-text generation tasks employ reinforcement learning that rewards task-specific metrics. Previous studies in text simplification employ the weighted sum of sub-rewards from three perspectives: grammaticality, meaning preservation, and simplicity. However, the previous rewards do not align with human-perspectives for these perspectives. In this study, we propose to use BERT regressors fine-tuned for grammaticality, meaning preservation, and simplicity as reward estimators to achieve text simplification conforming to human-perspectives. Experimental results show that reinforcement learning with our rewards balances meaning preservation and simplicity. Additionally, human evaluation confirmed that simplified texts by our method are preferred by humans compared to previous studies.

1 Introduction

Text simplification is one of the text-to-text generation tasks that rewrites complex sentences into simpler ones. Text simplification is useful for preprocessing of NLP tasks such as semantic role labeling (Vickrey and Koller, 2008; Woodsend and Lapata, 2014) and machine translation (Štajner and Popović, 2016, 2018). It also has valuable applications such as assisting language learning (Inui et al., 2003; Petersen and Ostendorf, 2007) and helping language-impaired readers (Carroll et al., 1999).

There are two problems in text-to-text generation with an encoder-decoder model: exposure bias and loss-evaluation mismatch (Ranzato et al., 2016; Wiseman and Rush, 2016). The former is that the model is not exposed to its own errors during training. The latter is that while the generated sentence is evaluated as a whole sentence during inference, it is evaluated at the token-level during training. To address these problems, reinforcement learning has been employed in text-to-text generation tasks, such as machine translation (Ranzato et al., 2016) and abstractive summarization (Paulus et al., 2018). These studies use metrics suitable for each task, such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004), as rewards. Although reinforcement learning based text simplification models (Zhang and Lapata, 2017; Zhao et al., 2020) have used rewards metrics such as SARI (Xu et al., 2016) and FKGL (Kincaid et al., 1975), these metrics do not align with human-perspectives, *i.e.*, human evaluation results (Xu et al., 2016; Sulem et al., 2018; Alva-Manchego et al., 2020).

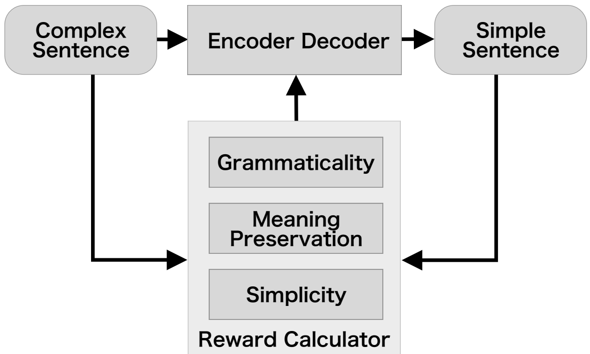

Figure 1: Overview of the reinforcement learning for text simplification.

In this study, we train a text simplification model based on reinforcement learning with rewards that highly agree with human-perspectives. Specifically, we apply a BERT regressor (Devlin et al., 2019) on grammaticality, meaning preservation, and simplicity, respectively, as shown in Figure 1. Experiments on the Newsela dataset (Xu et al., 2015) have shown that reinforcement learning with our rewards balances meaning preservation and simplicity. Further, manual evaluation has shown that our outputs were preferred by humans compared to previous models.

2 Background: Reinforcement Learning for Text Simplification

Reinforcement learning in text-to-text generation tasks is performed as additional training for pre-trained text-to-text generation models. It is a common technique to linearly interpolate a reward of reinforcement learning and the cross-entropy loss to avoid misleading training because of a large action space (Ranzato et al., 2016; Zhang and Lapata, 2017). We first explain an attention based encoder-decoder model (EncDecA) (Luong et al., 2015) in Section 2.1 and then reinforcement learning for text simplification in Section 2.2.

2.1 Encoder-Decoder Model with Attention

Let $X = (x_1, \cdots, x_{|X|})$ be a source sentence and $Y = (y_1, \cdots, y_{|Y|})$ be its reference sentence. In text simplification, the source and reference are complex and simple sentences, respectively. An encoder takes a source sentence as input, and outputs hidden states. Decoder generates a word distribution at time step $t + 1$ from all the encoder hidden states and the series of decoder hidden state $(h_1, \cdots, h_t)$. We generate a sentence $\hat{Y}$ by sampling words from the distribution at each time step.

The objective function for training is averaged cross entropy loss of sentence pairs:

$$\mathcal{L}_{\mathrm{C}} = -\sum_{t=1}^{|Y|} \log P(y_{t+1}|y_{1\cdots t}, X). \tag{1}$$

As the Equation (1) suggests, $y_{1\cdots t}$ is given at training but not at an inference (exposure bias situation). In addition, cross entropy loss cannot be evaluated at a sentence-level (loss-evaluation mismatch).

2.2 Reinforcement Learning

Similar to other text-to-text generation tasks (Ranzato et al., 2016; Paulus et al., 2018), reinforcement learning is applied for text simplification (Zhang and Lapata, 2017; Zhao et al., 2020) to address the problems of exposure bias and loss-evaluation mismatch. In the reinforcement learning step, the pre-trained text-to-text generation model is trained to increase the reward $R(\cdot)$. By employing a reward function that takes the entire sentence $\hat{Y}$ into account, the exposure bias and loss-evaluation mismatch problems are mitigated.

As automatic evaluation metrics for text simplification, BLEU, SARI, and FKGL have been used; however, there has not been a consensus of standard metrics because of their low correlation with human perspectives (Xu et al., 2016; Sulem et al., 2018; Alva-Manchego et al., 2020). Therefore, the previous studies designed rewards from the following three perspectives, based on the standards in manual evaluation for text simplification.

- Grammaticality: This reward assesses the grammatical acceptability of the generated sentence $\hat{Y}$. Previous studies used an neural language model implemented using Long short-term memory (Mikolov et al., 2010; Hochreiter and Schmidhuber, 1997).
- Meaning Preservation: This reward assesses the semantic similarity between the source sentence X and the generated sentence $\hat{Y}$. Zhang and Lapata (2017) used cosine similarity of the sentence representations from a sequence auto-encoder (Dai and Le, 2015). Zhao et al. (2020) used cosine similarity of sentence representations which consists of weighted average of word embeddings (Arora et al., 2017).
- Simplicity: This reward assesses the simplicity of the generated sentence $\hat{Y}$. Zhang and Lapata (2017) used SARI$(X, Y, \hat{Y})$ score, while Zhao et al. (2020) used FKGL$(\hat{Y})$ score.

Among different ways to conduct reinforcement learning, one of the standard approaches used in text simplification is directly maximizing the rewards by the REINFORCE algorithm (Williams, 1992; Ranzato et al., 2016). This approach optimizes the log probability weighted by the expected future reward as the objective function:

$$\mathcal{L}_{\mathrm{R}} = -\sum_{t=1}^{|Y|} r(h_t) \log P(y_{t+1}|y_{1\cdots t}, X), \tag{2}$$

where the expected future reward $r(h_t)$ is estimated using a reward estimator $R(\cdot)$ and a baseline estimator $b(h_t)$ calculated from the hidden state at time step t.

$$r(h_t) = R(\cdot) - b(h_t). \tag{3}$$

Following (Ranzato et al., 2016), the baseline estimator is optimised by minimizing $\|b_t - R(\cdot)\|^2$.

Hashimoto and Tsuruoka (2019) discussed problems in text-to-text generation by reinforcement learning; the expected future reward estimation is unstable due to the huge action space, which hinders convergence. This is because the action space of text-to-text generation corresponds to the entire target vocabulary, where many words are rarely used for prediction. Therefore, previous studies (Wu et al., 2018; Paulus et al., 2018; Hashimoto and Tsuruoka, 2019) proposed to stabilize the training in reinforcement learning by first pre-training a model with cross-entropy loss, and then adding weighted REINFORCE loss:

$$\mathcal{L} = \lambda\mathcal{L}_R + (1-\lambda)\mathcal{L}_C. \quad (4)$$

3 BERT-based Supervised Reward

We propose a reward estimator $R(X, \hat{Y})$ consisting of sub-rewards for grammaticality R_{G}, meaning preservation R_{M}, and simplicity R_{S}. These sub-rewards are combined by weighted sum with hyper parameters of δ and ϵ:

$$\begin{aligned} R(X, \hat{Y}) = \delta R_{\mathrm{G}}(\hat{Y}) + \epsilon R_{\mathrm{M}}(X, \hat{Y}) \\ + (1-\delta-\epsilon) R_{\mathrm{S}}(\hat{Y}). \end{aligned} \quad (5)$$

To achieve a better correlation between each sub-reward and human perspectives, we employ BERT regressors and fine-tune them using manually annotated datasets.

3.1 Implementation Details

For each sub-reward model, we fine-tuned a pre-trained bert-base-uncased model[1] from Hugging Face Transformers library (Wolf et al., 2019). Dropout of 0.2 was applied to all embedding and hidden layers. All the models were optimized using the Adam optimizer (Kingma and Ba, 2015). We linearly decrease learning rate with a warm up in the first $1,000$ training steps. The batch size was 32 sentences. We created a checkpoint for the model at every 100 steps. The training stopped after 10 epochs without improvement in the Pearson correlation measured on the validation set.

Each sub-reward estimator was fine-tuned using the following datasets. Table 1 shows the statistics for each dataset.

Grammaticality We use the GUG dataset[2] (Heilman et al., 2014) for estimating the grammaticality

[1] https://huggingface.co/bert-base-uncased

[2] https://github.com/EducationalTestingService/gug-data

	Train	Validation	Test
GUG	$1,518$	747	754
STS-B	$5,749$	$1,500$	$1,379$
Newsela	$94,208$	$1,129$	$1,077$

Table 1: The numbers of sentences in datasets for each sub-reward estimator

of a sentence. The GUG dataset consists of sentences written by English as the second language learners. Each sentence has four native English speakers assessing grammatical acceptability on a scale of 1 to 4. We estimate the average of these ratings.

Meaning Preservation We use the STS-B dataset[3] (Cer et al., 2017) for estimating the meaning preservation of sentence pairs. The STS-B dataset consists of sentence pairs from multiple sources such as news headlines and image captions. Each sentence pair is evaluated for semantic similarity by five cloud workers on a scale of 0 to 5. We estimate the average of these ratings.

Simplicity We use the Newsela dataset[4] (Xu et al., 2015) for estimating the simplicity of a sentence. The Newsela dataset is a parallel dataset of complex and simple sentences. Each sentence is assigned a U.S. elementary school reading level on a scale of 2 to 12. We follow the data split by Zhang and Lapata (2017). We estimate the grade level of a single sentence using the BERT regressor.

3.2 Intrinsic Evaluation of Rewards

We evaluated how well our sub-reward estimators correlate with human perspectives, compared to previous studies (Zhang and Lapata, 2017; Zhao et al., 2020).

Compared Models We reimplemented the sub-reward estimators introduced in Section 2.2. For R_{G} estimator, we used a 2-layer LSTM language model of 256 hidden dimensions and word embeddings of 300 dimensions. For R_{M} estimator in Zhang and Lapata (2017), we implemented a sequence auto-encoder with bidirectional LSTMs as an encoder. For R_{M} estimator in Zhao et al. (2020), we used 300-dimensional word2vec embeddings[5]

[3] http://ixa2.si.ehu.es/stswiki/index.php/STSbenchmark

[4] https://newsela.com/data/

[5] https://code.google.com/archive/p/word2vec/

(Mikolov et al., 2013).

Datasets For grammaticality and meaning preservation, we used the test sets of GUG and STS-B to evaluate the Pearson correlation with human evaluations. For simplicity, a reward estimator should be sensitive to grade levels of sentences with the same meaning, because text simplification intends to preserve the original meaning of the input sentence. Therefore, we extracted pairs of (Y_1, Y_2) of the same source sentence from the Newsela dataset and evaluated the Pearson correlation between the difference of estimated simplicity and the difference of gold-standard grade levels. Note that R_{S} in Zhang and Lapata (2017), *i.e.*, SARI, requires source and reference sentences. Hence, we regarded the simplest sentence of the same source as the reference. We extracted 323 sentence pairs of simplified versions of the same sentence from the Newsela test set.

Results Table 2 shows the evaluation results of each sub-reward estimator. In all perspectives, existing unsupervised sub-reward estimators have little or no correlation with human annotations. As expected, fine-tuning BERT for each task significantly improved the Pearson correlations.

	G	M	S
Zhang's sub-rewards	-0.135	0.041	0.034
Zhao's sub-rewards	-0.135	0.379	0.175
Our sub-rewards	$\mathbf{0.726}$	$\mathbf{0.846}$	$\mathbf{0.473}$

Table 2: Pearson correlation of each sub-reward estimator. Note that G, M, S correspond to grammaticality, meaning preservation, and simplicity, respectively.

4 End-to-End Evaluation on Text Simplification

In this section, we evaluate our rewards on an end-to-end text simplification task using the Newsela dataset[6] shown in Table 1.

4.1 Baseline Encoder-Decoder Model

We implemented and pre-trained the EncDecA model as a common base to add reinforcement learning with rewards of ours and previous studies (Zhang and Lapata, 2017; Zhao et al., 2020). The EncDecA model has a 2-layer LSTM of 256 hidden dimensions for both the encoder and decoder, and attention mechanism by multi-layer perceptron with a layer size of 256. It has word embedding layers of 300 dimensions tying the source, target, and the output layer's weight matrices. Dropout of 0.2 was applied to all embeddings and hidden layers. We used byte-pair encoding[7] (Sennrich et al., 2016) to limit the vocabulary size to $20,000$ in addition to the pre-processing by Zhang and Lapata (2017).

The EncDecA model was pre-trained by cross entropy loss with Adam optimizer ahead of reinforcement learning. The batch size was 32 sentences. We created a checkpoint for the model at every 100 steps. In the pre-training, training was stopped after 10 epochs without improvement of SARI score measured on the validation set. However, as the SARI is not stable at the beginning of the training, we ignored checkpoints whose BLEU scores measured on the validation set were less than 21, as suggested by Vu et al. (2018).

4.2 Hyper-Parameter Settings

In reinforcement learning, the hyperparameter λ in Equation (4) was initialized to 0.1, and linearly increased for each iteration until 0.9 during first 10 epochs for stabilizing training process. Following Zhang and Lapata (2017), we trained reinforcement learning models with stochastic gradient descent optimizer with a learning rate of 0.001 and a momentum term of 0.9. Additionally, we trained the baseline estimator with Adam optimizer with a learning rate of 0.001. In the reinforcement learning, training was stopped after 10 epochs without improvement on rewards measured on the validation set.

We set the equal weights to our sub-rewards, *i.e.*, assigned $1/3$ to δ and ϵ in Equation (5), respectively. Tuning of these weights is our future work.

4.3 Results of Automatic Evaluation

The performance of each method is automatically evaluated using the EASSE toolkit[6] by BLEU and SARI. Furthermore, we perform detailed automatic evaluations of grammaticality, meaning preservation, simplicity, and overall quality defined in Equation (5) using our sub-reward estimators.

Table 3 shows the experimental results. In both our reward and existing rewards, reinforcement learning has improved the EncDecA baseline in

[6]We did not experiment with the Simple English Wikipedia because it does not have a detailed, difficulty-by-difficulty rewrite.

[7]https://github.com/google/sentencepiece

	Metrics		Rewards				Human
	BLEU	SARI	G	M	S	O	Avg. Rank
Reference	100.0	100.0	0.909	0.585	0.708	0.734	–
EncDecA	21.57	37.64	0.862	**0.681**	0.648	0.730	n/a
RL w/ Zhang's Reward	23.30	**39.24**	**0.878**	0.659	**0.663**	0.734	1.91
RL w/ Zhao's Reward	**23.42**	39.20	**0.878**	0.662	0.662	0.734	1.69
RL w/ Our Reward	23.14	38.70	**0.878**	0.678	0.653	**0.736**	**1.45****

Table 3: Experimental results of text simplification. Note that G, M, S, and O correspond to grammaticality, meaning preservation, simplicity, and overall rewards, respectively. ** indicates a statistically significant difference between the others. (The p-value of the unpaired t-test of our method and both of the other methods were $p < 0.01$.)

Source	They are tired and it shows in their voices , but they 're still on the freedom highway .
Reference	Their voices sound tired .
EncDecA	They are tired and it shows in their voices , but they 're still on the freedom .
RL w/ Zhang's Reward	They are tired .
RL w/ Zhao's Reward	They are tired .
RL w/ Our Reward	They are tired and it shows in their voices .
Source	Historic architecture , crafts and music are being overwhelmed by China 's growth and its inability to effectively preserve traditions of the past .
Reference	Kite making is only part of a bigger story in China .
EncDecA	Historic architecture , crafts and music are being overwhelmed by China 's growth and its inability to effectively preserve traditions of the past .
RL w/ Zhang's Reward	Historic architecture , crafts and music are being overwhelmed by China 's growth .
RL w/ Zhao's Reward	The music of the city 's growth and music are being overwhelmed by China 's growth .
RL w/ Our Reward	Historic architecture , crafts and music are being overwhelmed by China 's growth and its actions .

Table 4: Examples of generated sentences by each simplification model.

both BLEU and SARI metrics. Reinforcement learning also improved rewards, but the EncDecA baseline was the best for meaning preservation. A trade-off relationship was observed between the rewards of meaning preservation and simplicity. This is expected because the meanings of the input and generated sentences deviate as the model replaces and deletes tokens for simplicity. Reinforcement learning based on our rewards does not improve simplicity as much as previous methods, but it does not worsen meaning preservation. This balance has made our model achieve the highest overall reward.

Table 4 shows generated sentences by each model. While the previous methods generates extremely simple sentence at the expense of meaning preservation, our model generates sentences with reasonable balances between meaning preservation and simplicity.

4.4 Results of Human Evaluation

We also conducted human evaluation using Amazon Mechanical Turk.[8] Human evaluators rank three sentences generated by a model based on reinforcement learning with different rewards, taking into account the source sentence. Three sentences were ranked on the basis of whether they were rewritten in a simple manner while preserving as much of the meaning of the source sentence as pos-

[8] https://www.mturk.com/

sible. We randomly selected 100 sets of generated sentences, excluding examples that all models generated the same sentences.[9] To ensure the quality of the human evaluation, we employed five master workers for each example and used 85 examples with at least three of them had the same ranking order.

The average ranking of each model is shown in the last column of Table 3. Our model was ranked significantly higher than previous models as confirmed by bootstrap testing. These results confirm that our rewards allow to generate simplified sentences preferred by humans.

5 Conclusion

We trained a text simplification model based on reinforcement learning with rewards that are highly correlated with human-perspectives. Experimental results showed that existing rewards employing evaluation metrics tend to generate extremely simple sentence at the expense of meaning preservation. Nevertheless, our BERT-based rewards succeeded in balancing meaning preservation and simplicity. In addition, we confirmed that human evaluators prefer our simplified sentences to those generated by previous rewards.

In this study, we set the equal weights to our sub-rewards. We plan to investigate the better weight balance of sub-rewards in the future.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP20K19861.

References

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.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A Simple but Tough-to-Beat Baseline for Sentence Embeddings. In *Proceedings of the 5th International Conference on Learning Representations*.

John Carroll, Guido Minnen, Darren Pearce, Yvonne Canning, Siobhan Devlin, and John Tait. 1999. Simplifying Text for Language-Impaired Readers. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*, pages 269–270.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, pages 1–14.

Andrew M. Dai and Quoc V. Le. 2015. Semi-supervised Sequence Learning. In *Proceedings of the 28th Conference on Neural Information Processing Systems*, pages 3079–3087.

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 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Kazuma Hashimoto and Yoshimasa Tsuruoka. 2019. Accelerated Reinforcement Learning for Sentence Generation by Vocabulary Prediction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3115–3125.

Michael Heilman, Aoife Cahill, Nitin Madnani, Melissa Lopez, Matthew Mulholland, and Joel Tetreault. 2014. Predicting Grammaticality on an Ordinal Scale. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 174–180.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Kentaro Inui, Atsushi Fujita, Tetsuro Takahashi, Ryu Iida, and Tomoya Iwakura. 2003. Text Simplification for Reading Assistance: A Project Note. In *Proceedings of the 2nd International Workshop on Paraphrasing*, pages 9–16.

J. Peter Kincaid, Robert P. Fishburne Jr., 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. *Technical report, Defence Technical Information Center (DTIC) Document*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations*.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Proceedings of the Workshop on Text Summarization Branches Out*, pages 74–81.

[9]We included examples that two models output the same sentences.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the 1st International Conference on Learning Representations*.

Tomas Mikolov, Martin Karafiát, Lukás Burget, Jan Cernocký, and Sanjeev Khudanpur. 2010. Recurrent Neural Network Based Language Model. In *Proceedings of the 11th Annual Conference of the International Speech Communication Association*, pages 1045–1048.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A Deep Reinforced Model for Abstractive Summarization. In *Proceedings of the 6th International Conference on Learning Representations*.

Sarah E. Petersen and Mari Ostendorf. 2007. Text Simplification for Language Learners: A Corpus Analysis. In *Proceedings of the SLaTE Workshop on Speech and Language Technology in Education*, pages 69–72.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence Level Training with Recurrent Neural Networks. In *Proceedings of the 4th International Conference on Learning Representations*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725.

Sanja Štajner and Maja Popović. 2016. Can Text Simplification Help Machine Translation? In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 230–242.

Sanja Štajner and Maja Popović. 2018. Improving Machine Translation of English Relative Clauses with Automatic Text Simplification. In *Proceedings of the 1st Workshop on Automatic Text Adaptation*, pages 39–48.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018. BLEU is Not Suitable for the Evaluation of Text Simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 738–744.

David Vickrey and Daphne Koller. 2008. Sentence Simplification for Semantic Role Labeling. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 344–352.

Tu Vu, Baotian Hu, Tsendsuren Munkhdalai, and Hong Yu. 2018. Sentence Simplification with Memory-Augmented Neural Networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 79–85.

Ronald J. Williams. 1992. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning*, 8:229–256.

Sam Wiseman and Alexander M. Rush. 2016. Sequence-to-Sequence Learning as Beam-Search Optimization. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1296–1306.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. Rush. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv:1910.03771*.

Kristian Woodsend and Mirella Lapata. 2014. Text Rewriting Improves Semantic Role Labeling. *Journal of Artificial Intelligence Research*, 51:133–164.

Lijun Wu, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018. A Study of Reinforcement Learning for Neural Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3612–3621.

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 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594.

Yanbin Zhao, Lu Chen, Zhi Chen, and Kai Yu. 2020. Semi-Supervised Text Simplification with Back-Translation and Asymmetric Denoising Autoencoders. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pages 9668–9675.