Extremely Low Resource Text simplification with Pre-trained Transformer Language Model

Takumi Maruyama and Kazuhide Yamamoto
Nagaoka University of Technology
1603-1, Kamitomioka Nagaoka, Niigata 940-2188, JAPAN
{*maruyama, yamamoto*}*@jnlp.org*

***Abstract*—Recent text simplification approaches regard the task as a monolingual text-to-text generation inspired by machine translation. In particular, the transformer-based translation model outperform previous methods. Although machine translation approaches need a large-scale parallel corpus, parallel corpora for text simplification are very small compared to machine translation tasks. Therefore, we attempt a simple approach which fine-tunes the pre-trained language model for text simplification with a small parallel corpus. Specifically, we conduct experiments with the following two models: transformer-based encoder-decoder model and a language model that receives a joint input of original and simplified sentences, called TransformerLM. Thus, we show that TransformerLM, which is a simple text generation model, substantially outperforms a strong baseline. In addition, we show that fine-tuned TransformerLM with only 3,000 supervised examples can achieve performance comparable to a strong baseline trained by all supervised data.**

***Keywords*-text simplification; language modeling; transfer-learning;**

I. INTRODUCTION

Automatic text simplification is a task that reduces the complexity of vocabulary and expressions while preserving the meaning of the text. This technique can be used to make many text resources available for a wide range of readers including children, nonnative speakers, and the disabled. As a preprocessing step, simplification can improve the performance of natural language processing tasks including parsing [1], summarization [2], [3], semantic role labelling [4], information extraction [5], and machine translation [6], [7].

Over the years, the number of tourists in Japan have increased. Japan hosts around 28 million visitors per year[1]. In addition, there are approximately 2.32 million foreign residents in Japan[2], and this number is on the rise. According to a survey conducted by the National Institute for Japanese Language and Linguistics, the number of people who can understand Japanese is more than the number of people who can understand English [8]. Hence, a simplified text is one of the important ways for providing information to foreigners, and therefore, a practical text simplification system is desired.

Recent approaches regard the simplification process as monolingual text-to-text generation task like machine translation [9], [10], [11], [12], [13], [14], [15]. Simplification rewritings are trained automatically from examples of original-simplified sentence pairs. Neural-machine-translation-based approaches have greatly improved simplification performance compared to statistical-machine-translation-based models or lexical simplification models. These require a large-scale parallel corpus. However, parallel corpora for text simplification are very few and small compared to machine translation tasks. In Japanese, there is no simplified corpus corresponds to Simple English Wikipedia[3] [16], [17], [18].

We focus on pre-training as a way to address a low-resource issue. Language model pre-training [19], [20] has led to impressive results on various tasks such as text classification, question answering, and sequence labeling [21], [22], [23]. Particularly, Shleifer et al. [22] have achieved strikingly performance in spite of slightly small supervised examples.

In this paper, we attempt a simple approach which fine-tunes the pre-trained language model for text simplification using an only small parallel corpus. Specifically, we experiment with the following two models, (1) the transformer-based encoder-decoder model, (2) the language model that receives a joint input of original and simplified sentences, called TransformerLM.

II. RELATED WORKS

Back-translation has substantially improved performance in the machine translation task. It is a method that constructs a synthetic parallel corpus by translating a monolingual corpus of a target language to a source language[24], [25]. In text simplification, Qiang et al. [26] use synthetic parallel corpus generated by back-translating the Simple English Wikipedia according to the method of Sennrich et al.[24]. By adding this synthetic data to training data, even a simple machine translation model can outperform more complex models such as model using reinforcement learning. However, back-translation cannot be applied to text simplification if there is no monolingual simplified corpus.

On the other hand, Kauchak [27] has combined a language model trained with a small simplified corpus and one trained with a large original corpus. The combined model performs as well as a model trained with a large simplified corpus on perplexity and lexical simplification tasks. Motivated by this result, we attempt to improve text simplification model using a large original corpus

[1]https://www.jnto.go.jp/jpn/statistics/visitor_trends
[2]https://www.e-stat.go.jp
[3]https://dumps.wikimedia.org/simplewiki/

978-1-7281-5014-7/19/$31.00 ©2019 IEEE

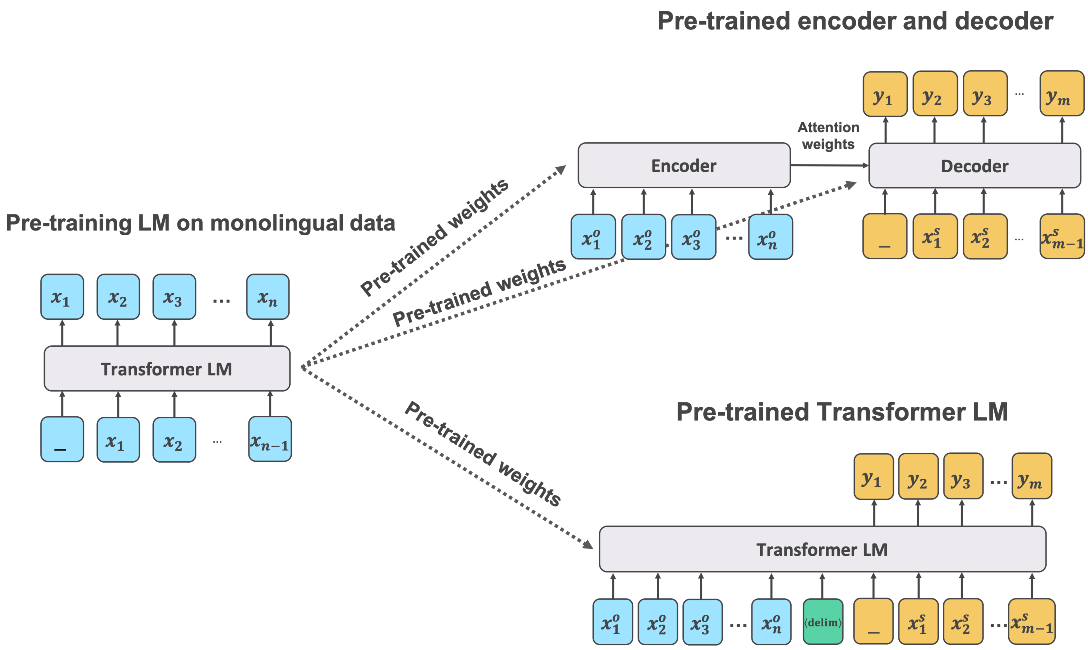

Figure 1. Two fine-tuned models, a transformer-based encoder-decoder model, and a language model that receives a joint input of original and simplified sentences.

instead of a large simplified corpus. Specifically, we train a language model using a large original corpus, and then, fine-tune it with a small parallel corpus for text simplification tasks.

III. Methods

As shown in Figure 1, we construct a text simplification model by fine-tuning a pre-trained language model. We conducted experiments in two ways: first, using a transformer-based encoder-decoder model; second, using a language model. In this section, we describe the pre-training method of a language model (section III-A). Then, we describe two methods for text generation from an encoder-decoder model (section III-B) and a language model (section III-C).

A. Language Model Pre-training

We use a language model based on transformer [28]. Instead of bidirectional models like ELMo [19] and BERT [20], we use unidirectional models such as GPT [29]. a sentence with N token (x_1, x_2, ..., x_N), our language model trains the parameter θ for maximizing the likelihood $p(x_1, x_2, ..., x_N; \theta)$.

$$p(x_1, x_2, ..., x_N; \theta) = \prod_{k=1}^{N} p(x_k | x_0, x_1, ..., x_{k-1}; \theta) \quad (1)$$

For pre-training, we use article extracted from Japanese Wikipedia [4] by *WikiExtractor* [5].

[4] https://dumps.wikimedia.org/jawiki/latest/jawiki-latest-pages-articles.xml.bz

[5] https://github.com/attardi/wikiextractor

B. Text Generation from Pre-trained Encoder-Decoder

We incorporate the weights of pre-training language model into standard encoder-decoder [30] models. The encoder-decoder model consists of a transformer encoder that reads the original sentences, a transformer decoder that generates the simplified sentences, and an attention mechanism [31] that allows the decoder to access encoder states during generation. Both encoder and decoder use the same structure. We compare three ways of incorporating the weights from a pre-trained language model, according to Ramachandran et al. [32]: (1) pre-training the encoder only, (2) pre-training the decoder only, and (3) pre-training both the encoder and decoder. In (3), the parameters of the encoder-decoder attention mechanism initialize randomly.

After the pre-trained weights incorporate into the encoder-decoder model, these are fine-tuned using a parallel corpus. This procedure often leads to catastrophic forgetting where the model's performance on language modeling tasks falls after fine-tuning [33], especially when trained on small supervised datasets. To avoid this problem, we add language modeling loss to translation loss in the fine-tuning step. The translation and language modeling losses are weighted equally.

Instead of a large-scale monolingual corpus, we conduct an experiment pre-training only using parallel corpus similar to Ramachandran et al. [32]. In pre-training using a parallel corpus, the encoder is initialized by a language model pre-trained on the original side, and the decoder is initialized by a language model pre-trained on the simplified side.

Table I
COMPARISON OF TEXT SIMPLIFICATION DATASETS

Datasets	Split Size			N-grams overlap in Simplified sentence [%]				Mean # words	
	Train	Validation	Test	unigrams	bigrams	trigrams	4-grams	Original	Simplified
Literal-translation	32,949	893	1,781	64.48	42.00	31.76	25.28	15.06	17.14
Free-translation	30,259	817	1,637	61.97	38.37	28.05	21.85	15.32	15.84

C. Text Generation from Pre-trained Language Model

We translate an original sentence to a simplified sentence using only a transformer decoder similar to Khandelwal[34] and Hoang[35]. Given the N tokens original sentence $X^o = (x_1^o, x_2^o, ..., x_N^o)$ and the M token simplified sentences $X^s = (x_1^s, x_2^s, ..., x_M^s)$, a transformer decoder receives the following input sequence, where, $\langle delim \rangle$ is a special token that means delimiter between an original sentence and a simplified sentence.

$$X = \left[X^o, \langle delim \rangle, X^s\right] \tag{2}$$

We use the same word embedding layer when the original sentence and the simplified sentence are vectorized. The positional embedding obtained from the following equations adds to word embeddings.

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{model}}) \tag{3}$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{model}}) \tag{4}$$

where pos indicates a position, i indicates the dimension, and d_{model} indicates embedding dimension. Note that when the delimitation token $\langle delim \rangle$ is reached, the position counter is reset. We add language modeling loss to translation loss in the fine-tuning step in the same way as the previous section III-B. The translation and language modeling losses are weighted equally.

IV. EXPERIMENTAL SETUP

A. Datasets

We experiment using two text simplification datasets contained *simplification corpus of local government announcement* as supervised data. When preprocessing, we excluded sentence pairs that are over 100 tokens on an original side or a simplified side. Some statistics of these datasets are shown in Table I.

The corpus is constructed by Moku et al[36]. One thousand one hundred official documents that are distributed in public facilities, such as a city office, hospital, and school, is simplified by 40 Japanese language teachers. This parallel corpus has three simplified versions; *literal-translation*, *free-translation*, *summary*. Each simplified level is defined as follow.

- **literal-translation**: The simplified version that rewrites difficult words or phrases into simple expressions;
- **free-translation**: The simplified version that rewrites a difficult sentence into a simplified sentence while preserving the meaning in the best possible manner;

Table III
COMPARISON OF EACH SYSTEM

Model	Literal-translation		Free-translation	
	BLEU	SARI	BLEU	SARI
Identical translation	34.65	17.87	29.31	15.86
Non Pre-training				
Encoder-Decoder	19.70	38.35	20.11	40.40
TransformerLM	42.86	51.91	35.96	49.78
Pre-training on parallel corpus				
Pre-train Encoder only	18.44	38.17	17.09	39.25
Pre-train Decoder only	10.86	31.10	8.92	31.19
Encoder-Decoder	14.38	33.92	15.04	36.18
TransformerLM	34.45	46.36	25.54	42.74
+ language modeling loss	30.03	43.52	24.67	41.99
Pre-training on Wikipedia				
Pre-train Encoder only	25.21	41.63	24.16	42.86
Pre-train Decoder only	7.44	30.88	10.38	33.70
Encoder-Decoder	13.32	34.41	13.67	36.16
TransformerLM	**44.15**	**52.46**	**37.37**	**50.39**
+ language modeling loss	40.69	50.37	34.22	48.55

- **summary**: The simplified version that contains document-level rewritings such as sentence extraction in addition to sentence-level rewritings.

These consist of grammar and vocabulary defined in the Japanese-Language Proficiency Test Level 2 (N2). Each simplified sentence is manually aligned. In this paper, we attempt to translate an original sentence into a *literal-translation* sentence or a *free-translation* sentence, which is word-level or sentence level simplification. The *summary*, which is a document-level simplification, will be addressed in the future.

B. Model Specifications and Training Details

We use a unidirectional transformer language model with six layers and 16 masked self-attention heads. We set the dimension of a word embedding layer to 512, and the dimension of feedforward networks to 2048. In the encoder-decoder model, both the encoder and decoder use the same parameters. We use the Stochastic Gradient Descent (SGD) for optimizing all models. We set the initial learning rate to 0.25, and 0.1 multiplies it when a validation loss has stopped improving during 10 epochs. The training ends if the learning rate becomes less than 1.0×10^{-5}.

C. Evaluation

We evaluated the model's output based on two metrics, BLEU [37] and SARI [10]. BLEU is a traditional evaluation metric for machine translation tasks. It has a positive correlation with fluency and meaning preservation in text simplification task that does not include sentence splitting [38]. SARI is a recently proposed simplification metric that compares the **S**ystem output **A**gainst **R**eferences and

Table II
EXAMPLES OF OUTPUT

	Examples	English translation of the left column
Input	健康 診査 票 が ない と 健診 を 受ける こと が でき ません (今回 ご 案内 さ せ て いただい た 郵便 物 に 同封 さ れ て い ます)。	If you do not have a medical checkup form, you will not be able to receive a medical checkup. (It is enclosed in this mail).
Encoder-Decoder	健康 診断 の 結果 が でき ませ ん 。 です 。	You can not get the result of your health check.
TransformerLM	健康 診断 の 紙 が ない と 健康 診断 を 受ける こと が でき ませ ん (今回 案内 し た 手紙 に 入っ て い ます) 。	If you do not have a form for medical checkup, you will not be able to receive a medical checkup. (It is in this mail).
Reference	健康 診断 票 が なかっ たら 健康 診断 を 受ける こと が でき ませ ん (今回 案内 し た 手紙 に 入っ て い ます) 。	If you do not have a medical checkup form, you will not be able to receive a medical checkup. (It is in this mail).
Input	警報 ・ 避難 の 指示 等 の 内容 の 伝達 訓練 及び 被災 情報 ・ 安否 情報 に 係る 情報 収集 訓練	Training to transmit information about warning and evacuation instructions and training to gather information regard to disaster and safety.
Encoder-Decoder	逃げる 住民 を 案内 の 情報 を 集め て 、 整理 し ます 。	Gather and organize guides for the people who will run away.
TransformerLM	警報 ・ 逃げる 指示 など の 内容 の 連絡 練習 と 災害 に つい て の 情報 を 集め て の 練習	Training to transmit information about warning and instructions to escape and training to gather information about disasters.
Reference	警報 や 逃げる 指示 など の 内容 を 伝える 練習 と 災害 に あっ た 情報 ・ 無事 か どう か の 情報 に つい て の 情報 を 集める 練習	Training to transmit information about warning and instructions to escape, and training to gather information about disaster and safety.

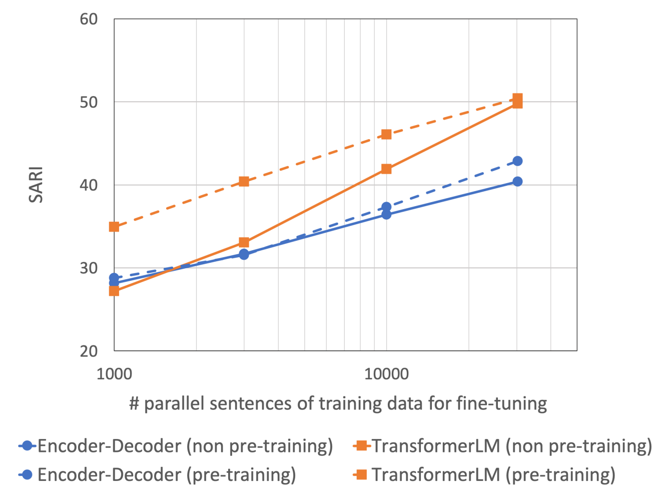

Figure 2. SARI in various data size. Round points (blue line) and square points (orange line) denote Encoder-Decoder and TransformerLM, respectively. The dotted line denotes a model pre-training by Wikipedia, and the solid line denotes a model without pre-training.

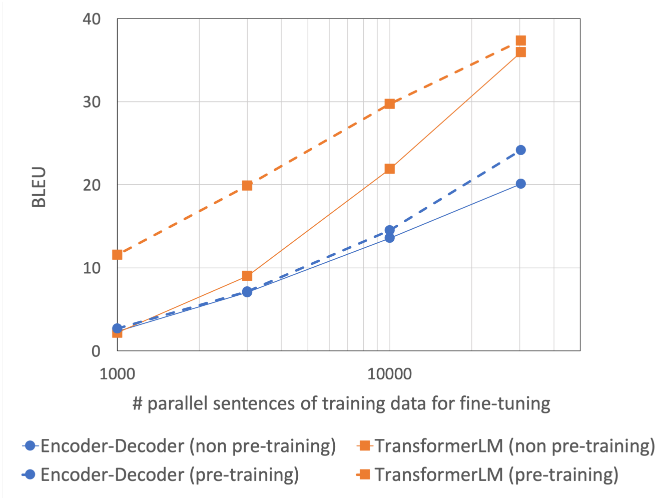

Figure 3. BLEU in various data size. Round points (blue line) and square points (orange line) denote Encoder-Decoder and TransformerLM, respectively. The dotted line denotes a model pre-training by Wikipedia, and the solid line denotes a model without pre-training.

against the **Input** sentence. This is an arithmetic average of n-gram precision and the recall of three rewrite operations: addition, retention, and deletion. It rewards addition operations, where system output was not in the input but occurred in the references. In addition, it rewards words retained/deleted in both the system output and the references. SARI has a positive correlation with simplicity [38], [39].

V. Results

Comparison of each system is shown in Table III. *Identical translation* denotes a system that outputs an input sentence. Furthermore, *Encoder-Decoder*, *Pre-train Encoder only*, and *Pre-train Decoder* are the models described in section III-B. *TransformerLM* is a model described in section III-C. +*language modeling loss* denotes a model in which language modeling loss adds to translation loss. As shown in Table II, TransformerLM can copy source words more correctly than the encoder-decoder model. Moreover, it outputs sentences close to reference sentence lengths, unlike the encoder-decoder model outputs. As a result, TransformerLM significantly outperforms Encoder-Decoder in BLEU and SARI.

The results of SARI and BLEU in various supervised data sizes are shown in Figure 2 and Figure 3. We use the encoder-decoder model for which only the encoder is pre-trained and TransformerLM without language modeling loss. As a result, pre-training with large-scale monolingual corpus is more effective on the TransformerLM than on the transformer-based encoder-decoder model. Especially, it is a surprising result that TransformerLM fine-tuned with only 3,000 examples has performance comparable to the encoder-decoder model trained with all the supervised data.

VI. Conclusion

We attempt a simple approach which fine-tunes the pre-trained language model for text simplification with a small parallel corpus. We experiment with the following two models: transformer-based encoder-decoder model and a language model that receives a joint input of original and simplified sentences, called TransformerLM. As a result, pre-training with large-scale monolingual corpus is more effective on the TransformerLM than on the transformer-based encoder-decoder model. We show that the simple TransformerLM outperforms the encoder-decoder model. Furthermore, TransformerLM fine-tuned with only 3,000 supervised examples can achieve performance comparable to a transformer-based encoder-decoder model trained all supervised data.

Acknowledgment

This work was supported in part by JSPS KAKENHI Grants-in-Aid for Challenging Research (Exploratory) Grant ID 17K18481.

References

[1] R. Chandrasekar, C. Doran, and B. Srinivas, "Motivations and Methods of Text Simplification," *Proceedings of the 16th conference on Computational linguistics-Volume 2*, vol. 0, no. 9, pp. 1041–1044, 1996. [Online]. Available: https://www.a@clweb.org/anthology/C96-2183

[2] A. Siddharthan, A. Nenkova, and K. McKeown, "Syntactic simplification for improving content selection in multi-document summarization," *Proceedings of the 20th International Conference on Computational Linguistics (COLING2004)*, pp. 896–902, 2004. [Online]. Available: https://www.aclweb.org/anthology/C04-1129

[3] W. Xu and R. Grishman, "A parse-and-trim approach with information significance for Chinese sentence compression," *Proceedings of the 2009 Workshop on Language Generation and Summarisation, ACL-IJCNLP 2009*, no. August, p. 48, 2010. [Online]. Available: https://www.aclweb.org/anthology/W09-2809

[4] D. Vickrey and D. Koller, "Sentence Simplification for Semantic Role Labeling," *Proceedings of ACL-08: HLT*, no. June, pp. 344–352, 2008. [Online]. Available: http://www.aclweb.org/anthology/P/P08/P08-1040

[5] M. Miwa, R. Sætre, Y. Miyao, and J. Tsujii, "Entity-focused sentence simplification for relation extraction," *Coling 2010 - 23rd International Conference on Computational Linguistics, Proceedings of the Conference*, vol. 2, no. August, pp. 788–796, 2010.

[6] H.-b. Chen, H.-H. Huang, H.-H. Chen, and C.-T. Tan, "A Simplification-Translation-Restoration Framework for Cross-Domain SMT Applications," *Coling-2012*, vol. 2, no. December, pp. 545–560, 2012. [Online]. Available: https://www.aclweb.org/anthology/C12-1034

[7] S. Štajner and M. Popovic, "Can Text Simplification Help Machine Translation?" *In Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, vol. 4, no. 2, pp. 230–242, 2016. [Online]. Available: https://www.aclweb.org/anthology/W16-3411

[8] K. Iwata, "The Preference for English in Linguistic Services: 'Japanese for Living: Countrywide Survey' and Hiroshima," *The Japanese Journal of Language in Society*, vol. 13, pp. 81–94, 2010.

[9] S. Wubben, A. van den Bosch, and E. Krahmer, "Sentence Simplification by Monolingual Machine Translation," *The 50th Annual Meeting of the Association for Computational Linguistics*, vol. 1, no. January 2014, pp. 1015–1024, 2012. [Online]. Available: https://www.aclweb.org/anthology/P12-1107

[10] W. Xu, C. Napoles, E. Pavlick, Q. Chen, and C. Callison-Burch, "Optimizing Statistical Machine Translation for Text Simplification," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 401–415, 2018. [Online]. Available: https://www.aclweb.org/anthology/Q16-1029

[11] S. Nisioi, S. Štajner, S. P. Ponzetto, and L. P. Dinu, "Exploring Neural Text Simplification Models," *Proceedings ofthe 55th Annual Meeting ofthe Association for Computational Linguistics (Short Papers)*, pp. 85–91, 2017. [Online]. Available: https://www.aclweb.org/anthology/P17-2014

[12] X. Zhang and M. Lapata, "Sentence Simplification with Deep Reinforcement Learning," *Proceedings ofthe 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 584–594, 2017. [Online]. Available: https://www.aclweb.org/anthology/D17-1062

[13] S. Zhao, R. Meng, D. He, S. Andi, and P. Bambang, "Integrating Transformer and Paraphrase Rules for Sentence Simplification," *Proceedings ofthe 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3164–3173, 2018. [Online]. Available: https://aclweb.org/anthology/D18-1355

[14] H. Guo, R. Pasunuru, and M. Bansal, "Dynamic Multi-Level Multi-Task Learning for Sentence Simplification," *Proceedings ofthe 27th International Conference on Computational Linguistics*, pp. 462–476, 2018. [Online]. Available: https://www.aclweb.org/anthology/C18-1039

[15] R. Kriz, J. Sedoc, M. Apidianaki, C. Zheng, G. Kumar, E. Miltsakaki, and C. Callison-Burch, "Complexity-Weighted Loss and Diverse Reranking for Sentence Simplification," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019. [Online]. Available: https://www.aclweb.org/anthology/N19-1317

[16] Z. Zhu, D. Bernhard, and I. Gurevych, "A Monolingual Tree-based Translation Model for Sentence Simplification," *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 1353–1361, 2010. [Online]. Available: https://www.aclweb.org/anthology/C10-1152

[17] K. Woodsend and M. Lapata, "Learning to Simplify Sentences with Quasi-Synchronous Grammar and Integer Programming," *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 409–420, 2011. [Online]. Available: https://www.aclweb.org/anthology/D11-1038

[18] W. Coster and D. Kauchak, "Simple English Wikipedia: A New Text Simplification Task," *Proceedings ofthe 49th Annual Meeting ofthe Association for Computational Linguistics:shortpapers*, pp. 665–669, 2011. [Online]. Available: https://www.aclweb.org/anthology/P11-2117

[19] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2227–2237, 2018. [Online]. Available: https://aclweb.org/anthology/N18-1202

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186, 2018. [Online]. Available: https://www.aclweb.org/anthology/N19-1423

[21] J. Howard and S. Ruder, "Universal Language Model Fine-tuning for Text Classification," *Proceedings ofthe 56th Annual Meeting ofthe Association for Computational Linguistics*, pp. 328–339, 2018. [Online]. Available: https://www.aclweb.org/anthology/P18-1031

[22] S. Shleifer, "Low Resource Text Classification with ULMFit and Backtranslation," *CoRR*, pp. 1–9, 2019. [Online]. Available: http://arxiv.org/abs/1903.09244

[23] A. Chronopoulou, C. Baziotis, and A. Potamianos, "An Embarrassingly Simple Approach for Transfer Learning from Pretrained Language Models," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2089–2095, 2019. [Online]. Available: https://www.aclweb.org/anthology/N19-1213

[24] R. Sennrich, B. Haddow, and A. Birch, "Improving Neural Machine Translation Models with Monolingual Data," *Proceedings ofthe 54th Annual Meeting ofthe Association for Computational Linguistics*, pp. 86–96, 2016. [Online]. Available: https://www.aclweb.org/anthology/P16-1009

[25] S. Edunov, M. Ott, M. Auli, and D. Grangier, "Understanding Back-Translation at Scale," *Proceedings ofthe 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 489–500, 2018. [Online]. Available: https://aclweb.org/anthology/D18-1045

[26] J. Qiang, "Improving Neural Text Simplification Model with Simplified Corpora," *CoRR*, 2018. [Online]. Available: http://arxiv.org/abs/1810.04428

[27] D. Kauchak, "Improving Text Simplification Language Modeling Using Unsimplified Text Data," *Proc. of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1537–1546, 2013. [Online]. Available: https://www.aclweb.org/anthology/P13-1151

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *Proceedings of the 31st Conference on Neural Information Processing Systems*, pp. 5998–6008, 2017. [Online]. Available: https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf

[29] A. Radford and T. Salimans, "Improving Language Understanding by Generative Pre-Training (transformer in real world)," *OpenAI*, pp. 1–12, 2018. [Online]. Available: https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

[30] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," *Proceedings of twenty-eighth Conference on Neural Information Processing Systems*, pp. 1–9, 2014. [Online]. Available: https://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf

[31] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," *Proceedings of 3rd International Conference on Learning Representations*, pp. 1–15, 2015.

[32] P. Ramachandran, P. J. Liu, and Q. V. Le, "Unsupervised Pretraining for Sequence to Sequence Learning," *Proceedings ofthe 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 383–391, 2017. [Online]. Available: https://www.aclweb.org/anthology/D17-1039

[33] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio, "An Empirical Investigation of Catastrophic Forgetting in Gradient-Based Neural Networks," *CoRR*, 2013.

[34] U. Khandelwal, K. Clark, D. Jurafsky, and L. Kaiser, "Sample Efficient Text Summarization Using a Single Pre-Trained Transformer," *CoRR*, 2019. [Online]. Available: http://arxiv.org/abs/1905.08836

[35] A. Hoang, A. Bosselut, A. Celikyilmaz, and Y. Choi, "Efficient Adaptation of Pretrained Transformers for Abstractive Summarization," *CoRR*, 2019. [Online]. Available: http://arxiv.org/abs/1906.00138

[36] M. Moku, K. Yamamoto, and A. Makabi, "Automatic Easy Japanese Translation for information accessibility of foreigners," *Proceedings of the Workshop on Speech and Language Processing Tools in Education*, pp. 85–90, 2012. [Online]. Available: https://www.aclweb.org/anthology/W12-5811

[37] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a Method for Automatic Evaluation of Machine Translation," *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, 2002. [Online]. Available: https://www.aclweb.org/anthology/P02-1040

[38] E. Sulem, O. Abend, and A. Rappoport, "BLEU is Not Suitable for the Evaluation of Text Simplification," *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 738–744, 2018. [Online]. Available: https://www.aclweb.org/anthology/D18-1081

[39] T. Vu, B. Hu, T. Munkhdalai, and H. Yu, "Sentence Simplification with Memory-Augmented Neural Networks," *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 79–85, 2018. [Online]. Available: https://www.aclweb.org/anthology/N18-2013