ISSN: 2664-9640

Textual Readability Assessment: A RoBERTa-based model

Guanchen Lv

School of Computer Science, Sichuan University, Chengdu, China

Abstract

The assessment of text readability is quite important for students and educators, and traditional methods evaluate it from word and syntactic and are not based on text-level features. For the task of text readability prediction, this paper provides a RoBERTa-based approach to generate word embeddings by fine-tuned RoBERTa and then use support vector machines to predict the readability score with word embeddings inputs. The evaluation shows that the approach demands fewer data and effectively reduces the inference time while keeping high accuracy.

Keywords

Textual Readability, RoBERTa, SVM.

1. Introduction

Reading is the key for students to ensure that they can achieve academic success. For reading materials, the only way to ensure students' better comprehension of texts and reliable long-term improvement in reading proficiency is to select texts that are close to their reading abilities. However, text readability assessment has been a difficult task. Traditional methods of assessing readability based either on characters or syllables or on syntactic complexity, such as Flesch-Kincaid or Lexile, ignore many text-level features and are too costly for schools and teachers to a practical level.

In 2017 Google proposed the Bidirectional Encoder Representations from Transformers (BERT) [1], a pre-trained model that was first used in machine translation, and then used widely in other tasks, even computer vision. Pre-trained BERT can be fine-tuned with just one additional output layer to models for a variety of natural language processing tasks. The ease of use for BERT provides a solid approach to predict text readability with less cost. Compared with BERT, RoBERTa [2], a revised BERT model, uses more parameters, larger batch sizes, and more training data, including CC-NEWS and other 160GB of plain text, while the original BERT uses 16GB BookCorpus dataset and English Wikipedia for training.

In this article, using the readability-rated dataset, CLEAR-Corpus, as the training data, I trained and finetuned the model on RoBERTa to predict textual readability first. Then inspired by ideas of using word embeddings to improve the performance of predicting [3-4]. I design the second approach to map the texts into vectors with RoBERTa and feed word embedding into the support vectors machine (SVM), which may increase the performance of predicting the reading difficulty of textual excerpts. Eventually, the accuracy across the two models is compared and analyzed. But constrained by the lack of datasets of long texts with labels, the method adopted in this paper does not necessarily perform well in longer texts.

In total, this paper provides two approaches for predicting textual readability. Both methods provide high estimation accuracy. Students, teachers, parents, and researchers would all benefit from reliable estimates of textual readability.

ISSN: 2664-9640

2. Related Work

To help educators match students to reading materials at the right level of difficulty, especially in language teaching [5], it is necessary to assess their readability. Textual readability assessment has been around for nearly a century, in the early era of textual readability assessment, people used weighted linear functions to measure text difficulty, taking into account variables such as the number of words, the number of sentences, length of text, percentage of difficult words, and there are many models based on those features such as Flesch [6] and SMOG [7]. Machine learning has only started to be gradually applied to this field in the last 20 years, and researchers have considered many new features such as syntactic complexity [8-9] and psycholinguistic processes [10].

In terms of models, textual readability assessment is usually modeled as a supervised classification problem. But it can also be modeled as a regression [11] and ranking [12-13] problem. For tasks in natural language processing, researchers convert text into word embedding in different ways such as Word2Vec [14], GloVe [15], FastText [16], and ELMo [17], then input these word vectors into neural networks or machine learning models to carry out specific tasks. But in traditional neural networks such as recurrent neural network (RNN) structure, their Encoder-Decoder mechanism relies on a fixed context vector, but it cannot fully express the information of the whole sequence, resulting in difficulties with long sequences. Google machine translation team proposes the Transformer [1], a new modeling paradigm, which almost completely replaces the previous RNN and CNN structure.

The recent readability ranking model proposed by Lee and Vajjala [13], uses only BERT embeddings as the input but shows very well cross-lingual learning transfer. And in this article, I will also use word embeddings to predict textual readability without other features.

3. Model Architecture

This paper uses two models to predict text readability—RoBERTa and RoBERTa-SVM. I trained and fine-tuned RoBERTa first and got the RoBERTa model. Then use RoBERTa to generate word vectors for the training set, and then use it to train the RoBERTa-SVM model. The mechanism of models will be introduced in detail in 3.1, 3.2, and 3.3.

3.1. RoBERTa

The RoBERTa model is based on the classic stacked encoder-decoder structure, where the encoder maps the sequence of words $(x_1, \dots, x_n)$ into a sequence $(z_1, \dots, z_n)$, and then the decoder generates an output sequence $(y_1, \dots, y_m)$ for the given z.

Every encoder and decoder mainly relied on two new elements: the multi-head attention mechanism and feed-forward blocks. The multi-head attention mechanism is an extension of the self-attention mechanism. Each input of a self-attention layer would receive three vectors: query, key, and value. And self-attention layer would map a query and a set of key-value pairs to an output. The output is calculated as:

$$Attention(q, k, v) = softmax(\frac{qk^t}{\sqrt{d_k}})v \quad (1)$$

And the authors of transformers proposed the idea of multi-head attention [1]. For every $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$, the mutil-head attention output is:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_\text{h})\text{W}^\text{O} \quad (2)$$

where W_i^Q, W_i^K, W_i^V is matrices with the size of word embedding dim and hidden size, and W^O is the matrix with input sequence length and hidden size.

ISSN: 2664-9640

After the multi-head attention layer, every encoder and decoder has one additional feed-forward network, which consists of two linear layers with ReLU non-linearity in between:

$$FFN(x) = max(0, xW_1 + b_1)\, W_2 + b_2 \tag{3}$$

The specific pre-trained and fine-tuned setting would be introduced in the next section.

3.2. SVM prediction

Given the word embeddings x_i generated by RoBERTa and textual readability y_i as training set: $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$. To find a function $p(x)$ with parameter w to predict textual readability y, the error function with ϵ-insensitive loss $E_\epsilon(p(x) - y_n)$ and regularization factor C is:

$$J(w) = C \sum_{n=1}^{N} E_\epsilon(p(x) - y_n) + \frac{1}{2}||w||^2 \tag{4}$$

For data points outside the tudes, use two slack variables to represent the outside degree:

$$y_i \le p(x_i) + \epsilon + \xi_i^+ \tag{5}$$

$$y_i \ge p(x_i) - \epsilon - \xi_i^- \tag{6}$$

The loss function can be rewritten as:

$$J(w) = C \sum_{n=1}^{N} E_\epsilon(\xi_i^+ + \xi_i^-) + \frac{1}{2}||w||^2 \tag{7}$$

Optimizing the above equation with the Lagrangian multipliers $a_n \ge 0, \hat{a}_n \ge 0$ to minimalize the loss function, the dual problem is to maximize:

$$\tilde{L}(a, \hat{a}) = -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}(a_n - \hat{a}_n)(a_m - \hat{a}_m){x_n}^T x_m - \epsilon\sum_{m=1}^{N}(a_n + \hat{a}_n) + \sum_{m=1}^{N}(a_n - \hat{a}_n)y_n \tag{8}$$

Use the radial basis function as the kernel function $k(x, x_n)$, the kernelized solution is:

$$p(x) = \sum_{n=1}^{N}(a_n - \hat{a}_n)k(x, x_n) + b \tag{9}$$

The above support vectors regression model is trained by word embeddings x_i generated by RoBERTa and it would predict the readability score as the output. Compared with RoBERTa, the loss function of SVM has a globally optimal solution, and when the dataset is not large, the convergence is fast and the hyperparameters are more explanatory.

3.3. 5-Folds Datasets

I adopted a broad applicable procedure [18] on the datasets to reduce the variance of the prediction, alleviate overfitting and improve the generalization of the model: (1) Split the training data set into k groups; (2) Use $(k-1)$ groups for training and the remaining 1 for validation, and this training will produce one model; (3) Repeat the step 2 until each unique group has been validated, which will produce k models; (4) Take the average of the output from the k models as the final outputs. And because of the averaging effect, the variance of predictions in the final model would be significantly reduced. In this article, the k is set as 5. So, the final prediction of the textual readability is:

$$\text{Predicted Readability} = \frac{\sum_1^5 p_i(x)}{5} \tag{10}$$

Figure 1 shows the overall architecture of the RoBERTa-SVM model.

ISSN: 2664-9640

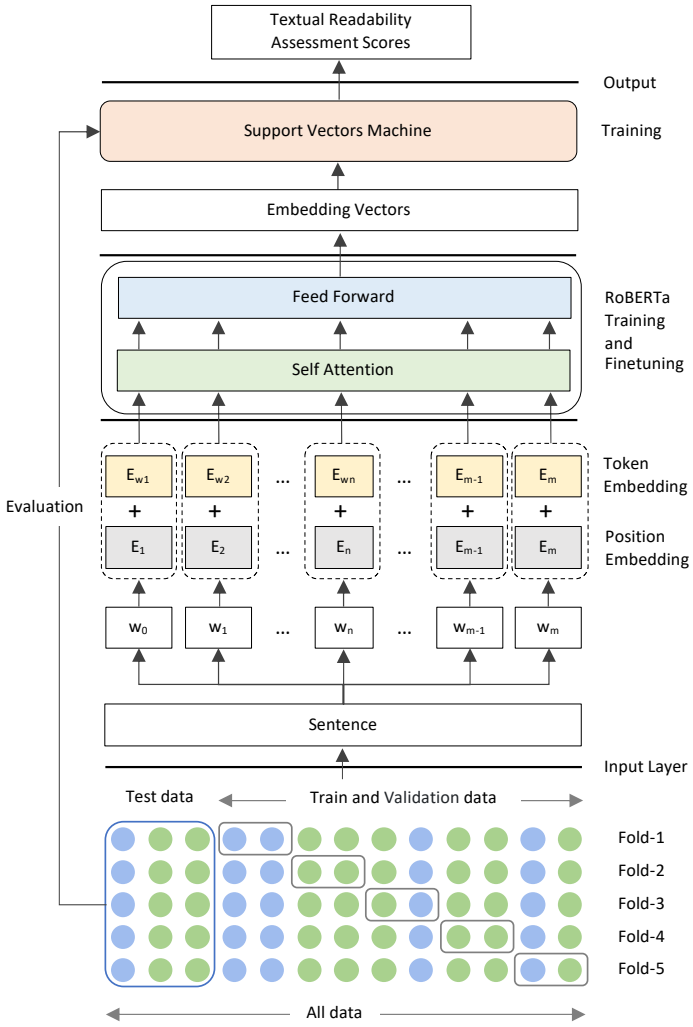

Figure 1: RoBERTa-SVM Architecture

4. Training

4.1. Dataset and Hardware

The dataset used in this article is CLEAR-Corpus [19], which provides readability scores for nearly 5000 excerpts leveled for 3rd to12th grade readers and covers over 250 years of writing in two different genres, along with information about the excerpts' year of publishing, genre, and other meta-data. The dataset is split into the training set and test set (~1900 excerpts). The models are trained with 2 NVIDIA P100 GPUs.

4.2. Optimizer and Scheduler

The AdamW [20] is adopted as the optimizer, with $\beta_1 = 0.9, \beta_2 = 0.99$. The parameter θ would be updated by the following formula:

$$\theta_t \leftarrow \theta_{t-1} - \eta_t \left(\frac{\alpha \hat{\boldsymbol{m}}_t}{\sqrt{\hat{\boldsymbol{v}}_t} + \epsilon} + \lambda \theta_{t-1} \right) \tag{11}$$

where the η is the schedule multiplier, and $\boldsymbol{m}$ and $\boldsymbol{v}$ are the first and second moment vectors accordingly. And the learning rate $\alpha = 1e-5$, weight decay $\lambda = 0.01$. The complete fine-tuning setting refers to Table 1. Learning rates vary from layer to layer, with higher learning rates applied to the top layer and lower learning rates applied to the bottom layer [21]. Specifically, the learning rate is 2.5 times for layers 4-7 and 5 times for layers 8-11.

ISSN: 2664-9640

To ensure the stable training of the transformer [22], warm-up and dropout is used in this article. Considering the common finetuning practice in transformers [23], the $drop\_out$ is set as 0.2, and the learning rate would be set to decrease following the values of the cosine function with $warmup\_steps = 50$.

Table 1: Model Setting

config	values
attention_probs_dropout_prob	0.1
bos_token_id	0
classifier_dropout	null
eos_token_id	2
hidden_act	gelu
hidden_dropout_prob	0.1
hidden_size	768
initializer_range	0.02
intermediate_size	3072
layer_norm_eps	1.00E-05
max_position_embeddings	514
num_attention_heads	12
num_hidden_layers	12
pad_token_id	1
position_embedding_type	absolute
transformers_version	4.23.0
type_vocab_size	1
use_cache	TRUE
vocab_size	50265
learning_rate	1.00E-05
warmup_steps	50
weight_decay	0.01
drop_out	0.2

Figures 2 and 3 are the learning curves of the RoBERTa and SVM. As can be seen from the figure, the SVM requires very few samples. Less than 1000 samples made it converge.

ISSN: 2664-9640

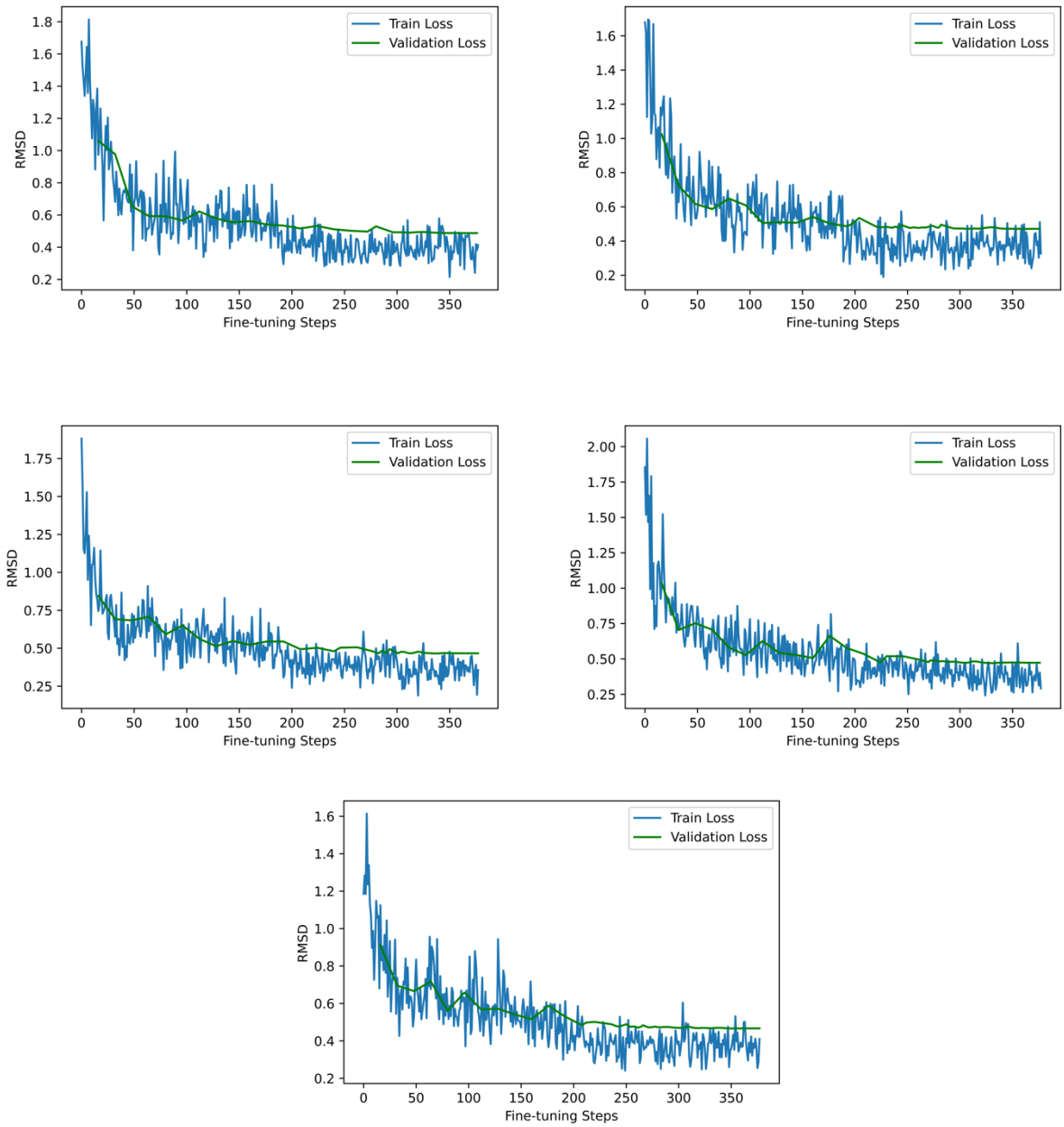

Figure 2: RoBERTa Fine-tuning (5-Folds)

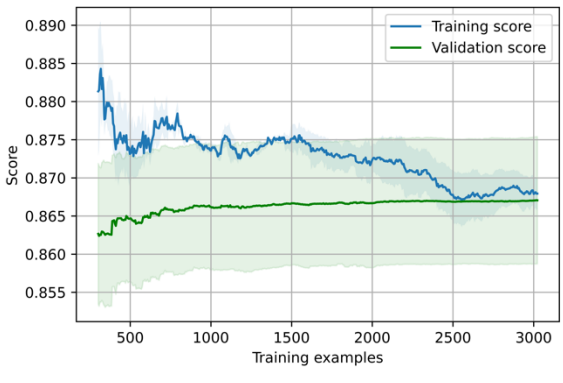

Figure 3: SVM Learning Curve (5-Folds)

ISSN: 2664-9640

5. Evaluation

In this paper, RMSD is selected as the evaluation metric, and it is defined as follows:

$$RMSD = \sqrt{\frac{\sum_{i=1}^{n}(p_i - y_i)^2}{n}} \quad (1)$$

Table 2 shows the average RMSD of 100 tests of the RoBERTa and RoBERTa-SVM models. As can be seen from Table 2, the prediction accuracy of RoBERTa is slightly higher than RoBERTa-SVM, but there is no noticeable difference. At the same time, by training five models on the five-fold dataset separately and taking the average of their outputs, the RMSD is lower, which means its generalization has been effectively improved.

Table 2: RMSD of Models

Model	RMSD
5-Folds RoBERTa-SVM	0.4516
RoBERTa-SVM	0.4638
5-Folds RoBERTa	0.4512
RoBERTa	0.4612

Another evaluation metric is inference time. Fast inference time is required to deploy a deep network model into a real-world environment. The inference time was calculated considering asynchronous execution and GPU warm-up. After completing GPU warm-ups with 50 random inputs, the inference time was obtained by averaging the running time of 300 test inputs considering GPU asynchronous execution Table 3 shows the inference times of models.

Table 3: Inference Time

Model	Inference Time(ms)
5-Folds RoBERTa-SVM	15027.357817
RoBERTa-SVM	3071.214675
5-Folds RoBERTa	74899.390625
RoBERTa	14005.799805

In general, the use of word embeddings is an effective approach to reducing the complexity of the model, which can significantly reduce the inference time. However, the reduction in inference time may come at the expense of RMSD. Also, the SVM model must be trained after Fine-tuned RoBERTa outputs the embedding vectors of the training set, so the overall training will be more complicated. The textual readability will not change particularly much in the coming period, so for SMEs, most of the time will be focused on inference rather than training, so it is more critical to select a model with less complexity.

6. Conclusion

In this article, I utilized two models, RoBERTa and RoBERTa- SVM, to predict the readability of text, which is slightly different from traditional NLP tasks. I trained the RoBERTa model on a small dataset and trained the SVM model by word embedding outputs generated by RoBERTa. I then compared the RMSD and inference time of both RoBERTa and RoBERTa-SVM models. From the evaluation results, RoBERTa-SVM did not improve the prediction accuracy but

ISSN: 2664-9640

significantly decreased the inference time due to the reduction in model complexity. For real-world deployment, RoBERTa-SVM may provide an alternative idea for text-related tasks on a small-size dataset.

Readability is not only on textual latitude but also on other dimensions such as images. This may involve some CV tasks, but it is certainly more challenging work because of the lack of a priori knowledge at the practical level. Also, the limitation of this paper is that there is a lack of large-scale datasets with annotations to train the model, and this paper can only be developed on a smaller dataset. This problem can be alleviated by in-domain and cross-domain pre-training [24], but a larger labeled dataset would certainly be more helpful.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” Advances in neural information processing systems, 2017.

[2] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, " RoBERTa: A Robustly Optimized BERT Pretraining Approach," ArXiv, 2019.

[3] P. P. Shelke and A. N. Korde, "Support Vector Machine based Word Embedding and Feature Reduction for Sentiment Analysis-A Study," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), 2020, pp. 176-179.

[4] O. Press and L. Wolf, “Using the output embedding to improve language models,” Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, 2017.

[5] K. Collins-Thompson and J. Callan, “Information retrieval for language tutoring: An overview of the reap project,” Proceedings of the 27th annual international conference on Research and development in information retrieval, pp. 544–545, 2004.

[6] R. Flesch, “A new readability yardstick.,” Journal of Applied Psychology, vol. 32, no. 3, pp. 221–233, 1948.

[7] G. H. McLaughlin, “Smog grading-a new readability formula,” Journal of reading, vol. 12, no. 8, pp. 639–646, 1969.

[8] M. Heilman, K. Collins-Thompson, J. Callan, and M. Eskenazi, “In Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics,” pp. 460–467, 2007.

[9] S. Vajjala and D. Meurers, “On improving the accuracy of readability classification using insights from second language acquisition. In Proceedings of the seventh workshop on building educational applications using NLP,” pp. 163–173, 2012.

[10] D. M. Howcroft and V. Demberg, “Psycholinguistic models of sentence processing improve sentence readability ranking,” Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, 2017.

[11] S. Vajjala and D. Meurers, “Readability assessment for text simplification,” Recent Advances in Automatic Readability Assessment and Text Simplification, vol. 165, no. 2, pp. 194–222, 2014.

[12] Y. Ma, E. Fosler-Lussier, and, R. Lofthus, “Ranking-based readability assessment for early primary children’s literature,” In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 548–552, 2012.

[13] J. Lee and S. Vajjala, “A neural pairwise ranking model for readability assessment,” Findings of the Association for Computational Linguistics, 2022.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” Proceedings of Workshop at ICLR, 2013.

[15] J. Pennington, R. Socher, and C. Manning, “GloVe: Global Vectors for Word Representation,” In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543, 2014.

ISSN: 2664-9640

[16] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification," In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, pp. 427-431, 2017.

[17] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep Contextualized Word Representations," In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 2227–2237, 2018.

[18] Y. Jung and J. Hu, "A k-fold averaging cross-validation procedure," Journal of Nonparametric Statistics, vol. 27, no. 2, pp. 167–179, 2015.

[19] S. A. Crossley, A. Heintz, J. Choi, J. Batchelor, and M. Karimi, "CommonLit Ease of Readability (CLEAR) Corpus," Proceedings of the 14th International Conference on Educational Data Mining, 2021.

[20] Loshchilov. I. and Hutter. F, "Decoupled Weight Decay Regularization," ICLR, 2019.

[21] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2018.

[22] R. Xiong, Y. Yunchang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T.-Y. Liu, "On Layer Normalization in the Transformer Architecture," ICML, 2020.

[23] P. Izsak, M. Berchansky, and O. Levy, "How to train Bert with an academic budget," Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021.

[24] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune bert for text classification?," Lecture Notes in Computer Science, pp. 194–206, 2019.