FERT: Fixed Error Rate Training for Language Models

Kanghyeon Kim
kaist19@kaist.ac.kr

Jiseon Kim
jiseon_kim@kaist.ac.kr

Alice Oh
alice.oh@kaist.edu

Abstract

This paper introduces a new approach to training language models, called Fixed Error Rate Training (FERT), focusing on the efficiency and effectiveness of the training process. Drawing inspiration from the 'Eighty Five Percent Rule', we propose a new method that dynamically adjusts the difficulty of the training data according to the model's training progress. Unlike conventional methods, our approach evaluates and modifies the training dataset in real-time, ensuring that the learning path is optimally challenging for the model. We implement this methodology by fine-tuning a pre-trained BERT base model, utilizing a variety of tasks from the GLUE Benchmark dataset. Our experiments demonstrate that an error rate around 50% significantly enhances both the training efficiency and the quality of the model's performance on unseen data, improving the traditionally accepted 'Eighty Five Percent Rule.' for language models. The study reveals a noticeable improvement in training outcomes when the model is exposed to an appropriately challenging dataset. The findings present a promising potential for optimizing language model training even further.

1 Introduction

Recent advancements in natural language processing (NLP) have been significantly driven by the development of large language models such as BERT (Devlin et al., 2019). These models have shown remarkable success in a wide range of NLP tasks, thanks to their deep understanding of language contexts captured during the pre-training phase on vast corpora. Nonetheless, the areas of training efficiency for these models are still open for further research and enhancement.

For more efficient training methodologies, the concept of curriculum learning (CL) has been introduced (Elman, 1993). This strategy, inspired by the ways humans learn, suggests that machine learning models can similarly benefit from a gradual increase in the difficulty of training dataset, starting with simpler tasks and progressively advancing to more challenging ones. However, most curriculum learning algorithms depend on a linear progression of difficulty (Xu et al., 2020, Nagatsuka et al., 2021), which doesn't consider the model's current state or learning progress during training. There has been an attempt to evaluate the model's capability dynamically and feed in modified dataset accordingly (Lalor and Yu, 2020) but it requires an "artificial crowd" to determine the difficulty of dataset beforehand, which an individual model might perceive differently.

This paper proposes a new method called **Fixed Error Rate Training** (**FERT**), which uses a hyperparameter *error rate* during training. Unlike the traditional approach, this method dynamically evaluates and adjusts the training data's difficulty after every epoch based on the model's evolving capabilities, ensuring an optimal learning path. By aligning with the 'Eighty Five Percent Rule (Wilson et al., 2019),' which posits an ideal error rate for maximum learning efficiency for basic machine learning models, our method seeks to find a 'sweet spot' that is applicable for langauge model training.

2 Related Work and Background

Wilson et al., 2019 examined the training difficulty for various learning algorithms, discovering a 'sweet spot' where learning is the most efficient. They determined that the best error rate for training is about 15.87%, leading to the 'Eighty Five Percent Rule.' While their work provides a foundational understanding of error rates in learning, our study extends this concept to language models, which present a more complex set of challenges due to their deep architectures and nuanced understanding of language. Through FERT, we adapt and refine the concept of dynamic error rates to training language models, seeking to identify a more precise and effective 'sweet spot' for these advanced models.

FERT is also based on Lalor and Yu, 2020 in that the dataset is dynamically selected in the middle of training based on the model's capability. However, this approach first determines the difficulty of dataset using "artificial crowd" as opposed to our work where the difficulty of dataset is determined real-time depending on whether the current model gives the correct answer or not. FERT can be a more flexible curriculum learning because 1) the difficulty of the dataset doesn't have to be determined beforehand using artificial crowd and 2) it accommodates diverse model competencies, as the perceived difficulty of the dataset may vary based on each model's unique architecture and pre-training background.

3 Methods

FERT is based on the algorithm proposed by Lalor and Yu, 2020 with some modifications.

3.1 Steps

1. **Load the Pre-trained Bert Base Model:** Initialize a pre-trained language model.

2. **Evaluate Model Accuracy:** Determine the current accuracy of the model on the training dataset. If the accuracy exceeds 99.0% or the amount of dataset used for training has exceeded four times of the entire dataset, the training is terminated as the model is deemed sufficiently accurate.

3. **Modify the Dataset:** Adjust the training dataset based on the model's performance. Specifically, include a pre-determined proportion (error rate) of data for which the model previously gave incorrect answers.

4. **Proceed with One Epoch of Training:** Train the model for one epoch on the modified dataset.

5. **Repeat Steps 2–4:** Continuously evaluate, adjust the dataset, and train in a cyclical manner, repeating 2–4 until the termination condition from 2. **Evaluate Model Accuracy** is met.

In order to compare FERT with the traditional method, we train a language model for various tasks and analyze two aspects: 1) the amount of dataset fed into the model to achieve certain accuracy (training efficiency) and 2) the validation/test accuracy after completing training (training quality).

3.2 Training Efficiency

In this experiment, the definition of an "epoch" is different from the traditional meaning. Instead of a complete iteration over the entire training dataset, an epoch here refers to a training dataset modified to meet the target error rate. For example, if the model's current accuracy on the training dataset is 45% and the error rate is set to 10%, only the 45% of the training dataset that the model gave a correct answer for and 5% that the model gave an incorrect answer for are used in the epoch. That is, in this case, the "epoch" corresponds to only 50% of the entire training dataset.

Hence, epoch cannot be a measure of the amount of dataset fed into the model during training. Instead, we use "step" here, which corresponds to one batch of size 16. For training efficiency, we analyze how many steps it took for the model to achieve certain validation accuracy during training.

3.3 Training Quality

Test accuracy is utilized to assess the training quality, indicating the model's ultimate proficiency after the completion of training. While validation accuracy offers immediate feedback for training efficiency, test accuracy serves as the measurement for the training quality.

4 Experiments

4.1 Dataset

In the GLUE Benchmark (Wang et al., 2018), we specifically use CoLA, SST-2, MRPC, QQP, MNLI, and RTE for training. CoLA and SST-2 are single-sentence tasks, MRPC and QQP are similarity and paraphrase tasks, and MNLI and RTE are inference tasks.

4.2 Model

We use a pre-trained BERT base model (Devlin et al., 2019). As FERT requires the dataset to be labeled discretely (from which the error rate can be calculated), the model is fine-tuned as a sequence classifier.

4.3 Training Efficiency

During fine-tuning with FERT, validation accuracy is measured after every epoch. For each dataset, we initialized a new pre-trained BERT base model and fine-tuned with the dataset 100 times. The result is illustrated in Figure 1.

4.4 Training Quality

To evaluate the training quality of the error rate-based training method, we compare it against the traditional static training approach. Specifically, we train a pre-trained BERT base model 100 times with the traditional method and FERT at error rate $= 50\%$ (where the model is trained the most efficiently, according to our analysis on training efficiency in Section 5.2), respectively, and analyze the distribution of the test accuracy of them. For this analysis, we train the model only with the CoLA dataset.

5 Analysis

5.1 Training Efficiency

To visualize at which error rate does the model achieve a high enough validation accuracy, we eliminated the points with low validation accuracy from Figure 1 and obtained Figure 3. No matter which dataset the model is fine-tuned with, the validation accuracy achieves a certain threshold accuracy with the fewest steps when the error rate is set closer to around 50%, in contrast to the eighty five percent rule suggested in Wilson et al., 2019. This means that when the model is trained the most efficiently when we feed in dataset of which the model can give the correct answer for half and incorrect answer for the other half. At error rate $\approx 100\%$, although it might sounds promising to train the model with dataset the model can't give the right answer for, the model ends up forgetting what it had been trained with in the end. At error rate $\approx 0\%$, because we only feed in the dataset that the model is already trained enough for, the model no longer can learn. Therefore, in this experiment setting, although not exactly calculated, 50% can serve as a rule of thumb for language model training, like the 'Eighty Five Percent Rule' from Wilson et al., 2019.

5.2 Training Quality

The result illustrated in Figure 2 indicates a distinct distribution shift towards higher accuracy for FERT. This suggests that the adaptive nature of the training, guided by our proposed hyperparameter, contributes to a better learning process. The reason why FERT leaded to a high training accuracy is that, as the training proceeds, the model becomes excessively good at the training dataset. It is equivalent to training the model with error rate $\approx 0\%$, leading to a lower test accuracy in the end.

6 Limitation

The application of the error rate hyperparameter in this study is confined to fine-tuning a BERT model and discretely labeled dataset, which present a limitation. The generalizability of our findings to other architectures and datasets, especially autoregressive language models, remains untested. Autoregressive models, which require a large amount of corpora for pre-training and fine-tuning, could benefit from adapting the error rate hyperparameter based on their predictive capabilities.

7 Conclusion

This research presents a new approach to language model training, FERT, which uses an error rate as a hyperparameter and dynamically adjusts the difficulty of the training data in response to the model's evolving proficiency. Our method aims to optimize the training trajectory of language models like BERT, ensuring efficient and effective learning. Through extensive experimentation with the GLUE Benchmark dataset and a pre-trained BERT base model, we demonstrated the superiority of our dynamic, adaptive training method over traditional static approaches. Our findings reveal that training models with an appropriately challenging dataset can significantly enhance both training efficiency and the quality of the model's performance on unseen data. However, we acknowledges its limitations, primarily the focus on fine-tuning the BERT model and the necessity to explore the applicability of the error rate hyperparameter across diverse model architectures. Future work should aim to validate and refine this approach, extending it to other language models and datasets. The implications of this research are far-reaching, offering a new paradigm for training language models more effectively and efficiently.

References

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, Volume 1 (Long and Short Papers)*, pages

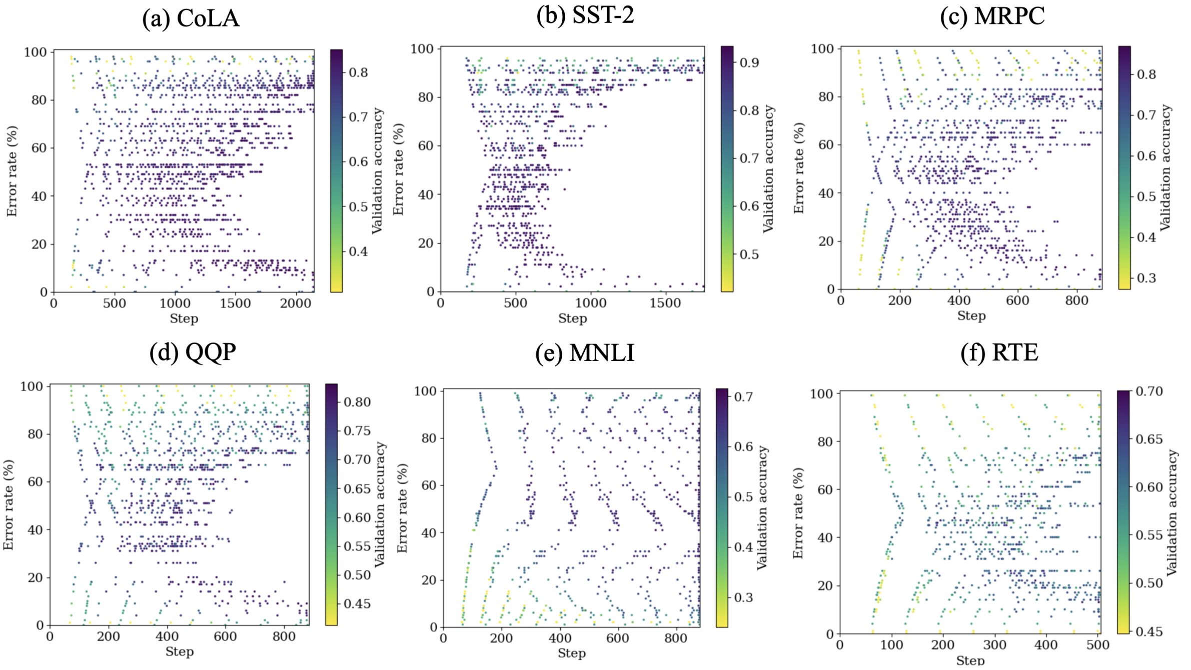


Figure 1: The validation accuracy measured after epochs during fine-tuning. Each step corresponds to training one batch of size 16. The y-axis represents the error rate, meaning the points on the same horizontal line are from training models with that error rate.

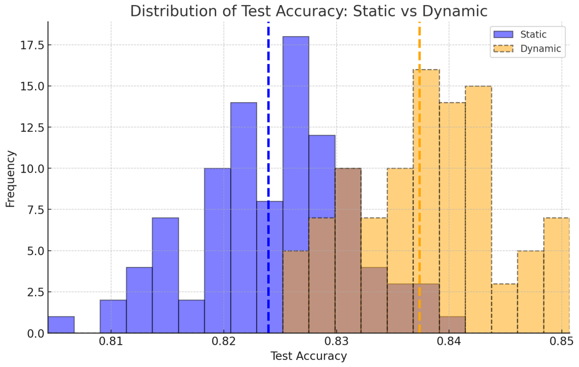


Figure 2: The distribution of test accuracy of the traditional method (blue) and FERT (yellow). The vertical dotted lines represent the average of the test accuracy of each method.

4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

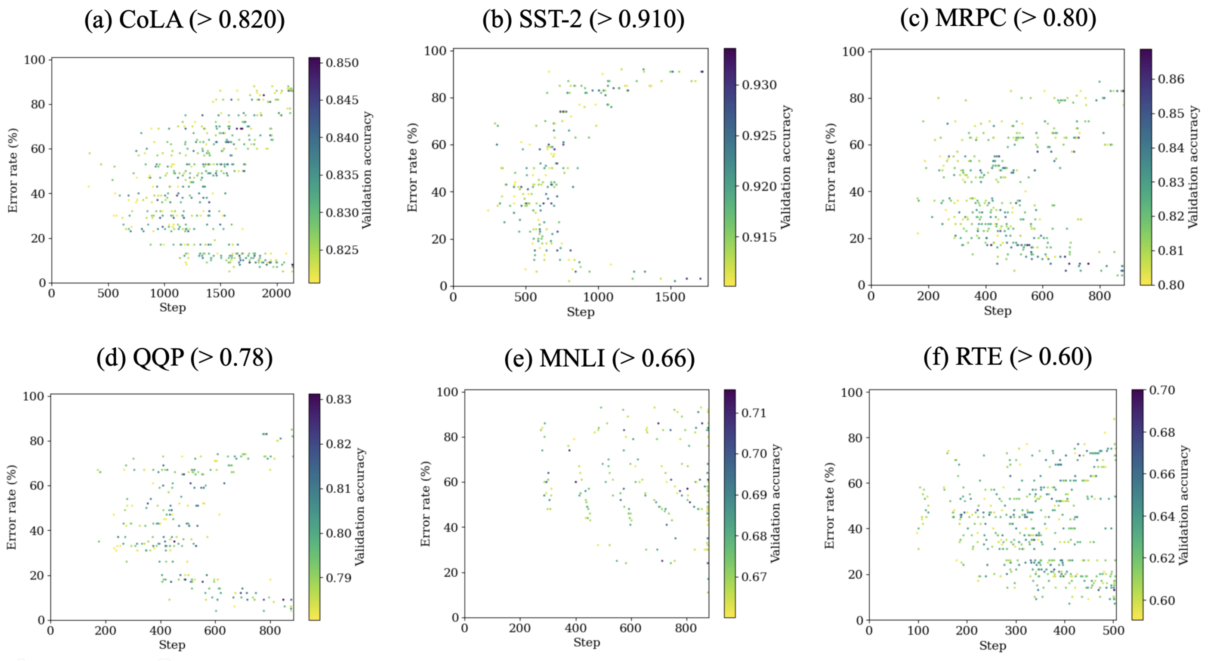
Jeffrey L. Elman. 1993. Learning and development in neural networks: the importance of starting small. *Cognition*, 48(1):71–99.

John P. Lalor and Hong Yu. 2020. Dynamic data selection for curriculum learning via ability estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 545–555, Online. Association for Computational Linguistics.

Koichi Nagatsuka, Clifford Broni-Bediako, and Masayasu Atsumi. 2021. Pre-training a BERT with curriculum learning by increasing block-size of input text. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 989–996, Held Online. INCOMA Ltd.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Robert C Wilson, Amitai Shenhav, Mark Straccia, and Jonathan D Cohen. 2019. The eighty five percent rule for optimal learning. *Nature Communications*, 10(1):4646.


(a) CoLA (> 0.820)

(b) SST-2 (> 0.910)

(c) MRPC (> 0.80)

(d) QQP (> 0.78)

(e) MNLI (> 0.66)

(f) RTE (> 0.60)

Figure 3: The validation accuracy measured after epochs during training. Each step corresponds to training one batch of size 16. Only the points with a validation accuracy higher than the threshold (next to the names of datasets) are plotted here.

Benfeng Xu, Licheng Zhang, Zhendong Mao, Quan Wang, Hongtao Xie, and Yongdong Zhang. 2020. Curriculum learning for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6095–6104, Online. Association for Computational Linguistics.