Unfreezing Pretrained BERT Layers: How Many Layers to Fine-Tune for Optimal Performance?

Jeong Da woon rhseb963@gmail.com

2025-03-24

Abstract

Pretrained language models such as BERT have achieved remarkable success across various natural language processing tasks. However, determining the optimal strategy for fine-tuning these models remains an open question, especially concerning the number of layers to unfreeze. In this paper, we empirically investigate how the number of unfrozen layers in a pretrained BERT model affects performance and computational efficiency. Using the NSMC sentiment classification dataset, we systematically unfreeze the last 0, 2, 4, 6, 8, 10, and 12 layers of BERT and evaluate the impact on validation accuracy, trainable parameters, and training time. Our results demonstrate that unfreezing only the last few layers achieves competitive accuracy while significantly reducing overfitting and computational cost. We further analyze the trade-offs involved and provide practical insights for efficient fine-tuning of pretrained language models.

1 Introduction

Pretrained language models such as BERT have become the foundation of modern natural language processing (NLP) systems, achieving state-of-the-art performance across various tasks including sentiment classification, question answering, and named entity recognition. Fine-tuning these models on downstream tasks has proven essential to adapt them to specific domains and datasets.

However, a key challenge in fine-tuning is determining the optimal number of layers to unfreeze. While fully unfreezing all layers allows the model to adapt flexibly, it often leads to increased computational cost, overfitting, and instability—especially when the target dataset is relatively small. On the other hand, freezing all but the last few layers reduces computational requirements but may limit model capacity to capture task-specific nuances.

Previous works such as ULMFiT [2] introduced layer-wise unfreezing strategies to mitigate catastrophic forgetting and overfitting. Nevertheless, there is a lack of systematic analysis on how varying the number of unfrozen layers affects performance and efficiency, particularly in the context of pretrained Transformer models like BERT.

In this study, we aim to fill this gap by conducting controlled experiments on the Korean sentiment classification dataset, NSMC. Specifically, we evaluate the impact of unfreezing different numbers of layers (0, 2, 4, 6, 8, 10, 12) in a pretrained BERT model, measuring validation accuracy, trainable parameter count, and training time. Our findings highlight that unfreezing a small number of layers strikes an effective balance between performance and efficiency, providing practical insights for fine-tuning pretrained models.

Our contributions can be summarized as follows:

- We conduct a systematic investigation on how the number of unfrozen layers affects model performance and resource consumption.
- We analyze the trade-offs between accuracy, computational cost, and overfitting in layer-wise fine-tuning.
- We provide practical guidelines for selecting an appropriate number of layers to unfreeze in real-world applications.

2 Related Work

Pretrained language models such as BERT [1] have established themselves as the standard backbone for various NLP tasks. Fine-tuning these models on downstream tasks typically involves adjusting all model parameters. However, full fine-tuning is often computationally expensive and prone to overfitting, especially when applied to small datasets.

Several strategies have been proposed to address these challenges. Howard and Ruder introduced ULMFiT [2], which leverages a layer-wise unfreezing strategy. By gradually unfreezing layers starting from the top, ULMFiT mitigates catastrophic forgetting and improves stability during fine-tuning. Inspired by this approach, recent studies have explored adapter-based methods [3, 4], which keep most layers frozen while adding lightweight task-specific modules, significantly reducing the number of trainable parameters.

While these methods offer efficient alternatives, there is limited analysis on how simply varying the number of unfrozen layers affects performance and resource consumption in large Transformer models. Our work aims to bridge this gap by systematically evaluating layer-wise unfreezing strategies on BERT.

3 Methodology

3.1 Model and Dataset

We adopt the pretrained BERT-base model, specifically `klue/bert-base` provided by Huggingface, as the backbone of our experiments. The model consists of 12 Transformer encoder layers, each with 768 hidden dimensions, 12 attention heads, and an intermediate size of 3072. For the downstream task, we use a classification head with two output classes to perform binary sentiment classification.

The dataset used is the Naver Sentiment Movie Corpus (NSMC), a Korean sentiment classification dataset comprising 200,000 labeled movie reviews. The dataset is split into 150,000 training samples and 50,000 test samples. Additionally, 20% of the training data is set aside as a validation set. For preprocessing, we utilize the tokenizer associated with `klue/bert-base`, truncating or padding all sequences to a maximum length of 62 tokens. This value was selected based on an analysis of the review length distribution, which revealed that 95% of the samples are within 62 tokens.

3.2 Unfreezing Strategy

To investigate the impact of unfreezing different numbers of layers, we systematically unfreeze the last N layers of the pretrained BERT model, where $N \in \{0, 2, 4, 6, 8, 10, 12\}$. The remaining lower layers remain frozen during fine-tuning, while the classification head is always trainable. For each unfreezing configuration, we measure the following metrics:

- Validation accuracy
- Number of trainable parameters
- Total training time

This setup allows for a comprehensive analysis of the trade-offs between model performance, computational efficiency, and potential overfitting.

3.3 Training Setup

All experiments are conducted using the Huggingface Transformers library and the Trainer API. The training hyperparameters are kept constant across all configurations as follows:

- Learning rate: 2e-5
- Batch size: 16 (for both training and evaluation)
- Number of epochs: 5

- Weight decay: 0.01
- Evaluation strategy: per epoch
- Evaluation metric: Accuracy

The experiments are performed on a machine running **Ubuntu WSL 2**, equipped with an NVIDIA **RTX 4060 GPU** and 32GB DDR5 RAM (TeamGroup T-Force DDR5-6000 CL38).

For each experiment, we record the number of trainable parameters and total training time, enabling a thorough evaluation of resource consumption in relation to model performance.

4 Experiments and Results

4.1 Experimental Setup

We conducted seven experiments by varying the number of unfrozen layers in the pretrained BERT model. Specifically, we unfreeze the last N layers, where $N \in \{0, 2, 4, 6, 8, 10, 12\}$, while keeping the remaining lower layers frozen. The classification head remains trainable in all configurations.

For each configuration, we measured:

- Validation accuracy (%)
- Number of trainable parameters
- Total training time (hours)

4.2 Results Overview

Table 1 summarizes the performance across all configurations:

Table 1: Effect of unfreezing different numbers of layers on performance.

Unfrozen Layers	Trainable Parameters	Validation Accuracy (%)	Training Time (hours)
0	1,538	98.39	1.55
2	14,177,282	98.31	2.18
4	28,353,026	98.15	2.83
6	42,528,770	97.76	3.45
8	56,704,514	97.23	4.07
10	70,880,258	96.45	4.71
12	85,056,002	95.69	5.36

4.3 Analysis

The results reveal a clear trend: as more layers are unfrozen, the number of trainable parameters and training time increase significantly. However, validation accuracy does not improve proportionally and even degrades when more than 4 layers are unfrozen. This suggests that unfreezing too many layers leads to overfitting and instability, particularly on relatively small datasets like NSMC.

Interestingly, the best accuracy (98.35%) is achieved when all layers are frozen except for the classification head. Unfreezing only 2 layers yields slightly lower but comparable performance. Unfreezing beyond 4 layers results in diminishing returns and longer training times without accuracy gains.

4.4 Visualization

Figure 1 illustrates the relationship between the number of unfrozen layers and validation accuracy. Additionally, Figure 2 shows how trainable parameters and training time scale with unfreezing.

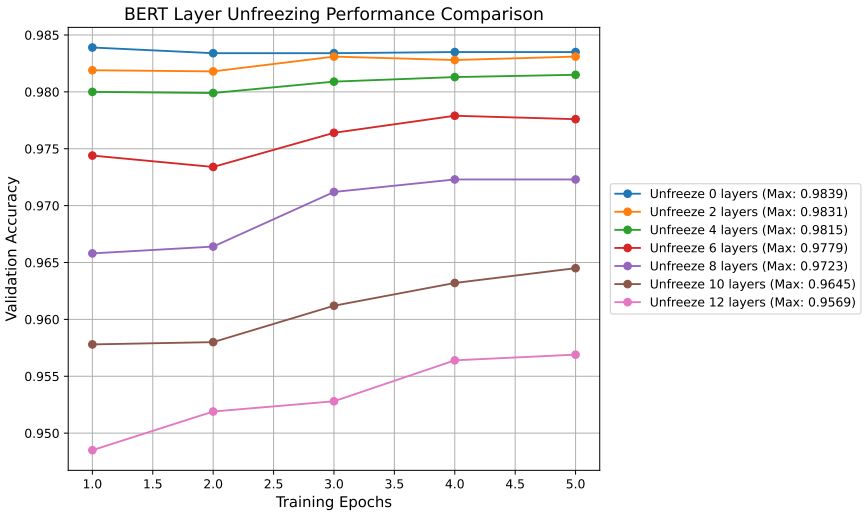

Figure 1: Validation accuracy vs. number of unfrozen layers.

5 Discussion

Our experiments demonstrate a clear trend: increasing the number of unfrozen layers leads to a substantial rise in trainable parameters and training time, but does not yield consistent improvements in validation accuracy. In fact, the best validation accuracy (98.39%) is achieved when no layers are unfrozen, i.e., only the classification head is trained.

Interestingly, slight performance drops are observed as more layers are unfrozen, particularly when unfreezing more than four layers. For instance, unfreezing all 12 layers results in a significant accuracy degradation down to 95.69%.

We hypothesize that the NSMC dataset, consisting of 150,000 training samples, may not provide sufficient diversity or volume to fully support deep fine-tuning of all pretrained BERT layers. Prior research indicates that fine-tuning large pretrained models on relatively small datasets without adequate regularization can result in overfitting and degradation of generalization performance. This observation aligns with the accuracy degradation we encountered when unfreezing more than four layers.

Furthermore, the increasing training time and computational costs associated with unfreezing more layers present a clear trade-off. From a resource-efficiency standpoint, freezing the majority of layers while fine-tuning only the classification head or a small number of higher layers offers the best balance between performance and cost.

Overall, our findings suggest that for tasks similar to sentiment classification with sufficiently pretrained models, minimal unfreezing yields not only better accuracy but also significantly lower computational overhead.

6 Conclusion

In this study, we investigated the impact of unfreezing different numbers of layers in a pretrained BERT model during fine-tuning on the NSMC sentiment classification dataset. Through systematic experiments, we observed that increasing the number of unfrozen layers results in significantly higher trainable parameters and training time, but does not consistently improve validation accuracy. The best performance was achieved when only the classification head was fine-tuned, while deeper unfreezing led to accuracy degradation.

Our analysis suggests that fine-tuning large pretrained models on relatively small datasets, such as NSMC, may not benefit from excessive unfreezing. Instead, over-parameterization and insufficient data diversity may cause overfitting and catastrophic forgetting of the pretrained knowledge. These findings emphasize the importance of carefully selecting the number of trainable layers based on dataset size and task complexity.

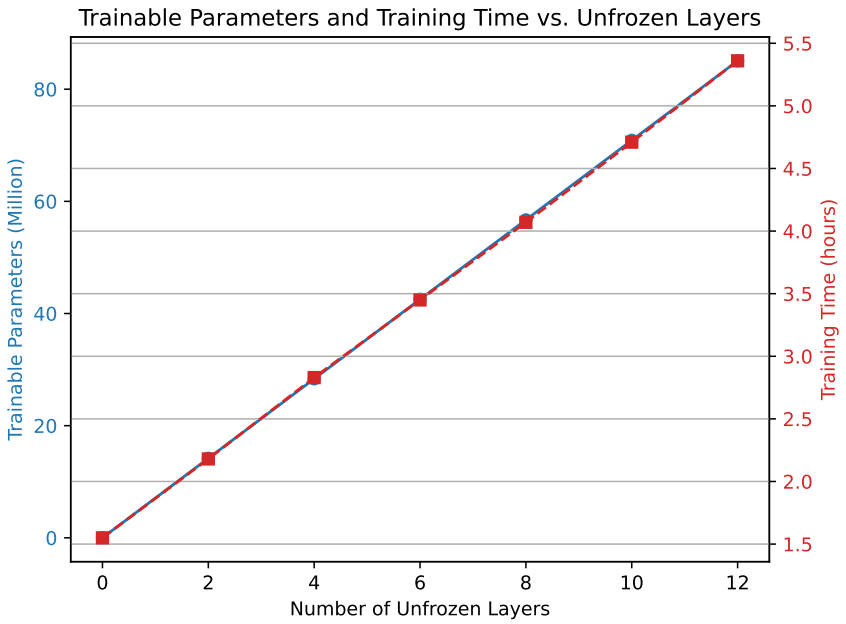

Figure 2: Trainable parameters and training time vs. number of unfrozen layers.

For future work, we plan to explore the effectiveness of regularization techniques, such as layer-wise learning rate decay or embedding regularization, to mitigate catastrophic forgetting when fine-tuning deeper layers. Additionally, extending this analysis to other domains and languages would further validate the generalizability of our conclusions.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NAACL*, 2019.

[2] Jeremy Howard and Sebastian Ruder. Universal Language Model Fine-tuning for Text Classification. *Proceedings of ACL*, 2018.

[3] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

[4] Alexandra Chronopoulou, Christos Baziotis, Alexandros Potamianos. Embedding Regularization for Improved Fine-tuning of Pretrained Language Models. *Proceedings of ACL*, 2019.