Prompting language models improves performance in imbalanced setting

Jay Mohta
Amazon
jaymoht@amazon.com

Abstract

Prompting is a widely adopted technique for fine-tuning large language models. Recent research by Scao and Rush (2021) has demonstrated its effectiveness in improving few-shot learning performance compared to vanilla fine-tuning and also showed that prompting and vanilla fine tuning achieves similar performance in high data regime ($\sim>$ 2000 samples). This paper investigates the impact of imbalanced data distribution on prompting. Through rigorous experimentation on diverse datasets and models, our findings reveals that even in scenarios with high data regimes, prompting consistently outperforms vanilla fine-tuning by exhibiting average performance improvement of $2-5\%$.

1 Introduction

Fine tuning language models is a common strategy in Natural Language Processing (NLP), where a classifier head is added on top of the base language model to obtain the desired classification output. This approach has been applied to various NLP models, including RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019), and DeBERTa (He et al., 2020), and has demonstrated exceptional performance on benchmark datasets such as GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019).

An alternate approach to adapting language models to downstream tasks involves the use of autoregressive text generation or prompt based fine tuning. This technique is commonly used in sequence-to-sequence models such as T5 (Raffel et al., 2019) leading to state-of-the-art performance on SuperGLUE benchmark. This type of fine tuning strategy has an added advantage of multi-task training (McCann et al., 2018). This technique has also shown to improve models zero shot capability (Puri and Catanzaro, 2019) where we can provide only task description and model is able to classify the input correctly.

Work by Scao and Rush (2021); Schick and Schütze (2020); Webson and Pavlick (2021) has shown that prompting language models really helps in few shot learning setting over vanilla fine tuned models. In high data regime setting prompting and vanilla fine tuned language models achieve the similar performance. However, these studies used balanced datasets where the number of examples from each class are equal.

The issue of class imbalance in machine learning is a well-known challenge, and occurs when the distribution of samples across classes is skewed. These types of problems are encountered in various real world settings like malware detection (Demirkiran et al., 2021), spam detection (Rao et al., 2023), medical domain (Altaf et al., 2023) and many more. Previous work by Buda et al. (2017); Leevy et al. (2018) has shown that if we use general supervised loss then it leads to poor generalization on the minority classes. In this work we ask the question: How does prompting impact performance in imbalanced setting? To the best of our knowledge this is the first work which explores impact of prompting in imbalanced setting.

In this work therefore we conduct an experimental study by varying imbalance ratio and compare performance of vanilla fine tuned model with that of prompting based models. Our study includes experiments with varying models like RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019) and DeBERTa (He et al., 2020), and different datasets like RTE (Dagan et al., 2007), QQP (Chen et al., 2017) and MRPC (Lan et al., 2017). To study how different imbalance ratios affect performance we vary imbalance ratio from 0.1%-30%. We also compare the impact of model size on the performance of models in imbalanced settings i.e. for ALBERT we run experiments on large and its base counterpart. To make our finding more robust we experiment across different prompts as work by Webson and Pavlick (2021) points out that different

Proceedings of The Fourth Workshop on Simple and Efficient Natural Language Processing (SustaiNLP), pages 201–211

prompts could impact performance of the models. To isolate the impact of prompting we don't use any special technique like PET (Schick and Schütze, 2021), AdaPET (Tam et al., 2021) for performing fine tuning. We suspect that using those techniques may further improve performance of prompt based models.

Our results show that prompting helps in imbalanced setting over vanilla fine tuning in mild imbalanced setting even in high data regime by $2-5\%$ increase in performance on average. In high and no imbalanced setting the prompting and vanilla fine tuning gives a very similar performance. This insight will help practitioners decide what fine tuning strategy works the best for their use case.

The rest of the paper is organized as follows section 2 will provide some background on vanilla fine tuning and prompting. section 3 describes our experimental setup and results. We conclude in section 4 with a summary of our findings and suggestions for future work.

2 Background

The aim of this work is to evaluate the impact of prompting on language model performance compared to traditional fine tuning. To achieve this, we conduct experiments with different imbalance ratios from severe to mild to low/no imbalance. The following sections provides background on vanilla fine tuning, prompting based fine tuning and imbalanced classification problems before delving into our empirical study.

2.1 Vanilla fine tuning

This is very simple and widely adopted fine tuning stratgey where we add classifier head on the top of language models. In the case of RoBERTa, ALBERT and DeBERTa the classification head is added on top of $[CLS]$ token and the embedding generated for that token are fed into this classification head to generate the classification output.

2.2 Prompt based fine tuning

Prompting is a fine tuning technique that utilizes masked language modeling to obtain the classification output, converting each classification task into sequence-to-sequence problem. Similar to PET (Schick and Schütze, 2021), the prompt is decomposed into two parts: the *pattern* and the *verbalizer*. The pattern transforms the input into clozed task, i.e., a sequence with a $mask$ token that needs to be filled by the model, serving as the classification output. The verbalizer then converts the output space into a token or sequence of tokens in the vocabulary. The goal of prompting is to guide the model by providing a pattern that contains the $mask$ token, and for the model to predict the correct output based on the defined verbalizer pattern.

To illustrate the technique of prompting, consider an example from the RTE dataset (Dagan et al., 2007). The task is to predict whether the premise *No Weapons of Mass Destruction Found in Iraq Yet.* entails the hypothesis, *Weapons of Mass Destruction Found in Iraq*. The prompt is generated using the verbalizer pattern that maps entailment to *yes* and non-entailment to *no*. The prompt pattern is defined as follows

> ***Given*** No Weapons of Mass Destruction Found in Iraq Yet. ***Should we assume that*** Weapons of Mass Destruction Found in Iraq ***mask***

The bolded text represents the prompt pattern, while the non-bolded text is the sample from RTE dataset. The model predict yes or no at the $mask$ token based on the verbalizer pattern we defined. Different pattern-verbalizer pattern can be used for single task and prior work (Webson and Pavlick, 2021; Brown et al., 2020) has shown that different choices of prompt pattern and verbalizer pairs can impact the performance of the model. To ensure robust results, we experiment with various prompt pattern-verbalizer pairs.

2.3 Class imbalance

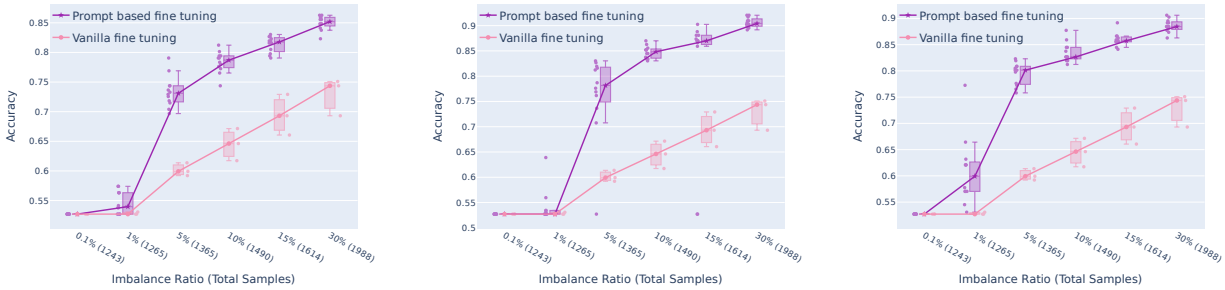
In this work we define something called imbalance ratio which represents how imbalanced is your train set. It is defined as follows

$$\text{Imbalance Ratio} = \frac{\text{Number of negative samples}}{\text{Total number of samples}} \tag{1}$$

In order to effectively study prompting based technique in class imbalance we start from an imbalance ratio of 0.1%, we incrementally increase the imbalance ratio up to 30%, allowing us to study the effect of various levels of class imbalance.

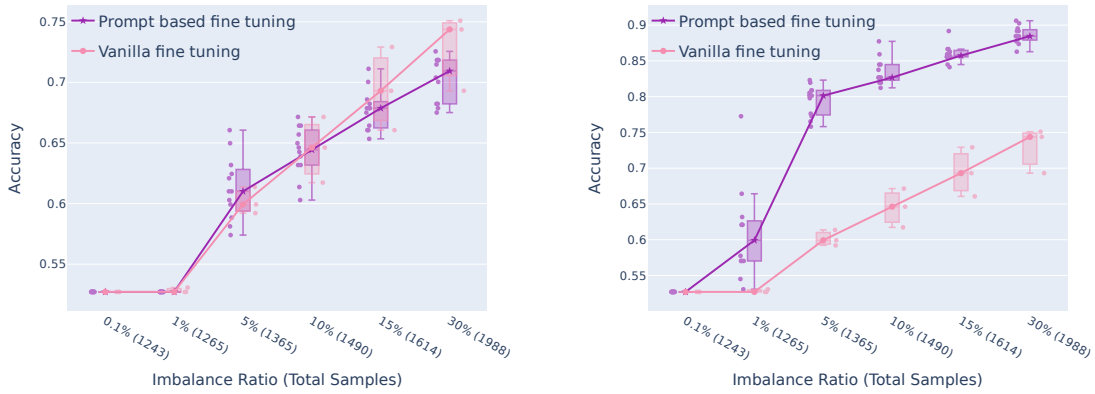
3 Experimental setting and results

We now present our main experimental results to show that prompting improves performance of the



(a) RoBERTa (b) DeBERTa (c) ALBERT

Figure 1: These figures are similar to figures plotted in Webson and Pavlick (2021). Here each dot represents one prompt under one random seed (random seed controls different negative examples selected to create an imbalance). The plot compares fine tuning and prompt based tuning performance with varying imbalance ratios on RTE dataset (reported accuracies are on validation set). The boxes span from first quartile to third quartile while the lines inside the box mark the median.



(a) ALBERT-Base (b) ALBERT-Large

Figure 2: Comparing performance of ALBERT-Base with ALBERT-Large on RTE dataset.

model than vanilla fine tuned model in imbalanced setting (even in high data regime). To improve the robustness of our results we experiment with different models, datasets and different training splits. In the following subsection we describe the setup and main takeways from the experiments.

3.1 Setup

We experimented with RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019) and DeBERTa (He et al., 2020) pre-trained models. These are all encoder-only models trained using masked language modelling objective during pretraining phase. We experimented with different prompts from the open-source library *prompt-source* (Bach et al., 2022) to understand the impact of different prompts on performance. We provide experimental results on three different datasets Recognizing Textual Entailment (RTE) (Dagan et al., 2007), Quora Question Pairs (QQP) (Chen et al., 2017) and Microsoft Research Paraphrase Corpus (MPRC) (Lan et al., 2017). Except QQP which has $>$ 100k samples all other datasets have about 2400 samples. In order to check how prompting affect performance in imbalanced setting we experiment with different imbalance ratios defined in eq. (1). We start from as low as 0.1% imbalance ratio and incrementally increase it up to 30%. To ensure the reliability of our results, we conduct multiple experiments by varying the random seed three times which is used for selecting a new subset of samples from the training set. On each downstream task we fine tune RoBERTa, ALBERT and DeBERTa using prompting based fine tuning and vanilla fine tuning with varying prompts and varying seeds. For all of our prompt based fine tuning experiments we use a

learning rate of $1e-5$ and we train the model for 5 epochs. For all of our vanilla fine tuning experiments we use learning rate of $2e-5$ and train the model for 5 epochs as well. In the main text of the paper we provide results on RTE dataset. We ask the readers to refer to Appendix for results on all datasets. We also provide different prompts used for different datasets in appendix A.

3.2 Prompting improves performance in imabalanced setting

The results of our experiments are depicted in fig. 1. Our findings demonstrate that in high data regime and imbalanced settings, prompt-based fine tuning consistently outperforms vanilla fine tuning. In scenarios where the imbalance ratio is between 0.1% and 1%, both prompt-based and vanilla models perform similarly, almost equivalent to predicting the more labels class. However, when the imbalance ratio is between 5% and 15%, we observe significant improvement in the performance of prompt-based models compared to vanilla fine tuning. Especially, for RTE dataset we observe $10-15\%$ improvement in performance across different models. The difference in performance between the two methods becomes smaller at 30% imbalance ratio. As stated by previous studies (Brown et al., 2020; Webson and Pavlick, 2021), in balanced high data regimes, the performance of prompt-based models becomes similar to vanilla fine tuning. For more comprehensive results obtained from various datasets and models, please refer to appendix B. Overall, our findings indicates that when dealing with an imbalance ratio ranging from 5% to 15% there is an average improvement in performance of approximately $2-5\%$.

As shown in fig. 2, the comparison of the performance between the prompted model and ALBERT-Base and Large reveals that using the base models of these models does not significantly improve performance. Both the vanilla fine-tuned model and the prompt-based fine-tuned model yield similar results. This finding aligns with previous studies such as (Schick and Schütze, 2021; Tam et al., 2021), which also noted that prompted base models (or smaller models) do not enhance performance in the few-shot learning setting. The same holds true for imbalanced settings, as indicated by our results. For further analysis of different model sizes, please refer to appendix C in the paper.

4 Conclusion

This paper investigated the impact of prompt-based fine-tuning and vanilla fine-tuning on the performance of models in high data regimes and imbalanced settings. The findings revealed that prompt-based fine-tuning outperforms vanilla fine-tuning by about $2-5\%$, particularly in scenarios where the imbalance ratio is between 5% to 15%. The results in balanced high data regimes were in accordance with previous studies, showing that prompt-based models perform similarly to vanilla fine-tuning. A comparison between the prompted base models and large models found that the former did not provide significant improvement in performance. To explain these phenomenons we aim to further study the pretraining dataset on the performance the models, as the output distribution of masked language modelling may play a role in the enhanced performance of prompt-based models compared to vanilla fine-tuned models.

References

Fouzia Altaf, Syed M. S. Islam, Naeem Khalid Janjua, and Naveed Akhtar. 2023. Pre-text representation transfer for deep learning with limited imbalanced data : Application to ct-based covid-19 detection. *ArXiv*, abs/2301.08888.

Stephen H. Bach, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, Zaid Alyafeai, Manan Dey, Andrea Santilli, Zhiqing Sun, Srulik Ben-David, Canwen Xu, Gunjan Chhablani, Han Wang, Jason Alan Fries, Maged S. Al-shaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Xiangru Tang, Mike Tian-Jian Jiang, and Alexander M. Rush. 2022. Promptsource: An integrated development environment and repository for natural language prompts.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *ArXiv*, abs/2005.14165.

Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. 2017. A systematic study of the

class imbalance problem in convolutional neural networks. *Neural networks : the official journal of the International Neural Network Society*, 106:249–259.

Zihang Chen, Hongbo Zhang, Xiaoji Zhang, and Leqi Zhao. 2017. Quora question pairs.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2007. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*.

Ferhat Demirkiran, Aykut Çayir, Uğur Ünal, and Hasan Dag. 2021. An ensemble of pre-trained transformer models for imbalanced multiclass malware classification. *Comput. Secur.*, 121:102846.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *ArXiv*, abs/2006.03654.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. *arXiv preprint arXiv:1708.00391*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *ArXiv*, abs/1909.11942.

Joffrey L. Leevy, Taghi M. Khoshgoftaar, Richard A. Bauder, and Naeem Seliya. 2018. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5:1–30.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Raul Puri and Bryan Catanzaro. 2019. Zero-shot text classification with generative language models. *arXiv preprint arXiv:1912.10165*.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv*, abs/1910.10683.

Sanjeev Rao, Anil Kumar Verma, and Tarunpreet Bhatia. 2023. Hybrid ensemble framework with self-attention mechanism for social spam detection on imbalanced data. *Expert Systems with Applications*.

Teven Le Scao and Alexander M. Rush. 2021. How many data points is a prompt worth? In *North American Chapter of the Association for Computational Linguistics*.

Timo Schick and Hinrich Schütze. 2020. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*.

Timo Schick and Hinrich Schütze. 2021. True few-shot learning with prompts—a real-world perspective. *Transactions of the Association for Computational Linguistics*, 10:716–731.

Derek Tam, Rakesh R Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. 2021. Improving and simplifying pattern exploiting training. In *Conference on Empirical Methods in Natural Language Processing*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *ArXiv*, abs/1905.00537.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP@EMNLP*.

Albert Webson and Ellie Pavlick. 2021. Do prompt-based models really understand the meaning of their prompts? *ArXiv*, abs/2109.01247.

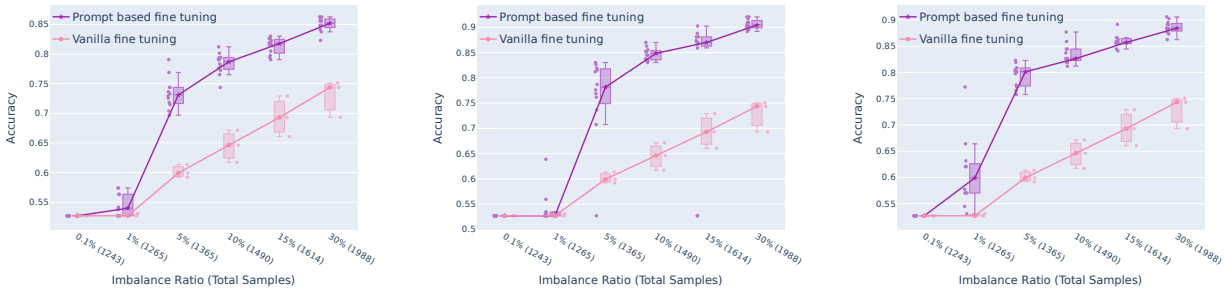
A Different prompts used for different datasets

This section describes the different prompts and verbalizer patterns used for the experiments.

Dataset	Patterns	Verbalizer
RTE	Given {premise} Should we assume that {hypothesis} is true?	yes-no
	{premise} Based on the previous passage, is it true that {hypothesis}?	yes-no
	Given {premise} Is it guaranteed true that {hypothesis}?	yes-no
	Suppose {premise} Can we infer that {hypothesis}?	yes-no
QQP	Can an answer to {question1} also be used to answer {question2}?	yes-no
	I received the questions {question1} and {question2}. Are they duplicates?	yes-no
	Are the questions {question1} and {question2} asking the same thing?	yes-no
	I am an administrator on the website Quora. There are two posts, one that asks {question1} and another that asks {question2}. I can merge questions if they are asking the same thing. Can I merge these two questions?	yes-no
MRPC	Are the following two sentences equivalent? {sentence1}. {sentence2}	yes-no
	I want to know whether the following two sentences mean the same thing. {sentence1}. {sentence2}	yes-no
	Do the following two sentences mean the same thing? {sentence1}. {sentence2}	yes-no
	Can I replace the sentence {sentence1} with the sentence {sentence2} and have it mean the same thing?	yes-no

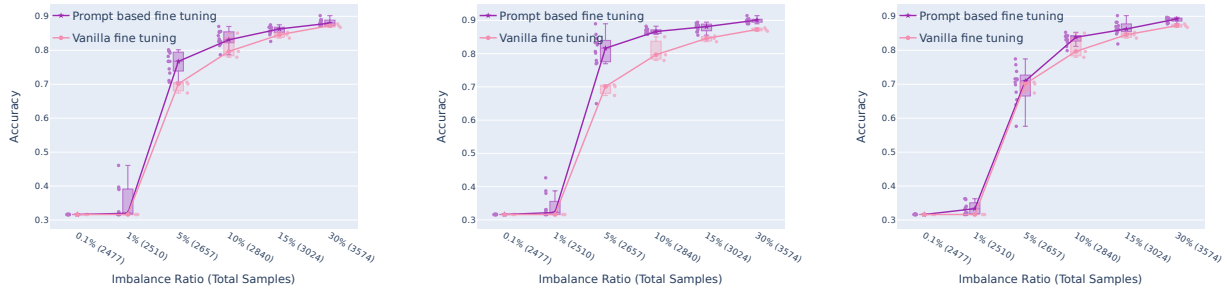
Table 1: Table showing different Datasets, Patterns, and Verbalizers.

B Prompt based fine tuning vs vanilla fine tuning on different datasets and models



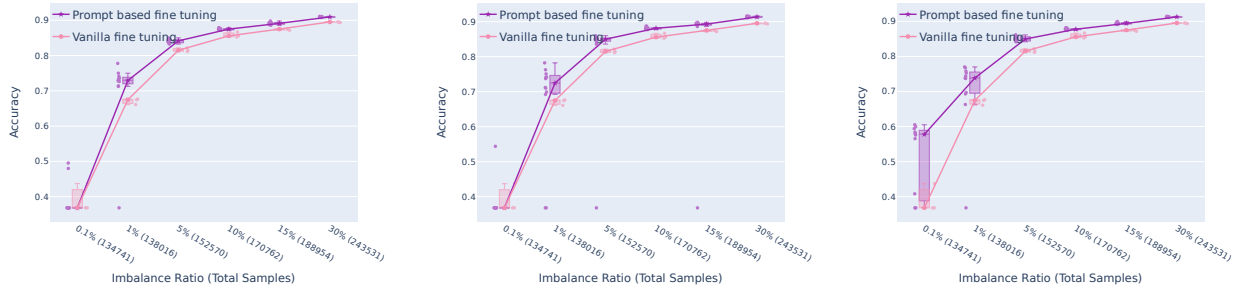
(a) RoBERTa (b) DeBERTa (c) ALBERT

Figure 3: RTE dataset performance on different models



(a) RoBERTa (b) DeBERTa (c) ALBERT

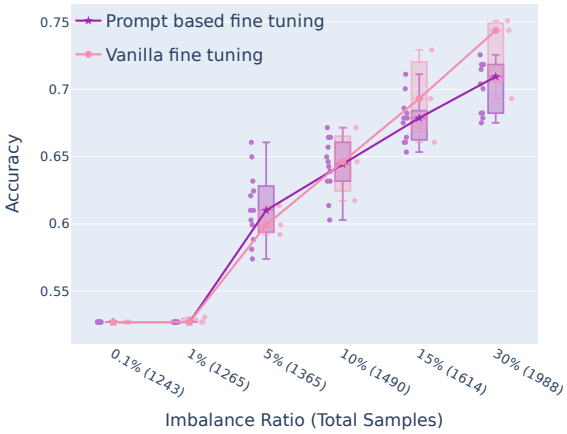
Figure 4: MRPC dataset performance on different models



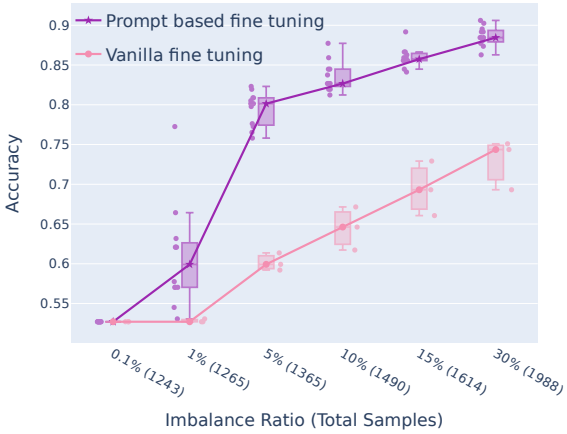
(a) RoBERTa (b) DeBERTa (c) ALBERT

Figure 5: QQP dataset performance on different models

C Large vs base model comparison on different datasets

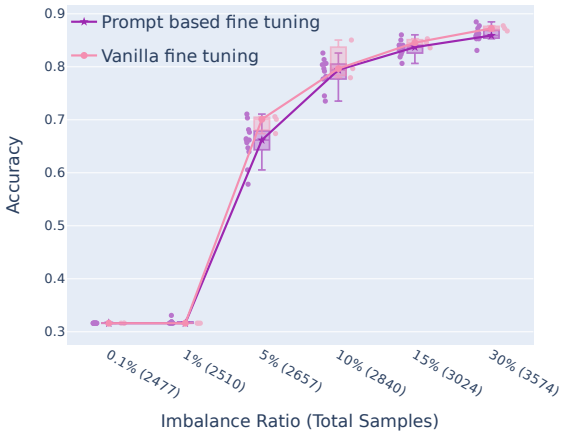


(a) ALBERT-Base

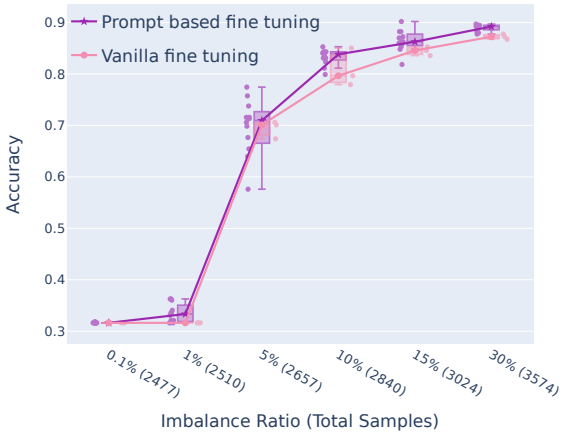


(b) ALBERT-Large

Figure 6: Comparing performance of ALBERT-Base with ALBERT-Large on RTE dataset

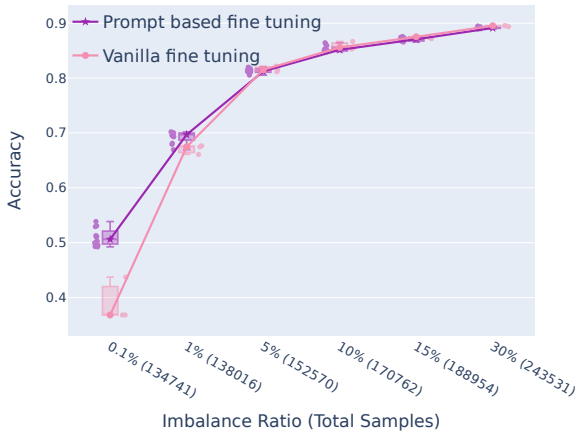


(a) ALBERT-Base

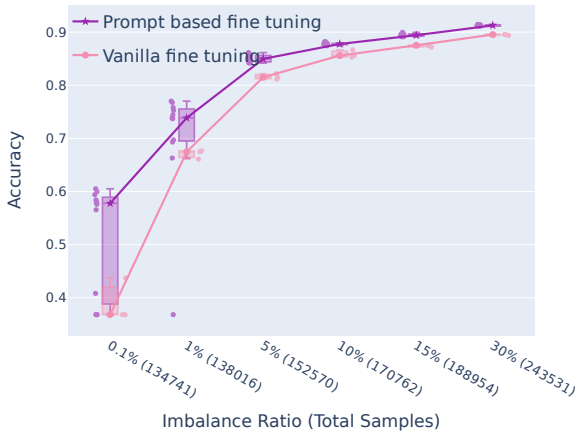


(b) ALBERT-Large

Figure 7: Comparing performance of ALBERT-Base with ALBERT-Large on MRPC dataset



(a) ALBERT-Base



(b) ALBERT-Large

Figure 8: Comparing performance of ALBERT-Base with ALBERT-Large on QQP dataset