Qwen-2.5-0.5B在Alpaca数据集上的全量微调

刘翰文
522030910109

Abstract

模型微调通常是在某一特定数据集上进一步训练预训练模型，使模型在特定领域上取得更好的理解和执行任务的性能。随着近年来大语言模型的快速发展，对大语言模型的微调自然成为了提升模型性能的一种简单而又有效的方法。我们使用了Qwen-2.5-0.5B模型，并在Alpaca指令数据集上进行了全量微调，随后在一系列测评数据集上对微调前后的模型进行了评估，最后，我们通过评估结果和模型输出分析总结了此次微调的效果。

1 Introduction

近年来，随着深度学习的不断发展，越来越多的模型(Elman, 1990; Gers et al., 2000; Vaswani, 2017; Cho et al., 2014)被提出用来解决自然语言处理的问题，其中大语言模型(Devlin, 2018; Brown et al., 2020; Touvron et al., 2023; Chowdhery et al., 2023)的快速发展更给自然语言处理领域带来了前所未有的变革。它们凭借超大规模参数和丰富的知识语料，展现了卓越的文本理解与生成能力，已成为许多实际应用的核心技术支撑。然而，在涵盖了各个领域的大型数据集上进行训练的后果则是难以在特定领域上取得卓越的效果。因此，模型性能的进一步提升往往依赖于针对特定任务或数据集的精细化训练，模型微调作为一种经典方法，包括PEFT(Ding et al., 2023)、LoRA(Hu et al., 2021)和IT(Zhang et al., 2023)等，通过模型在特定数据集上的进一步训练，使模型能够在对特定任务的理解和执行上获得显著的提升。

在本次作业中，我们使用拥有较少参数量的Qwen-2.5-0.5B(Hui et al., 2024)作为基底模型，使用指令数据集Alpaca(Taori et al., 2023)进行全量微调。在报告中，我们首先说明了进行实验所使用的平台设备以及训练参数的设置，随后我们展示了微调前后模型在一系列测评数据集上的结果，最后我们结合模型的测评结果和部分输出分析了模型在微调前后的性能变化。

2 Experiments

在本节中，我们详细的说明了实验的设置并展示分析了模型在微调前后的性能。在2.1节中，我们介绍了实验所使用的平台设备、微调参数和微调流程等具体实现模型微调的细节。在2.2节中，我们展示了微调前后模型在一系列测评数据集上的表现，并进行了对比，同时展示了微调前后模型对同一输入的不同输出。2.3节中，我们结合微调数据集和测评数据集的特征分析了2.2节中展示的结果，并结合模型的具体输出对模型的性能变化进行分析。

2.1 Implementation Details

在本次对Qwen-2.5-0.5B模型进行微调的实验中，我们使用了Kaggle平台并借助Kaggle平台提供的免费算力GPU T4 × 2进行实验。我们根据使用的GPU和微调任务的特点修改了部分训练参数，以便能够在16G显存限制下得到较好的实验效果。首先，我们将训练批次大小设置为1，最大序列长度设置为1024，并将模型参数的精度设置为bfloat16，以最大程度地减小对显存的需求，提高微调过程的稳定性。在训练设置上，我们将优化器设置为AdamW(Loshchilov, 2017)，并以学习率为1e-5，权重衰减为1e-3，在经过0.03的预热后对学习率进行余弦退火，受限于kaggle算力，我们将模型在指令数据集训练30000个steps后得到微调后的模型。同时，我们观察到模型的训练损失早在30000steps之前就几乎不再下降，故我们的结果应与训练完的结果相同。其他具体的训练参数设置见代码。在对模型的评测上，我们同样采取了大小为1的训练批次大小和1024的最大序列，对微调前后的模型在MMLU(Hendrycks et al., 2020)、HellaSwag(Zellers et al., 2019)和ARC(Clark et al., 2018)等评测数据集上进行评估。

2.2 Comparisons with Pre-fine-tuned Model

在本节中，我们使用了一系列评测数据集对

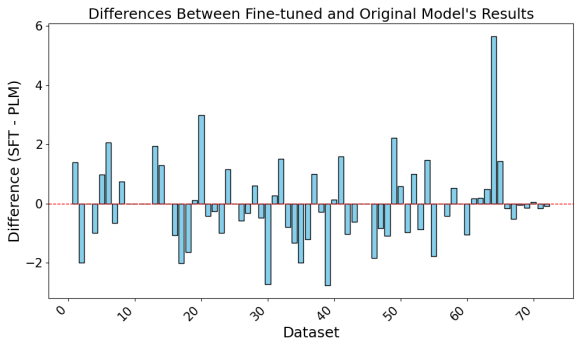

Figure 1: 微调前后模型评测结果：横坐标为数据集在结果文件中的索引，纵坐标为微调后模型在该数据集上的结果减去微调前模型在相同数据集上的结果(%)

微调前后模型的性能进行评估，范围涵盖了数学、物理、历史等众多领域，模型被要求从提供的选项中选择出尽可能满足要求的选项，并以选择的准确率作为评测标准。图1和表格1展示了我们对模型进行评测的结果。由于测评数据集的数量过多，表格1只展示了模型在部分出现显著变化的数据集上的表现，完整的结果见上传的结果文件。在所有的72个评测数据集中，微调前的模型在35个数据集上的表现优于原模型，在11个数据集上的表现与微调后的模型持平，在剩下的26个数据集上的表现逊于微调后的模型。我们还计算了模型在72个数据集上的平均准确率，微调前模型的平均准确率为49.17%，微调后模型的平均准确率为49.13%，微调前模型的平均准确率比微调后模型的准确率高出了0.04%，基本持平。初次之外，我们还尝试分别使用微调前后的模型对同一句话生成回复。例如，我们将“Introduce Natural Language Processing(NLP).”嵌入到训练模板中作为输入，对模型的输出解码后得到目标输出。对比两个模型输出，可以明显的感受到微调后的模型可以输出更长、更详细的信息。

此外，我们还比较了在自回归训练时，只算Output部分的损失算整个序列的损失的区别。结果显示，计算整个序列的损失时，在训练过程中损失收敛于1.13，而只算Output部分的损失时，损失在训练过程中收敛于1.23，比计算整个序列时的损失高出0.1。推测是由于Output部分的预测难度高于instruction和input部分，因为在大多数情况下，instruction和input为固定和结构化的输入。

2.3 Analysis of Results

在本节中，我们结合微调数据集Alpaca和测评数据集的特征对模型在微调前后的评测结果进行了分析。在2.3.1节中，我们介绍了微调数据集和部分评测数据集的特征，在2.3.2节中，我们根据数据集的特征和模型的输出对结果做了一个可能的解释。

输入：Introduce Natural Language Processing(NLP).
模型微调前：Natural Language Processing(NLP) is a subfield of artificial intelligence that deals with the interaction between computers and human language. It involves the use of computer systems to understand, interpret, and generate human language. NLP is used in various applications such as text analysis, speech recognition, machine translation, and sentiment analysis. It is a complex field that requires a deep understanding of natural language processing concepts and techniques.

输入：Introduce Natural Language Processing(NLP).
模型微调后：Natural Language Processing (NLP) is a subfield of artificial intelligence that deals with the interaction between computers and humans. It involves the use of computer algorithms to understand, interpret, and generate human language. NLP is used in a wide range of applications, including speech recognition, text analysis, machine translation, and sentiment analysis. It is a complex field that involves several techniques, including machine learning, natural language processing, and natural language generation. NLP is an important tool for improving the efficiency and accuracy of human-computer interactions, and it has the potential to revolutionize many industries, including healthcare, finance, and education.

2.3.1 Characteristics of the Dataset

对于指令数据集Alpaca，它包括52K由Instruction、Input和Output三部分组合而成的指令数据。对于测评数据集，我们以MMLU和HellaSwag数据集为例，MMLU数据集收集了数学、历史、物理、计算机科学等57个学科、不同难度的选择题，涵盖了非常广阔的知识面，旨在全面评估模型的在知识掌握、推理和问题解决方面的能力。HellaSwag数据集将重点放在模型的上下文感知和常识推理的能力，提出的问题通常需要对上下文的深入理解，而不仅仅是对单词和词语的简单匹配。

Dataset	Metric	Mode	Model	
			Original	Fine-tuned
MMLU College Biology	Accuracy	ppl	43.75	**45.14**
MMLU College Chemistry	Accuracy	ppl	**43.00**	41.00
MMLU Electrical Engineering	Accuracy	ppl	53.10	**55.17**
MMLU Management	Accuracy	ppl	63.11	**65.05**
MMLU Nutrition	Accuracy	ppl	57.52	**58.82**
MMLU Professional Accounting	Accuracy	ppl	**38.30**	37.23
MMLU High School Geography	Accuracy	ppl	**60.10**	58.08
MMLU International Law	Accuracy	ppl	**69.42**	67.77
MMLU Computer Security	Accuracy	ppl	69.00	**72.00**
MMLU Formal Logic	Accuracy	ppl	35.71	**37.30**
MMLU High School Mathematics	Accuracy	ppl	32.59	**34.81**
MMLU High School Physics	Accuracy	ppl	**37.09**	35.76
MMLU High School Computer Science	Accuracy	ppl	**55.00**	53.00
MMLU High School European History	Accuracy	ppl	**58.18**	56.97
MMLU Business Ethics	Accuracy	ppl	48.00	**49.00**
Winogrande	Accuracy	ll	52.96	**54.70**
ARC-c-Test	Accuracy Input Contaminated	ppl	30.19	**35.85**

表格 1: 模型微调前后在测评数据集上的表现

2.3.2 Interpretation of the Results

对表格1、图1中展示的结果和2.2节得到的比较结论进行深入比较，可以发现在部分测试数据集上，模型在微调前后出现了较大的差异。在平均准确率相近的情况下(49.17%：49.13%)，微调后模型在小部分数据集上表现优秀(26/72)同时也意味在这些数据集上的提升较为明显。为此，我们计算得到微调后模型在这些数据集上的结果平均比微调前高1.21%，而在大部分表现较差的数据集上平均仅差0.98%，这与我们对模型进行微调的目的相符，即在特定领域上拥有更为优秀的推理、理解和解决问题的能力。

从数据集特征的角度进行分析，Alpaca指令数据集涵盖了广大领域和大量类型的任务，包括数学问题的解答、文本的翻译等。其核心——指令结构为模型提供了清晰的任务定义和明确的学习目标，使模型在接触到更加丰富的场景同时建立任务理解与输入输出之间的映射关系，帮助模型强化上下文感知和问题求解能力。微调模型在部分MMLU学科数据集上的优秀表现因此可以部分解释为模型学习了Alpaca数据集中相同领域的复杂指令，强化了模型对这部分内容的多步推理能力并增加了模型接触到的知识面，或是从指令数据集中学习了部分与评测数据集非常相似的指令，这些指令对于模型来说属于少样本学习；而在部分MMLU学科数据集上出现下降则可解释为涉及到了较少或者没有涉及该领域的指令，归属于零样本学习，难以在这些题目上进行较好的泛化。在HellaSwag数据集上的小幅度提高可以解释为模型通过在Alpaca上的理解指令学习提高了上下文感知能力和遵循指令进行输出时提高了模型的推理能力所致。

最后，通过模型对示例输入“Introduce”这个指令的不同输出，也能够看出模型对指令响应发生的变化，倾向于输出满足客观规律的更为完整和详细的答案，而这也恰恰符合Alpaca指令数据集所期望得到的输出。

3 Conclusion

在本次大作业中，我们使用Alpaca-cleaned数据集对Qwen-2.5-B大语言模型进行全量微调，并充分比较了微调前后模型在一系列评测数据集上评测指标的变化，并通过微调前后模型对同一输入的不同输出和微调、评测数据集的特征对微调后模型的评测结果进行分析，随后对该结果做出了可能的解释，说明了此次微调达成了预期的结果。

Acknowledgements

本次大作业的实现代码部分参考了《大模型微调入门：SFT Qwen2-7B，基于 Hugging Face Transformers库》和Fine-tuning Alpaca and LLaMA: Training on a Custom Dataset，但大部分都由我自己通过查阅官网和其他相关网页进行实现，小组成员为我一个人。

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235.

Jeffrey L Elman. 1990. Finding structure in time. *Cognitive science*, 14(2):179–211.

Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. 2000. Learning to forget: Continual prediction with lstm. *Neural computation*, 12(10):2451–2471.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. 2024. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*.

I Loshchilov. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6):7.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.