A Comparative Study of Fine-Tuning Pipelines for Integrating Large Language Models in Multimodal Data Analysis

Cătălin Griu (c.griu@tudelft.nl) [1] Supervisor: Atahan Akyıldız (t.a.a.akyildiz@tudelft.nl) [1] Responsible Professor: Kubilay Atasu (kubilay.atasu@tudelft.nl) [1]

[1]Delft University of Technology

Introduction

- Multimodal Data
  - In today's data-driven environment, companies generate vast amounts of information in tables containing categorical, numeric, and textual data.
  - Artificial intelligence capabilities are necessary for utilizing this data across various tasks.
- FT-Transformer
  - The FT-Transformer [1] is an adaptation of the Transformer [2] architecture designed for tabular data.
  - It is designed to process numerical and categorical features.
- LLMs
  - Large Language Models (LLMs) [3] offer a powerful way to integrate textual data with other data types and subsequent models.
  - There are numerous approaches to achieving this integration.
  - An example of integrating LLMs with a graph newral network can be seen in Figure 1.

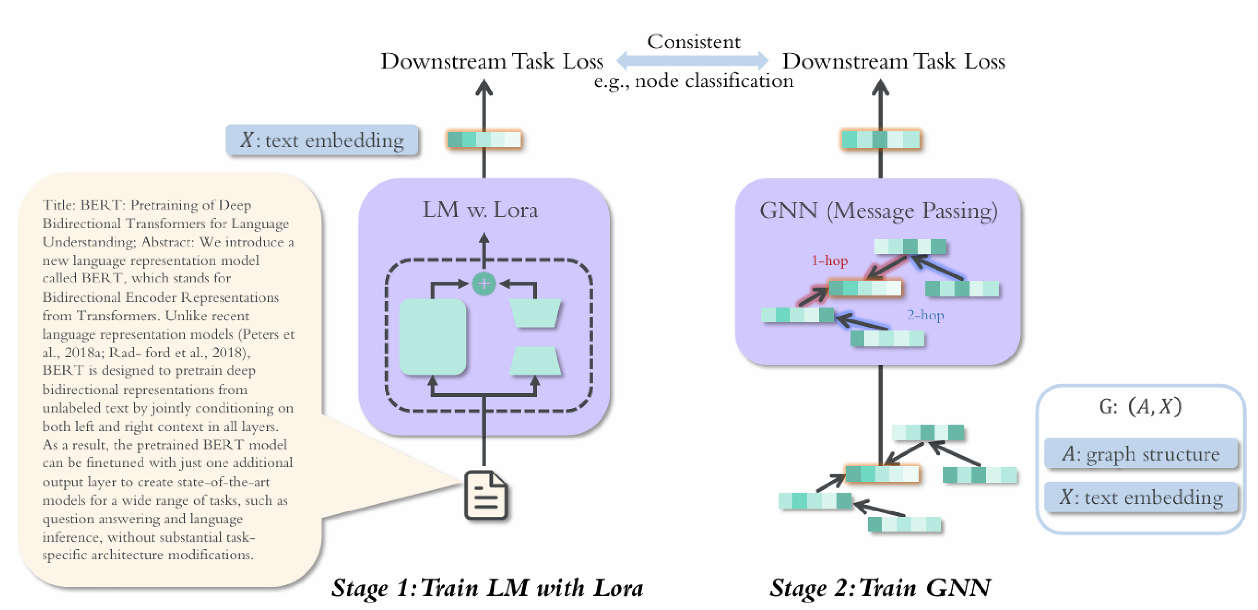

Figure 1. Combined Architecture of a LLM and a GNN. Taken from [4]

Research questions

- How can we integrate pre-trained LLMs with the FT-Transformer for handling multimodal tabular data?
- Is fine-tuning the LLMs beneficial, and if so, which fine-tuning method works best?
- Should the LLM be fine-tuned separately or together with the downstream model?
- How does the size of the LLM impact both cost and performance?

Methodology

- Combining LLM with FT-Transformer
  1. The LLM generates embeddings for text fields.
  2. Text embeddings are concatenated with embeddings of categorical and numerical features.
  3. A linear transformation ensures uniform embedding dimensions if they differ in size.
  4. The resulting matrix of embeddings is processed by Transformer layers.

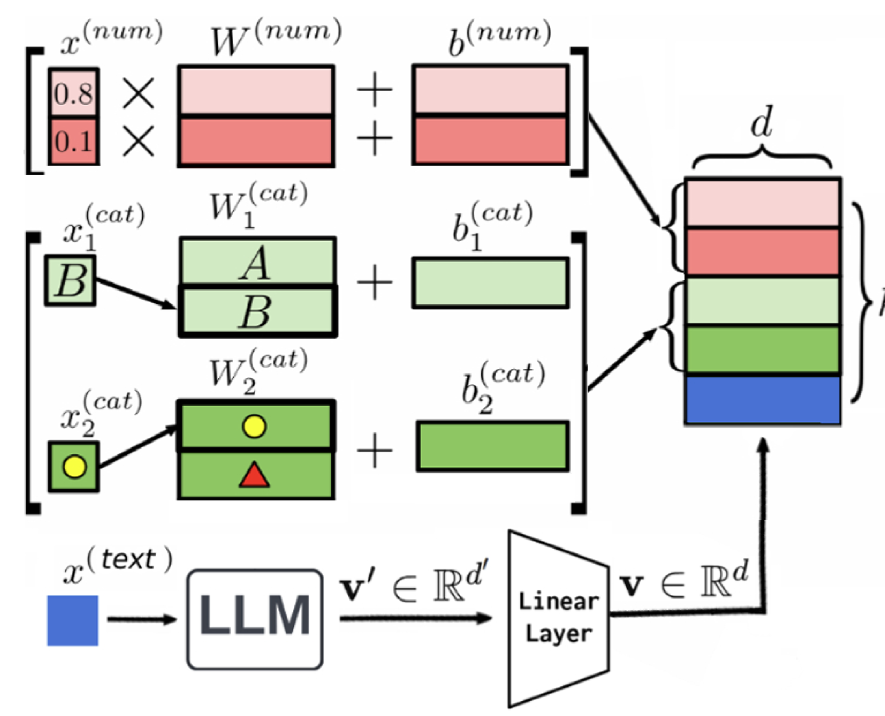

Figure 2. Integration of the LLM with the Feature Tokenizer stage of FT-Transformer. Adapted from [1].

- LLMs
  1. all-distilroberta-v1 (82M parameters)
  2. e5-mistral-7b-instruct (7B parameters)

- Training Pipeline

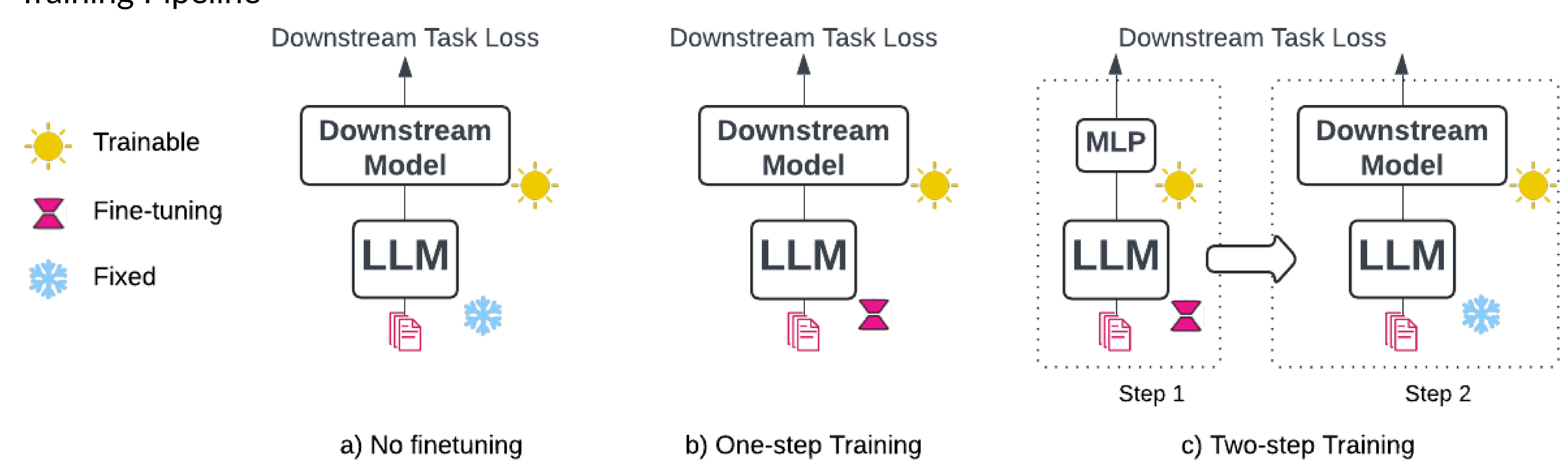

Figure 3. The illustration of 3 training methods: (a) No Fine-tuning, (b) One-step Training, (c) Two-step Training. Adapted from [5].

- LLM Fine-tuning methods

LLM	LLM Fine-tuning Strategy	LLM Trainable Params
DistilRoBERTa	No fine-tuning	0
	LoRA (rank 64)	1.18M
	Prompt (24 tokens)	18,432
	Full fine-tuning	83.1M
e5-mistral-7b	No fine-tuning	0
	LoRA (rank 64)	27.26M
	Prompt (24 tokens)	98,304
	Full fine-tuning	7.11B

Table 1. Trainable parameter analysis of DistilRoBERTa and e5-mistral-7b under different fine-tuning methods.

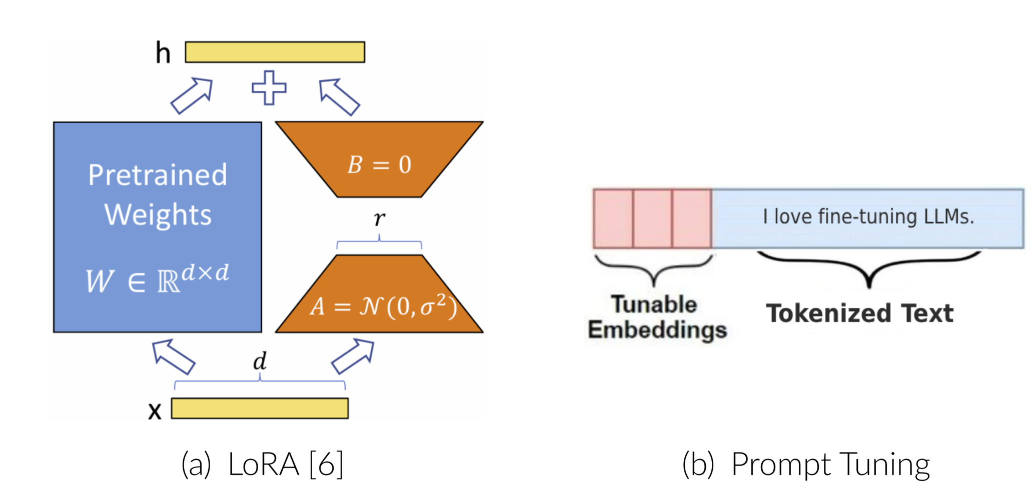

Figure 4. PEFT (Parameter-Efficient Fine-Tuning)

Results

Table 2. Comparison of LLM fine-tuning pipelines and their impact on downstream model performance across multimodal datasets. For each configuration: LLM - Downstream Model - Dataset, we **bold** the best and underline the second-best result.

LLM	Downstream Model	LLM Fine-tuning Pipeline	Amazon Fashion MSE ↓	Amazon Fashion Time	OGBN-ArXiv Accuracy ↑	OGBN-ArXiv Time
DistilRoBERTa	MLP	No fine-tuning	0.3087	13min	73.62	20min
		One-step (LoRA)	0.2244	5h	**74.42**	5.7h
		One-step (Prompt)	0.3043	4.5h	73.48	5.6h
		One-step (Full)	**0.1972**	5.8h	74.38	6.5h
	FT-Transformer	No fine-tuning	0.5379	30min	73.00	50min
		One-step (LoRA)	0.5196	6.4h	73.22	9.4h
		One-step (Prompt)	0.63	6.4h	73.24	9.4h
		One-step (Full)	0.6012	9.2h	73.22	10.7h
		Two-step (LoRA)	0.5112	5.5h + 30min	73.85	5.7h + 50min
		Two-step (Prompt)	0.5704	4.5h + 30min	73.45	5.6h + 50min
		Two-step (Full)	**0.3598**	5.8h + 30min	**74.02**	6.5h + 50min
e5-mistral-7b	MLP	No fine-tuning	**0.1778**	3h	76.02	20h
		One-step (LoRA)	0.1858	60h	**76.79**	75.7h
	FT-Transformer	No fine-tuning	**0.4544**	10h	**75.61**	20h
		Two-step (LoRA)	0.4857	60h + 3h	75.18	75.7h + 20h

Conclusion

In this study, we explore the use of pre-trained Large Language Models (LLMs) combined with FT-Transformer to advance learning techniques for multimodal data. Our study highlights effective LLM fine-tuning pipelines and examines how model size influences both performance and cost, providing practical guidelines for future implementations.

- **Impact of LLM Choice**: Larger models, such as e5-mistral-7b, significantly outperform smaller models like DistilRoBERTa, indicating the importance of model selection in achieving superior results.
- **Effectiveness of Fine-Tuning**: Fine-tuning LLMs on specific datasets can significantly improve performance metrics. Both full fine-tuning and LoRA are effective methods.
- **Training Method**: Decoupling the fine-tuning of the LLM from the downstream model training is the most effective approach.

References

[1] Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting deep learning models for tabular data," *Advances in Neural Information Processing Systems*, vol. 34, pp. 18932–18943, 2021.

[2] 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*, vol. 30, 2017.

[3] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, *et al.*, "Mistral 7b," *arXiv preprint arXiv:2310.06825*, 2023.

[4] K. Duan, Q. Liu, T.-S. Chua, S. Yan, W. T. Ooi, Q. Xie, and J. He, "Simteg: A frustratingly simple approach improves textual graph learning," *arXiv preprint arXiv:2308.02565*, 2023.

[5] B. Jin, G. Liu, C. Han, M. Jiang, H. Ji, and J. Han, "Large language models on graphs: A comprehensive survey," *arXiv preprint arXiv:2312.02783*, 2023.

[6] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.