Fine-Tuning Llama 3.2 3B for Romanian Contextual Question-Answering via LoRA

Adela-Nicoleta Corbeanu, Iulia-Georgiana Talpalariu

Faculty of Mathematics and Computer Science

Artificial Intelligence Master's Program

July 2025

1 Introduction

In this project, we want to fine-tune LLaMA 3.2 3B (which does not officially support Romanian) for contextual question answering in Romanian language. To enable this, we will develop a custom Romanian dataset consisting of context-question-answer triples, where each question is designed to be answered solely based on the provided context. The dataset will focus on common knowledge and factual reasoning, aiming to evaluate and enhance the model's ability to understand Romanian in an instructional setting. Using LoRA, we will apply parameter-efficient fine-tuning to adapt LLaMA 3.2 to this task. This work aims to contribute both a valuable Romanian-language dataset and an efficient methodology for improving LLM performance in under-represented languages.

2 Dataset creation

We have build the dataset using data from Wikipedia[5] about topics enumerated in table 1. We split our work and have built the dataset in two phases described in same table. The dataset is available on Hugging Face [4].

The dataset is exclusively in Romanian about diverse subjects.

Phase	Topics	Train Samples	Test Samples
1	Pisici, Uniunea Europeană, Cehia, Germania Nazistă, Mozart, Vatican, UNESCO, Wikipedia, Premiul Nobel pentru Literatură, Haruki Murakami, Cutremurul din 1977, Mihai Eminescu, Donald Trump, Colivă, Simona Halep, Ploaie, FCSB, Astronomie	349	51
2	Munții Carpați, Munții Făgăraș, Munții Bucegi, Munții Rodnei, Transilvania, Bucovina, Marea Neagră, București, Brașov, Dunărea, Vulturul, Parlamentul României, Revoluția din 1989, Palatul Parlamentului, Parcul Herăstrău, Președintele României, Vulcan noroios, Munții Măcin, Timișoara, Munții Retezat, Suceava	251	149
Total	**(Combined 39 topics)**	**600**	**200**

Table 1: Overview of dataset construction phases and topic coverage

A train sample has this structure:

Field	Value
`text`	Cehia se învecinează cu Polonia la nord, cu Germania la vest, cu Austria la sud și cu Slovacia la est.
`question`	Cu ce țară se învecinează Cehia la sud?
`answer`	Austria
`source`	Wikipedia

Table 2: Sample training data entry for QA task

3 Methods

3.1 LoRA

Transfer learning has become a common practice in order to use a pre-trained model on a broader task to do a more specific task through fine-tuning. The fine-tuning process implies that the model (with all its weights) is trained starting from already trained weights on a more general task. But as the models have grown in the number of parameters (GPT-4 is thought to have a trillion parameters), fine-tuning meant a huge need for performant hardware resources.

In [1], the authors state that bigger models do not have to move very far in parameter space in order to adapt, which means that the bigger the model, the lower the rank in their weights' matrix.

LoRA paper [2] further prove that not only weights' matrix has a way lower rank that its dimensionality, but the update matrix (ΔW) has a lower rank. They use matrix decomposition in order to build two matrices A and B from ΔW.

If $W_0 \in \mathbb{R}^{d\times k}$ are the weights of the pre-trained model. This means that we can decompose $\Delta W = BA$, where $B \in \mathbb{R}^{d\times r}, A \in \mathbb{R}^{r\times k}$ and the rank $r \ll \min(d, k)$. The new weights would look:

$$W_0 + \Delta W = W_0 + BA \tag{1}$$

If r is the rank and α a scaling factor, the formula for the forward pass becomes:

$$h = W_0 x + \frac{\alpha}{r} BAx \tag{2}$$

3.2 Llama 3.2 3B

We chose Llama 3.2 3B [3] as pre-trained model to fine-tune. This model has a total of trainable 3,221,924,864 parameters and a dimension of 3072 (matrix Q).

As authors say, Llama 3.2 has official support for only 8 languages : English, German, French, Italian, Portuguese, Hindi, Spanish, and Thai, but it has been trained on data from another languages too (probably Romanian). We first tested that the tokenizer can work with Romanian words and accepts the special characters in Romanian.

3.3 Hardware

We were limited by our hardware, we first tried some experiments in Colab with T4 GPU, but we then switched to A100 (Colab Pro) to manage to do the experiments and also analyse them.

4 Metrics

As metrics we used exact match and F1.

$$\text{EM} = \frac{1}{N} \sum_{i=1}^{N} 1_{\hat{y}_i = y_i}$$

$$\text{F1}_i = \frac{2 \times \text{precision}_i \times \text{recall}_i}{\text{precision}_i + \text{recall}_i}$$

$$\text{Precision}_i = \frac{|\text{Pred}_i \cap \text{GT}_i|}{|\text{Pred}_i|}, \quad \text{Recall}_i = \frac{|\text{Pred}_i \cap \text{GT}_i|}{|\text{GT}_i|}$$

$$\text{F1} = \frac{1}{N} \sum_{i=1}^{N} \text{F1}_i$$

5 Experiments and Results

5.1 Different ranks

We run an experiment by trying out different ranks for LoRA. We have set α to be 4 times bigger.

According to results in table 3, all LoRA configurations significantly outperform the base model (EM: 0.100, F1: 0.299), confirming that even low-rank adaptation improves the model's performance on our QA task. The best F1 score (0.553) and EM (0.410) are achieved with $r = 2, \alpha = 8$, suggesting that very lightweight fine-tuning is sufficient - and possibly optimal — for our small dataset.

Rank (r)	Alpha	Exact Match	F1 Score	Trainable Params (%)
1	4	0.395	0.514	0.0089
2	8	0.410	0.553	0.0178
4	16	0.325	0.455	0.0357
8	32	0.230	0.419	0.0713
16	64	0.340	0.540	0.1426
32	128	0.145	0.428	0.2848
Base Model		0.100	0.299	–

Table 3: Evaluation results for various LoRA configurations on QA task, with corresponding trainable parameter percentages (target modules Q and V and LR=10^{-4})

Field	Content
Question	Ce specie de vultur este simbol național al Statelor Unite și are capul alb distinctiv?
Context	Vulturul pleșuv, emblematic pentru Statele Unite, este o specie cunoscută pentru penajul alb pe cap și coadă, distinctiv față de corpul său maro închis.
True Answer	Pleșuv
Predicted Answer (Base Model)	Vulturul american, care este simbolul național al Statelor Unite, este un vultur de mare dimensiune, cu o lungime de 2,5 metri și o greutate de 15-20 de kilograme. Are un cap alb și o coadă
Predicted Answer (LoRA r=1, α=4)	Vulturul pleșuv
Predicted Answer (LoRA r=2, α=8)	Pleșuv
Predicted Answer (LoRA r=4, α=16)	Pleșuv
Predicted Answer (LoRA r=8, α=32)	Pleșuv
Predicted Answer (LoRA r=16, α=64)	Vulturul pleșuv
Predicted Answer (LoRA r=32, α=128)	Vulturul pleșuv.

Table 4: Comparison of predicted answers from base model and LoRA fine-tuned models on a test example. The answer is extracted from model generation (the text after "Answer:" and before first "\n")

5.2 Different target modules

We tried different target modules for our LoRA fine-tuning. Performance metrics are shown in tables 5 and 6. We can see that With only Q projection adapted, performance is quite low (EM: 0.110), suggesting that adapting the query projection alone is insufficient for this task. Adding more modules gives a boost both in EM and F1.

Target Modules	Rank (r)	Alpha	Exact Match	F1 Score
q	2	8	0.165	0.305
q, v	2	8	0.165	0.511
q, k, v, o	2	8	0.190	0.485

Table 5: Performance of different LoRA target module configurations (LR=3×10^{-4})

Target Modules	Rank (r)	Alpha	Exact Match	F1 Score
q	2	8	0.110	0.485
q, v	2	8	0.345	0.533
q, k, v, o	2	8	0.415	0.542

Table 6: Performance of different LoRA target module configurations (LR = 10^{-4})

6 Conclusion

We have built a custom Romanian dataset and used it in order to fine-tune Llama 3.2 3B with it. Even if this model does not officially support Romanian language, its tokenizer works good on it, and even with no fine-tuning it seems to have some understanding in Romanian. All experiments showed improved performance of the fine-tuned model over the base non-fine-tuned model, which means that learning occurs in spite of the small fraction of trainable parameters. The fact that a lower rank (2) performed better that greater ranks may be explained by both the low complexity of the task and the dimension of the dataset. More target modules can boost metrics as the model can better understand.

References

[1] A. Aghajanyan, S. Gupta, and L. Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7319–7328, Online, August 2021. Association for Computational Linguistics.

[2] 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, 2021. Presented at ICLR 2022.

[3] Meta AI. Llama 3.2 3b. https://huggingface.co/meta-llama/Llama-3.2-3B, 2024. Accessed: 2025-07-05.

[4] OnnieNLP. Information Extraction QA. https://huggingface.co/datasets/OnnieNLP/InformationExtractionQA, 2025.

[5] Wikipedia contributors. Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Main_Page. Accessed: 2025-07-05.