LlamaLingual: A Fine-tuning Method for Romance Language Translation

Troy Dutton, Jasper Tan, Divyam Kamboj, Anusha Razdan, Varun Arumugam
The University of Texas at Austin

1 Abstract

Large language models (LLMs) excel in various natural language processing tasks but often face challenges in delivering high-quality, domain-specific translations. To address these limitations, we propose fine-tuning Llama 2, an open-source LLM, using a specialized multilingual dataset to enhance translation accuracy for Romance languages. The code and resources for this project are available on GitHub at `https://github.com/troydutton/LlamaLingual`.

2 Introduction

LLMs are typically trained on expansive datasets encompassing diverse text sources. While these models are proficient in general tasks like text generation, summarization, and translation, their performance diminishes when tasked with domain-specific translations. This project aimed to address this gap by fine-tuning Llama 2 with a Romance languages parallel corpus, optimizing its capabilities for accurate translations in Spanish, French, Italian, Portuguese, and Romanian.

The following sections detail the setup, methodology, and findings of this fine-tuning effort, concluding with benchmarks, results, and potential future improvements.

3 Related Work

Large Language Models (LLMs) have demonstrated significant potential in multilingual machine translation. Recent studies highlight strategies to improve these models for tasks such as translating Romance languages. One study focuses on enhancing document-level translation in LLMs by addressing limitations in handling longer texts exceeding 512 tokens. Typically, LLMs trained with sentence-level instructions fail to maintain context across sentences, leading to fragmented or incomplete translations in longer texts. To address this, the authors propose incorporating both sentence-level and document-level translation instructions during fine-tuning. This hybrid approach ensures that the model retains the ability to translate individual sentences effectively while also capturing broader contextual information necessary for document-level coherence. By integrating this methodology, models can translate documents up to 2048 tokens, improving accuracy and consistency across diverse text lengths.[1]

Another study systematically evaluates the multilingual machine translation performance of various LLMs, including ChatGPT and GPT-4. While these models continue to improve, the analysis reveals that they lag behind commercial translation systems for low-resource languages. The study also emphasizes the effectiveness of cross-lingual exemplars during training, demonstrating that such examples provide better task guidance for low-resource translations than same-language pair data. These findings suggest that incorporating diverse training instructions and cross-lingual exemplars during the fine-tuning process can significantly enhance translation capabilities. However, these studies do not discuss the performance or specific methodologies for fine-tuning Llama models. This gap motivated our choice to focus on fine-tuning Llama 2, leveraging its architecture and capabilities to address the challenges of translating Romance languages.[2]

4 Methodology

4.1 Model Selection

Llama 2 was chosen for this project due to its open-source nature, ease of integration with frameworks like Hugging Face, and reduced computational requirements compared to newer models like Llama 3. Specifically, the 7-billion parameter version of Llama 2 was used, balancing performance and computational efficiency.[3]

4.2 Optimization Techniques

To reduce computational overhead, the model parameters were quantized from 32 bits to 4 bits. Quantization reduces memory usage, making the model more suitable for resource-constrained environments, albeit with some trade-offs in precision. Additionally, we employed LoRA (Low-Rank Adaptation) to update a small subset of the model's parameters, further reducing the number of trainable weights.

4.3 Hardware Requirements

The fine-tuning process required:

- A GPU with at least 16 GB of VRAM.
- Adequate storage space for checkpoints and datasets.
- A Hugging Face account and API token for model access.

4.4 Dataset

For this project, we utilized the **OPUS Dataset**, a well-established and comprehensive collection of multilingual text corpora. OPUS provides parallel text data for a variety of languages, making it particularly suitable for machine translation tasks [4]. Our focus was on the Romance language subset of OPUS, Spanish, French, Italian, Portuguese, and Romanian. These languages share structural and linguistic similarities, making them ideal for evaluating translation models.

The OPUS dataset includes text from diverse domains such as legal documents, subtitles, technical manuals, and online content, offering a wide range of sentence structures and linguistic complexities. This diversity ensured the training data covered both formal and informal text styles, enhancing the generalization capability of the model.

The dataset was characterized by:

- **Languages:** Five Romance languages (Spanish, French, Italian, Portuguese, Romanian).
- **Sentence Pairs:** Approximately 2 million aligned pairs after preprocessing.
- **Domains:** Balanced representation across legal, technical, and informal domains.
- **Sentence Length:** Ranged from 5 to 50 words, ensuring a mix of short and complex sentences.

This curated dataset served as the backbone of the fine-tuning process, ensuring both linguistic diversity and quality. By leveraging the robust structure of the OPUS dataset and incorporating additional fields, we enhanced the model's ability to handle nuanced translation tasks effectively.

4.5 Data Processing and Preprocessing

To tailor the OPUS dataset for our specific use case, we organized the data into **source-target pairs**, where each pair consisted of:

- **Source:** A sentence in the original language.
- **Target:** Its corresponding translation in the target language.

Used the tokenizer provided with Llama 2 to convert the source and target text into tokenized formats.

4.6 Data Split

The processed dataset was divided into three subsets to facilitate training, and testing. Our dataset was sufficiently large such that a validation set was not needed:

- **Training Set (90%):** Used to fine-tune the model on source-target pairs.
- **Test Set (10%):** Held out for final evaluation of the fine-tuned model using BLEU and other metrics.

4.7 Model and Fine-Tuning Approach

We employed a parameter-efficient fine-tuning approach using Low-Rank Adaptation (LoRA) for adapting the Llama 2 large language model to romance language translation tasks. LoRA is a lightweight fine-tuning technique that introduces trainable low-rank matrices to the original model's weight matrices, allowing for targeted adaptation with minimal computational overhead.

4.7.1 Model Configuration

The LoRA configuration was carefully designed with the following hyperparameters:

- Rank (r): 20
- LoRA alpha: 8
- Target modules: Attention projection matrices (q_proj, k_proj, v_proj, o_proj)
- Dropout rate: 0.4
- Bias: None
- Task type: Causal Language Modeling

This configuration focuses on adapting the attention mechanisms of the model while maintaining the majority of the pre-trained weights frozen. The rank of 20 allows for a compact representation of model adaptations, and the 0.4 dropout rate helps prevent overfitting.

4.8 Training Procedure

The fine-tuning process was meticulously designed to optimize the Llama 2 model for Romance language translation through a comprehensive multi-stage approach. First, the pre-trained Llama 2 model was initialized using a custom `load_model()` utility. This ensured proper initialization of both the model architecture and its associated tokenizer, which was critical for maintaining the model's pre-trained knowledge while preparing it for domain-specific adaptation.

Next, Low-Rank Adaptation (LoRA) was configured using the Parameter-Efficient Fine-Tuning (PEFT) library. This adaptation targeted the model's attention mechanisms by selectively modifying the query projection (q_proj), key projection (k_proj), value projection (v_proj), and output projection (o_proj) matrices. This selective approach enabled precise adaptation while preserving the majority of the pre-trained weights, thereby mitigating catastrophic forgetting and maintaining the model's generalizability.

The dataset for translation underwent rigorous preprocessing to ensure optimal performance during training. This involved the implementation of a custom `TranslationDataset` class for efficient data handling, tokenization using the model's native tokenizer, and dynamic batching facilitated by a custom `collate` function. Additionally, the data was carefully separated into training and evaluation subsets to ensure robust performance assessment.

The training process itself was executed using the Hugging Face `Trainer`, which provided a reliable framework for model fine-tuning. Key features of this setup included gradient accumulation to optimize memory utilization, periodic model checkpointing for continuity, built-in evaluation mechanisms for ongoing performance validation, and seamless integration with experiment tracking platforms.

4.9 Performance Monitoring and Experiment Tracking

Performance monitoring was conducted through a multi-layered approach designed to provide comprehensive insights into the model's training dynamics. Weights & Biases (wandb) was leveraged for real-time experiment tracking, offering capabilities such as comprehensive logging of training metrics, visualization of learning curves, hyperparameter tracking and comparison, and resource utilization monitoring. To further enhance the reporting process, a custom logging callback was developed. This callback streamlined progress reporting by removing extraneous metrics (e.g., total floating-point operations), implementing a more efficient progress bar, and improving the overall readability of training updates.

Throughout the training process, key performance indicators (KPIs) were closely monitored. These included the number of trainable parameters, training and validation loss, learning rate progression, and convergence indicators. The combination of the custom logging mechanism and wandb integration ensured minimal overhead while providing detailed insights into the model's learning progression, allowing for real-time assessment and potential mid-training interventions.

5 Results

Table 1. Model Testing Results

Model	BLEU ↑	CHRF ↑	ROUGE-1 ↑	ROUGE-2 ↑	WER ↓
Llama-2	9.02	28.74	25.40	13.12	6.00
LlamaLingual (40k Samples)	37.77	51.56	59.62	37.27	**0.66***
LlamaLingual (80k Samples)	**38.96**	**53.09**	**60.86**	**38.56**	**0.66***

* Indicates a tie in performance

The evaluation metrics in Table 1 provide a comprehensive view of the quality of translations produced by each model. BLEU (Bilingual Evaluation Understudy) measures the precision of n-grams in the translated text compared to a reference, emphasizing how well the model captures literal overlaps [5]. A higher BLEU score indicates translations closer to the reference. CHRF (Character F-score), on the other hand, evaluates character-level matches with an emphasis on recall, which is particularly useful for assessing translations of morphologically rich languages [6]. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) scores focus on recall, comparing the overlap of n-grams (e.g., unigrams for ROUGE-1 and bigrams for ROUGE-2) between the candidate and reference texts, making them valuable for capturing fluency and content preservation [7]. Finally, WER (Word Error Rate) quantifies the word-level differences between the translation and the reference by counting substitutions, deletions, and insertions required to align the two. Unlike the others, lower WER values are better, as they signify fewer errors in word-level alignment.

6 Discussion

The results in Table 1 demonstrate a marked improvement in translation quality after fine-tuning Llama-2 with 40k and 80k samples, showcasing the effectiveness of task-specific training. The baseline Llama-2 model achieves modest scores across all metrics, with a BLEU score of 9.02

and a WER of 6.00, indicating that its out-of-the-box performance for romance language translation is limited. After fine-tuning with just 40k samples, the BLEU score jumps to 37.77, and CHRF rises to 51.56, illustrating a substantial increase in both n-gram precision and character-level recall. Similarly, ROUGE-1 and ROUGE-2 scores improve significantly, indicating that the fine-tuned model captures content more effectively and produces translations with better fluency and coherence. The WER plummets to 0.66, a dramatic reduction in word-level errors, which further underscores the fine-tuned model's reliability.

Notably, the performance gain between the 40k-sample and 80k-sample fine-tuned models is incremental but valuable, suggesting diminishing returns with increased data. Both fine-tuned models achieve the same WER score, indicating robust word-level accuracy even with fewer samples. These results highlight the transformative impact of fine-tuning, even with a relatively small dataset, in adapting a general-purpose LLM like Llama-2 to excel at specific tasks such as romance language translation.

7 Future Work

- **Expanding Model Capacity:** Fine-tuning larger versions of Llama 2, such as the 13B or 70B parameter models, could enhance translation accuracy, particularly for complex sentence structures and low-resource language pairs.
- **Diverse and Robust Datasets:** Incorporating additional datasets for underrepresented Romance languages or specialized domains can improve generalizability and robustness.
- **Advanced Training Techniques:** Explore data augmentation, knowledge distillation, and curriculum learning to improve training efficiency and quality.
- **Enhancing Computational Efficiency:** Implement mixed-precision training or sparsity techniques and leverage distributed training.
- **Evaluation Beyond BLEU:** Use metrics like METEOR, TER, or human evaluation to provide a more holistic assessment.
- **Real-World Application Testing:** Conduct usability studies in real-world scenarios to guide refinements.
- **Linguistic Features:** Explore features based on Linguistic data like Glossing and Semantic Similarity of words during the training process to improve accuracy by providing additional context

References

[1] Yachao Li, Junhui Li, Jing Jiang, and Min Zhang. Enhancing document-level translation of large language model via translation mixed-instructions, 2024.

[2] Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. Multilingual machine translation with large language models: Empirical results and analysis, 2024.

[3] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, and Ross Taylor and. Llama 2: Open foundation and fine-tuned chat models, 2023.

[4] Jörg Tiedemann. OPUS – parallel corpora for everyone. In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation: Projects/Products*, Riga, Latvia, May 30–June 1 2016. Baltic Journal of Modern Computing.

[5] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In Pierre Isabelle, Eugene Charniak, and Dekang Lin, editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[6] Maja Popović. chrF: character n-gram F-score for automatic MT evaluation. In Ondřej Bojar, Rajan Chatterjee, Christian Federmann, Barry Haddow, Chris Hokamp, Matthias Huck, Varvara Logacheva, and Pavel Pecina, editors, *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[7] Kavita Ganesan. Rouge 2.0: Updated and improved measures for evaluation of summarization tasks, 2018.