Energy-Efficient Fine-Tuning of Large Language Models on Devanagari Script-based Languages

Enhancing the ability of Meta's open source LLM - Llama 2 using QLoRA technique for better utility with Indic languages

Aarsh Desai
Data Science and Artificial Intelligence
IIIT Dharwad
Surat, India
21bds001@iiitdwd.ac.in

Ashish Gidijala
Data Science and Artificial Intelligence
IIIT Dharwad
Bengaluru, India
21bds007@iiitdwd.ac.in

N.V.J.K Kartik
Data Science and Artificial Intelligence
IIIT Dharwad
Bhilai, India
21bds041@iiitdwd.ac.in

Priyesh Gupta
Data Science and Artificial Intelligence
IIIT Dharwad
Kota, India
21bds050@iiitdwd.ac.in

Vinayak
Data Science and Artificial Intelligence
IIIT Dharwad
Bengaluru, India
21bds069@iiitdwd.ac.in

***Abstract*—This report presents fine-tuning of the Llama 2 language model for Devanagari script languages, emphasizing Hindi and Sanskrit. The model showcases enhanced proficiency in language understanding and generation. Evaluation metrics, linguistic variations, and practical applications are discussed.**

***Index Terms*—Llama 2, Devanagari script, Fine-Tuning, Hindi, Sanskrit, Natural Language Processing**

I. Introduction

While Llama 2 has emerged as a robust solution for English language fine-tuning, its uncharted territory in Devanagari script languages, such as Hindi and Sanskrit, marks a unique venture. This pioneering study seeks to bridge the gap in linguistic resources for Devanagari, addressing the intricacies vital for effective language generation and comprehension.

A central challenge arises from the notable disparity in linguistic resources compared to well-established adaptations for English. The essence of Devanagari nuances, crucial for nuanced language processing, poses a distinctive hurdle to overcome. This study is motivated by the vision to refine Llama 2 for Devanagari, paving the way for an inclusive linguistic repertoire.

Resource efficiency becomes paramount in our pursuit, given the computational demands often associated with cutting-edge methods. The conventional 32 GB VRAM requirement for fine-tuning is a challenge that we successfully navigate, achieving commendable results with a 16 GB VRAM setup—readily available on platforms like Google Colab and Kaggle. Our commitment to energy-efficient strategies ensures wider accessibility, particularly in resource-constrained environments.

This fine-tuning endeavor focuses on the 7B parameter model within Meta's Llama 2 framework, promising a nuanced understanding of Devanagari linguistic structures. In the subsequent sections, we delve into the methodology, experimental setup, and results, providing valuable insights into the augmentation of Llama 2's capabilities for Devanagari script languages. This endeavor not only contributes to advancing natural language processing but also holds significant promise for the empowerment of Hindi speakers and the broader Indian community.

II. Literature Review

A. Language Models and Multilingual Adaptations

In prior research, a hybrid approach combining GPT generation, multilingual embeddings, and contextual bandit learning algorithms showed a 15-20% average improvement across languages on blackbox LLMs [1].

The robust multilingual capabilities of Large Language Models (LLMs), such as LLaMA, are underpinned by diverse multilingual data and vocabulary [2]. LLaMA is pre-trained on an extensive dataset of over 1.6 trillion tokens, incorporating less than 4.5% multilingual data across 20 languages. LLaMA2 further enhances multilingual data to approximately 11% and expands language support to around 26. Other models, like PolyLM and BLOOM, also demonstrate significant language coverage in their pre-training phases. This substantial inclusion of diverse language data during pre-training establishes a robust foundation for the multilingual capabilities of LLMs [3].

B. NLP in Devanagari Script Languages

In this study, the impact of leveraging language relatedness within the Indo-Aryan (IA) family through multilingual fine-tuning is investigated. The research demonstrates that such an approach outperforms individual language fine-tuning in downstream NLP applications. The study systematically adds related languages to a base language, revealing that a carefully

selected subset significantly improves performance, with up to a 150% relative improvement in downstream tasks compared to monolingual fine-tuning [4].

C. Resource-Efficient Fine-Tuning and Computational Considerations

In addressing the resource-intensive nature of fine-tuning large language models (LLMs), the QLoRA method has proven instrumental. With innovations like 4-bit NormalFloat, Double Quantization, and Paged Optimizers, QLoRA significantly reduces memory requirements for finetuning a 65B parameter model, enabling training on a single GPU [5].

III. Dataset Amalgamation

In order to curate a comprehensive and linguistically diverse training dataset for the fine-tuning of the Llama 2 model on Devanagari script-based languages, we undertook the amalgamation of various existing datasets. This amalgamated dataset serves as the foundation for the model's training and evaluation.

A. Source Datasets

We gathered data from the following existing datasets, each contributing unique linguistic nuances to enrich the training corpus:

1) **Bhagavad Gita verse-wise (English, Hindi, Sanskrit)**
   - Description: This dataset comprises verse-wise content from the Bhagavad Gita in three languages: English, Hindi, and Sanskrit. The Bhagavad Gita is a sacred Hindu scripture that contains a conversation between Prince Arjuna and the god Krishna, offering profound philosophical teachings. This dataset provides a unique linguistic resource for exploring the nuances of language in the context of spiritual and philosophical discourse.
   - Dataset Link: https://www.kaggle.com/datasets/yashnarnaware/bhagavad-gita-versewise
2) **Hindi Wikipedia Articles - 55k**
   - Description: Comprising a collection of Hindi Wikipedia articles, this dataset includes diverse content covering a wide range of topics. With 55,000 articles, it serves as a valuable linguistic resource for training and fine-tuning language models. The dataset spans various domains, providing a rich source for understanding language usage and capturing the linguistic diversity present in Hindi Wikipedia.
   - Dataset Link: https://www.kaggle.com/datasets/disisbig/hindi-wikipedia-articles-55k
3) **hindichat**
   - Description: The hindichat dataset is available on the Hugging Face Datasets platform and is a collection of Hindi conversational data. It is designed for tasks related to natural language understanding and generation in Hindi. This dataset is valuable for training models that can comprehend and generate text in a conversational context. The inclusion of chat-based data enhances the model's ability to understand and respond to informal language.
   - Dataset Link: https://huggingface.co/datasets/rishiraj/hindichat

B. Data Preprocessing

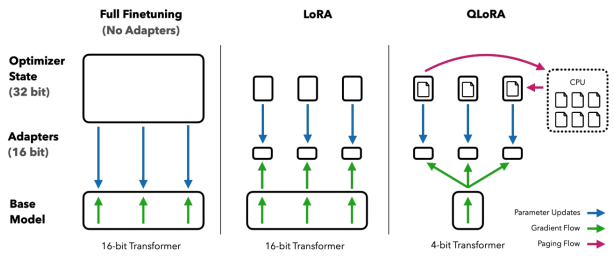
The datasets were preprocessed to ensure uniformity and compatibility. This preprocessing included:

- **Language Alignment**: Aligning the languages within the datasets to ensure a consistent linguistic context.
- **Tokenization and Standardization**: Employing tokenization and standardization techniques to maintain consistency in data representation.
- **Quality Control**: Removal of noisy or irrelevant data points to enhance the overall quality of the training corpus.

IV. Methodology

A. Environment Setup

In this study, fine-tuning of the Llama 2 model was conducted on Kaggle's infrastructure, utilizing 2 x T4 GPUs. The experiment employed the Accelerate, PEFT, BitsandBytes, Transformers, and Trl libraries to optimize and train the model. Additionally, QLoRA (Quantization Low-Rank Adapter) was integrated for efficient memory usage during training.



B. Configuration of Parameters

The following key parameters were configured for QLoRA and BitsandBytes, optimizing the fine-tuning process contributing towards efficient memory usage and optimized resource utilization:

1) **LoRA Attention Dimension**: Set to 64, controlling the dimensionality of the attention mechanism in QLoRA.
2) **LoRA Scaling**: Defined as 16, determining the scaling factor for QLoRA.
3) **LoRA Dropout**: Maintained at 0.1, regulating the dropout probability for LoRA layers during training.
4) **4-Bit Precision**: Activated to reduce memory usage, employing a 4-bit quantization technique for enhanced computational efficiency.

C. Training Procedure

*1) **Training Configuration**:*

- Batch Size (4 per GPU): The training process involves processing batches of data, and in this case, each batch contains four examples per graphics processing unit (GPU). This configuration optimizes the use of available GPU resources.
- Mixed Precision (Standard precision - fp32): Standard precision (floating-point 32-bit) is employed for mixed precision training, balancing computational efficiency with numerical precision.

*2) **Gradient Handling**:*

- Accumulation: Gradients are accumulated over multiple iterations before performing a single update step. This allows for more stable and efficient training.
- Clipping (Gradient norm clipped at 0.3): To prevent exploding gradients, the norm of the gradient is capped at 0.3 during training.
- Checkpointing: Efficient memory usage is achieved through gradient checkpointing, a technique that enables training of large models without exhausting GPU memory.

*3) **Optimization/Learning Rate**:*

- Optimizer (Paged AdamW with 32-bit precision): The paged AdamW optimizer is used with 32-bit precision to update the model parameters during training.
- Learning Rate Schedule (Cosine schedule): The learning rate follows a cosine schedule, dynamically adjusting during training to enhance convergence.

*4) **Training Steps/Logging**:*

- Warmup (3% linear warmup): The learning rate experiences a 3% linear warmup, gradually increasing from 0 to the target learning rate to stabilize the training process.
- Sequence Batching (Uniform-length sequence batching): Batches are organized to group sequences of similar lengths together, enhancing training efficiency.
- Checkpointing (Regular saving and logging every 25 steps): Model checkpoints are saved, and training progress is logged every 25 steps for monitoring and potential recovery.

*5) **Sequence Fine-Tuning (SFT)**:*

- Max Sequence Length (Adapted to varying input lengths): The model is designed to handle varying input sequence lengths efficiently, adapting to the specific demands of the training data.
- Packing (Balanced efficiency and performance): Sequences are efficiently packed to balance computational efficiency with overall training performance.
- GPU Allocation (Entire model loaded on 2 x T4 GPUs): The model is distributed across two T4 GPUs, leveraging parallel processing capabilities for accelerated training.

V. Results

In evaluating the performance of our fine-tuned Llama 2 model, we employed the BLEU (Bilingual Evaluation Understudy) score, a widely used metric in machine translation tasks. The BLEU score measures the similarity between the generated text and a set of reference texts, providing a quantitative assessment of translation quality.

Model	BLEU Score
Baseline Llama 2	5.20
OURS	**34.87**
ChatGPT-3.5 Turbo	72.69

In our thorough evaluation of the adapted Llama 2 for Devanagari languages, specifically Hindi and Sanskrit, we witness a significant performance leap, effectively addressing the unique linguistic nuances of these languages. Our fine-tuned model stands out with a substantial improvement over the baseline, evident in the BLEU score.

Comparing against ChatGPT, a language model boasting an impressive BLEU score of 72.69 for Hindi, trained on an expansive dataset with considerable computational resources, our fine tuned model achieves a noteworthy BLEU score of 34.6. Trained on a more focused dataset from three sources, this substantial improvement over the baseline score of 5.2 underscores the effectiveness of our tailored adaptation.

It's vital to note that ChatGPT's outstanding performance is largely attributed to its access to extensive and diverse training data, coupled with substantial computational capabilities. In contrast, our model showcases its efficacy by achieving significant improvements with a more targeted dataset. This not only highlights the adaptability of our approach but also positions it as a practical alternative in scenarios where computational resources are constrained.

VI. Conclusion

In conclusion, our pioneering adaptation of Llama 2 for Devanagari languages, Hindi and Sanskrit, addresses the inherent linguistic complexities, resulting in a fine-tuned model that significantly outperforms the baseline. The observed substantial improvement, reflected in the BLEU score, underscores the efficacy of our tailored approach.

While ChatGPT, with its remarkable BLEU score, sets a high standard, our model's noteworthy performance, achieved with a more focused dataset and limited computational resources, positions it as a practical and efficient alternative. This emphasizes the adaptability and accessibility of our approach in scenarios where computational constraints are a significant consideration.

Our study not only contributes to the expanding landscape of language models for Devanagari languages but also opens avenues for further exploration and refinement. As we navigate the intricacies of linguistic adaptation, our work lays the foundation for more nuanced and resource-efficient models tailored to specific language families.

References

[1] Akshay Nambi et al., Breaking Language Barriers with a LEAP: Learning Strategies for Polyglot LLMs, 2023

[2] Hugo Touvron et al., Llama 2: Open Foundation and Fine-Tuned Chat Models, 2023
[3] Fei Yuan, Shuai Yuan, Zhiyong Wu, and Lei Li, How Multilingual is Multilingual LLM?, 2023
[4] T.I. Dhamecha, Rudra Murthy V, S. Bharadwaj, K. Sankaranarayanan, and P. Bhattacharyya, Role of Language Relatedness in Multilingual Fine-tuning of Language Models: A Case Study in Indo-Aryan Languages, 2021
[5] Tim Dettmers, A. Pagnoni, Ari Holtzman, and Luke Zettlemoyer, QLoRA: Efficient Finetuning of Quantized LLMs, 2023
[6] Sanjana Kolar and Rohit Kumar, Multilingual Tourist Assistance using ChatGPT: Comparing Capabilities in Hindi, Telugu, and Kannada, 2023