Reproducibility Appendix
Project Report for NLP Course, Winter 2023/4

Filip Kucia
Warsaw University of Technology
filip.kucia.stud@pw.edu.pl

Szymon Trochimiak
Warsaw University of Technology
szymon.trochimiak.stud@pw.edu.pl

Bartosz Grabek
Warsaw University of Technology
bartosz.grabek.stud@pw.edu.pl

Supervisor: Anna Wróblewska
Warsaw University of Technology
anna.wroblewska1@pw.edu.pl

1 Overall Results

1.1 Model Description

The project combines **Low-Rank Adaptation (LoRA)** and **Retrieval-Augmented Generation (RAG)** framework to achieve efficient, domain-specific adaptation of a general-purpose language model. This design ensures computational efficiency while addressing the nuanced demands of the artistic domain.

1.1.1 Low-Rank Adaptation (LoRA)

LoRA fine-tuning methodology is applied to the pre-trained model **LLaMA 3.2-1B-Instruct**. LoRA introduces trainable low-rank decomposition matrices into selected layers of the model while freezing the pre-trained weights. This reduces the number of trainable parameters, thus optimizing computational resource usage without compromising performance.

1.1.2 Retrieval-Augmented Generation (RAG)

The RAG pipeline incorporates external knowledge into the model's outputs. This retrieval system accesses a curated database of artistic materials, including books, articles, and archival resources provided by the Faculty of Media Art. RAG ensures:

- Real-time retrieval of the most relevant knowledge, enhancing the factual accuracy of the chatbot's responses.

- Contextually aware information delivery, crucial for addressing complex queries about historical and creative aspects of the Faculty of Media Art.

- Scalability, allowing updates and expansions to the database without requiring extensive re-training.

1.2 Link to Code

A complete Python implementation of the fine-tuning and rag pipeline is provided. The code includes detailed documentation and installation instructions in requirements.txt , along with all dependencies, such as `Transformers` and `PEFT` libraries. The source code for this project is available on GitHub at the following link: GitHub Repository: Artistic-Chatbot .

1.3 Infrastructure

- **Hardware:** Training and inference were conducted on NVIDIA 4090 in 8-bit precision.

- **Software:** The pipeline relies on the `PyTorch` framework, `Hugging Face Transformers` for model handling. The `PEFT` library is used to manage LoRA-based training setups.

1.4 Runtime Parameters

The training runtime was 2 hours for finetuning on first round of human feedback. The average inference time for an answer is 15.94 seconds over 10 runs.

1.5 Model Parameters

- The LoRA setup involves 16 trainable parameters per target module, including `q_proj`, `k_proj`, `v_proj`, and `o_proj`. This design ensures efficient fine-tuning by modifying only a small subset of the model parameters.

- The base model (LLaMA 3.2-1B) contains 1 billion parameters. The LoRA-specific parameters are added on top of these, maintaining a low computational overhead.

1.6 Validation Performance

The validation performance of the model is evaluated using token-level accuracy on a held-out test set. Additional metrics such as BLEU and ROUGE scores can be computed to assess the quality of generated outputs for creative and factual tasks.

1.7 Metrics

Evaluation metrics include:

- **Accuracy:** Used to assess token-level correctness, especially on structured tasks.
- **BLEU/ROUGE:** Applicable for creative and free-form text generation, providing insights into linguistic and semantic alignment.

2 Multiple Experiments

2.1 Number of Training and Evaluation Runs

The training process spanned across **30 epochs**, with a gradient accumulation step size of **4**.

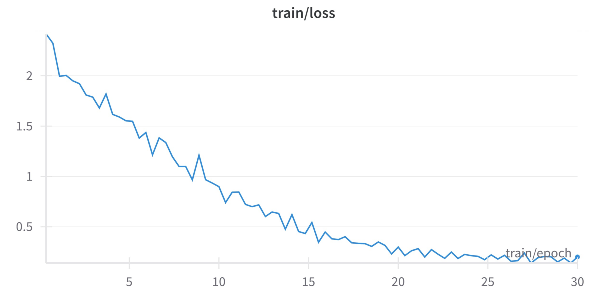

Figure 1: Cross-entropy between each target token and the prediction

2.2 Hyperparameter Bounds

The following hyperparameters were employed during the fine-tuning process:

- **Learning Rate:** `2e-4`
- **Batch Size:** `4`
- **LoRA Rank (r):** `16`
- **LoRA Dropout:** `0.05`

These bounds were set to optimize training performance while minimizing the risk of overfitting or underutilization of computational resources.

2.3 Hyperparameter Search

Hyperparameter tuning was primarily achieved through **manual selection**. While this method yielded satisfactory results based on iterative testing and performance evaluation, automated optimization tools such as `Optuna` or `Ray Tune` could further enhance the search process. Future work may incorporate these tools to explore a broader search space efficiently.

2.4 Expected Performance

The artistic chatbot is designed to deliver high-quality performance across multiple aspects, leveraging fine-tuned language modeling, retrieval-augmented generation (RAG), and human-centric feedback loops. The expected outcomes are as follows:

1. **Factual Accuracy**
   The chatbot is expected to correctly recall facts about the Faculty of Media Art, its history, and the achievements of its professors based on its curated knowledge base. The RAG framework ensures the system dynamically retrieves relevant and accurate information, even for nuanced or complex queries.

2. **Creative and Context-Aware Responses**
   The chatbot should engage users with creative and meaningful conversations that reflect the artistic domain's unique style, incorporating metaphors, symbolism, and abstract ideas. Fine-tuning using domain-specific datasets allows the chatbot to adopt stylistic nuances and ensure contextually appropriate responses.

3. **Multilingual Support**
   The chatbot's primary mode of interaction is in Polish, both for input and output. Expected performance includes seamless translation of user queries, accurate retrieval of relevant knowledge, and generation of fluent, coherent responses.

4. **Controlled Creativity**
   The chatbot is expected to generate creative responses within predefined boundaries, ensuring artistic originality while avoiding uncontrolled hallucinations. Controlled hallucinations will be fine-tuned based on user feedback to balance factual reliability and imaginative outputs.

Language	Mean	Median
Polish	199.21 k	151.12 k
English	208.38 k	179.59 k

Table 1: Mean and Median Length of Articles in thousands (words) in Polish and English Translations

3 Datasets

3.1 Dataset Statistics

The dataset consists of **165 PDF documents** curated from books, articles, and archives provided by the Faculty of Media Art. Key statistics include:

3.2 Data Splitting

The dataset was not divided into splits.

3.3 Data Processing

The PDF documents were converted into plain text using automated parsing tools. The raw text underwent the following preprocessing steps:

- **Cleaning:** Removal of extraneous characters and formatting inconsistencies.
- **Tokenization:** Chunking the text into manageable units for model input.
- **Truncation and Padding:** Ensuring all sequences conform to a maximum length of **2048 tokens**.

3.4 Data Accessibility

The cleaned and processed dataset and first human feedback can be accessed via a Google Drive.

3.5 New Data Description

New annotations were curated based on feedback from **ASP employees**, incorporating domain-specific expertise. Quality control measures included:

- Iterative refinement based on pilot evaluations.
- Validation by domain experts to ensure accuracy and relevance.

3.6 Dataset Languages

The primary language of the dataset is **Polish**. For processing, the data was:

- **Translated to English:** Using `GPT-4` to leverage model compatibility.
- **Translated Back to Polish:** Ensuring the chatbot's output aligns with user expectations in Polish.