# Using Large Language Models to Translate Persian Poetry into Modern Prose

Amirhossein Ghavi[1], Toktam Ghafarian[2]

[1]Department of Computer Engineering, Khayyam University, Mashhad, Iran, a.qavi@khayyam.ac.ir
[2]Department of Computer Science, SWPS University of Social Sciences and Humanities, Poland, tghafarian-mabhoot@swps.edu.pl

## Abstract

Large language models (LLMs) have recently become powerful tools for multilingual translation and text interpretation. Their performance depends strongly on the quality and coverage of training data, which has traditionally limited their development to large companies. With the rise of open-source LLMs, researchers can now fine-tune models for specific domains. In this study, we fine-tuned a Google-developed open-source model on the poems of Hafez and their interpretations. The model was then applied to unseen works, successfully generating coherent and meaningful interpretations. Our findings highlight the potential of LLMs to support literary and cultural analysis.

Keywords: large language models, fine-tuning, open source, Hafez poetry

## 1.Introduction

The field of Artificial Intelligence (AI) and modern technologies has led to rapid advancements in various research domains, particularly in Natural Language Processing (NLP). NLP offers numerous applications, including text translation, textual data analysis, and automated interpretation. Despite these advancements, a significant challenge remains in understanding and interpreting complex literary texts, especially classical Persian poetry, such as the works of Hafez. The intricate linguistic style and unique literary devices employed in Hafez's poetry often make it difficult for non-specialists and even enthusiasts to fully grasp its profound meanings and interpretations. Current large language models (LLMs), while powerful, are not yet capable of accurately translating and interpreting literary texts with this level of complexity. This gap highlights a critical need for more sophisticated tools to bridge the understanding between classical Persian literature and contemporary readers.

Researchers have explored various approaches for interpreting and analyzing literary texts, primarily relying on rule-based linguistic models and machine learning systems. Traditional models, while adept at understanding texts through linguistic rules and literary concepts, suffer from inefficiency in automated processing due to their inherent complexity and the requirement for deep human expertise. Similarly, statistical and machine learning models, leveraging large datasets, have struggled with deep and complex semantic analysis when confronted with texts like Hafez's poetry, failing to provide accurate interpretations. The advent of the Transformer architecture [1] marked a significant turning point in NLP by utilizing attention mechanisms for dependency modeling, outperforming previous recurrent and convolutional neural networks. Subsequent research demonstrated the capability of LLMs like GPT-2 [2] to perform various tasks, including translation, summarization, and question answering, without task-specific supervision. Techniques like RoBERTa [3] and ULMFiT [4] have further advanced the performance of text processing models. Optimization techniques such as Quantization and LoRA have enabled efficient deployment of LLMs on hardware-constrained systems. Evaluation metrics commonly used include ROUGE [5] for summarization and BLEU [6] for translation.

In parallel, the field of Large Language Models (LLMs) has increasingly turned its attention to the unique challenge of poetry translation, which has long been regarded as the "last bastion of human translation." Comparative studies show that LLM-generated translations often display translationese, characterized by reduced lexical diversity and simplified syntax compared to human renderings. Nevertheless, research has demonstrated that with carefully designed prompt engineering, these models can be guided to respect formal constraints, such as preserving rhyme schemes in sonnets [7]. Furthermore, innovative methods like Explanation-Assisted Poetry Machine Translation (EAPMT)—which leverages monolingual explanations of the source poem as an intermediate step—have shown superior performance in terms of accuracy and overall impression, particularly for modern poetry. These findings underscore the potential for LLMs to play a meaningful role in Computer-Assisted Literary Translation (CALT) workflows, where they augment rather than replace human creativity [8].

Persian literature, particularly its classical poetry, including the divan of Hafez, constitutes a vital component of Iranian culture and national identity. However, the linguistic intricacies and unique stylistic characteristics of these poems often impede a correct and precise understanding for many individuals. In today's fast-paced world, where readily accessible information is highly valued, there is a significant demand for quick and straightforward methods to comprehend the meanings and interpretations of literary texts, especially classical Persian poetry. This research addresses this need by leveraging artificial intelligence tools to simplify and streamline the interpretation of such complex literary works. By doing so, it aims to assist enthusiasts of Persian literature in achieving a more accessible understanding of these texts. Furthermore, this endeavor underscores the growing necessity for a native, specialized language model, cultivated from the rich repositories of Iranian culture and literature, to fill a critical gap in localized AI applications.

To address these challenges, this study fine-tunes an open-source language model, Gemma-2, developed by Google, on Hafez's poetry and their interpretations, enabling the model to generate meaningful explanations for new poems. By focusing on domain-specific fine-tuning, the research aims to demonstrate how LLMs can be adapted for literary and cultural analysis. The remainder of the paper is structured as follows: it reviews fundamental concepts and previous studies related to the research, details the dataset preparation and methodology (including data collection, preprocessing, and model training with optimization techniques), presents experimental results with a thorough analysis, and finally concludes the paper, discussing limitations and providing directions for future research.

## 2.Methods

This study employed a methodology centered on leveraging large language models (LLMs) to interpret classical Persian poetry, with a focus on Hafez's works. At the core of the approach lies the fine-tuning of an open-source model, specifically Google's Gemma-2 in its instructional version, chosen for its ability to provide precise and instruction-based responses when interpreting poetic verses.

The dataset was collected through web scraping from hafez.top[9], a resource that provides Hafez's ghazals along with their interpretations. The BeautifulSoup[10] library was used to extract poem verses, separated by div tags, together with corresponding interpretations, which occasionally included comparisons to the works of other poets. The extracted data was stored in JSON format for further processing.

A preprocessing pipeline was then applied to prepare the text for training. This included the removal of diacritics and tashdid, elimination of Arabic words and special characters that could hinder model comprehension, and conversion of Persian numerals into English digits for consistency. Unwanted characters such as the zero-width non-joiner (\u200c) were also removed. The Dadmatools library was employed to support Persian language normalization. Finally, the data was formatted into a prompt–response structure, with each verse serving as the input and its interpretation as the output. An End-of-Sequence token (EOS_TOKEN) was appended to each entry to clearly signal the termination of input for the model [figure 1].

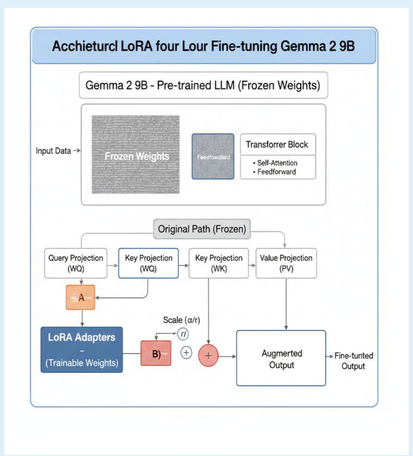


figure.1 architecture summary

For model training, the base model selected was Gemma-2-9b-it-bnb-4bit from the Unsloth library, a pre-trained LLM optimized for speed and efficiency. Although originally trained primarily on English data, this model required fine-tuning to handle Persian literary texts effectively. The architecture of the optimized model after Parameter-Efficient Fine-Tuning (PEFT) operations is illustrated in figure 2.

**Algorithm 1:** Prompt Formatting Function

**Input:** Examples of poems and translations
**Output:** Formatted text with input, output, and EOS token

1 **1. Define the prompt structure:** - Instruction: Describe the task. - Input: Provide poem as input. - Response: Provide the response for the input.
2 **2. Set EOS token:** EOS_TOKEN = tokenizer.eos_token
3 **3. Define the function** `formatting_prompts_func`**:** a. Extract inputs from examples["poem"] ;
4 b. Extract outputs from examples["translation"] ;
5 c. Initialize an empty list called `texts` ;
6 **foreach** *input_text, output_text in zip(inputs, outputs)* **do**
7 - Format the input and output with the alpaca_prompt template ;
8 - Add EOS_TOKEN at the end of the formatted text ;
9 - Append the formatted text to the `texts` list
10 **Return:** A dictionary containing the formatted texts ;

figure.2 pseudo-code for model training constructing instructions

This algorithm illustrates the process of formatting input-output examples for fine-tuning a language model. It first defines a structured prompt with Instruction, Input, and Response, sets an end-of-sequence token (EOS_TOKEN), and then uses a function (formatting_prompts_func) to iterate through the dataset. Each poem and its translation are formatted according to a predefined template, appended with the EOS token, and collected into a list. The function finally returns a dictionary containing all formatted prompts, ready for training [figure.2].

To address the considerable hardware demands typically associated with training LLMs, the study employed PEFT strategies. Two key techniques were utilized: Low-Rank Adaptation (LoRA) and quantization. LoRA significantly reduces memory and computational requirements by decomposing large parameter matrices into smaller, lower-rank approximations, allowing the training process to focus on a reduced number of parameters while preserving performance comparable to full fine-tuning

. In parallel, 4-bit quantization was applied (load_in_4bit=True) to further minimize the memory footprint and enhance computational speed. These optimizations were implemented using the Unsloth library, which is specifically designed for efficient LLM fine-tuning on constrained hardware. Additional optimizations included gradient checkpointing, which reduces memory consumption during backpropagation.

The training process was conducted on Paperspace using a carefully selected set of hyperparameters (summarized in figure 3). Parameters such as sequence length, batch size, learning rate, optimizer type, number of epochs, weight decay, and warmup steps were tuned to ensure stable convergence and avoid overfitting. A fixed random seed was applied to maintain reproducibility of results.

| Hyperparameter | Value | Description |
|---|---|---|
| Sequence length | 512 | Maximum number of tokens processed per input sequence |
| Batch size | 8 | Number of samples processed per iteration |
| Learning rate | 2e-4 | Initial step size for parameter updates |
| Optimizer | AdamW | Optimizer with decoupled weight decay for stable convergence |
| Epochs | 3 | Number of full passes through the training dataset |
| Weight decay | 0.01 | Regularization to prevent overfitting |
| Warmup steps | 50 | Gradual learning rate increase to stabilize early training |
| Gradient checkpointing | Enabled | Reduces memory usage during backpropagation |
| Quantization | 4-bit (bnb-4bit) | Reduces model size and speeds up training |
| LoRA rank (r) | 16 | Low-rank decomposition dimension in LoRA modules |
| LoRA $\alpha$ | 32 | Scaling factor for LoRA updates |
| Random seed | 42 | Ensures reproducibility of experiments |

figure.3 training hyperparameters used in fine-tuning Gemma-2

Through this methodology, the study was able to fine-tune Gemma-2 efficiently for the interpretation of Hafez's poetry. The adoption of PEFT techniques, particularly [11]LoRA and [12]quantization, proved instrumental in making the fine-tuning of such a large model feasible under limited hardware resources, ultimately enabling the development of a specialized system for the interpretation of Persian classical poetry.

## 3.Results

The model's performance was evaluated both before and after training using several metrics to quantify the improvement. The training was conducted on hardware with specifications mentioned in.

3.1. Evaluation Metrics

• BLEU (Bilingual Evaluation Understudy): Measures the similarity between the model's output and human reference texts, focusing on n-gram overlap and applying a brevity penalty. Scores range from 0 to 1, with higher scores indicating greater similarity.

• Perplexity: Indicates the model's ability to predict subsequent tokens in a sequence. Lower perplexity values signify a more accurate and better language model.

• ROUGE (Recall-Oriented Understudy for Gisting Evaluation): Evaluates text quality by comparing the generated text to a reference. Unlike BLEU, ROUGE emphasizes recall. This research used:

◦ ROUGE-1: Compares unigram (single-word) overlap.

◦ ROUGE-2: Compares bigram (two-word) overlap.

3.2. Performance Comparison (Before vs. After Training)

| Metric | Before Training | After Training | Improvement |
|---|---|---|---|
| BLEU Score | 48% | 55% | +7% |
| Perplexity | 52 | 29 | -44% (*Lower is better*) |
| ROUGE-1 | 0.06 | 0.07 | +16% |
| ROUGE-2 | 0.23 | 0.29 | +26% |

figure.4 comparison of results before and after training

3.3. Detailed Performance Analysis

• BLEU Score:

◦ Before Training: 0.48 (48%), indicating moderate performance in generating text similar to the reference.

◦ After Training: 0.55 (55%), showing a significant improvement in the quality of the generated text, suggesting better internal parameter adjustment. The graph of mean BLEU scores showed a decreasing trend before training, improving after training .

• Perplexity:

◦ Before Training: 52, meaning the model had 52 options for predicting the next token, reflecting moderate prediction accuracy.

◦ After Training: 29, indicating a substantial reduction in ambiguity and much higher accuracy in predicting the next token. The mean perplexity graph also demonstrated a notable improvement [Figure 4].

• ROUGE Scores (ROUGE-1 and ROUGE-2):

◦ Before Training: ROUGE-1 at 0.23 and ROUGE-2 at 0.07.

◦ After Training: ROUGE-1 at 0.29 and ROUGE-2 at 0.06. These results indicate improved alignment between the generated text and the reference text, particularly in unigram overlap.

3.4. Qualitative Results (Example Interpretations) figure 5 (Examples of model output before and after training) provides specific instances of how the model interpreted Hafez's poems before and after the fine-tuning process, demonstrating the qualitative improvements. For instance, a complex verse from Hafez received a rather vague interpretation before training but a much more coherent and contextually relevant interpretation after training.

| No. | Poem (brief in English) | Ground Truth | Prediction (Before) | Prediction (After) |
|---|---|---|---|---|
| 1 | He who wronged me—I kiss the dust of his path and ask forgiveness. | Devotional, self-effacing tone: the speaker accepts suffering and remains loyal, wishing the oppressor's prosperity (Hafez). | Reads the verse as generic forgiveness and magnanimity. | Identifies the courtly humility; links it to Hafez and the theme of loyal submission. |
| 2 | If union with the Beloved comes even for a moment, seize it—for all desires are fulfilled. | Urges seizing fleeting union with the Beloved; a carpe-diem motif in Sufi love poetry (Hafez). | Misattributes the verse (e.g., to Rumi) and gives a generic encouragement to act. | Attributes it to Hafez and explains the spiritual urgency of grasping the moment. |

figure.5 ground truth vs. model predictions before and after fine-tuning

Figure.5 provides a qualitative evaluation of our methodology, comparing the performance of the baseline model against the fine-tuned model (using LoRA) in interpreting Persian poetry. The Poem Excerpt column presents a brief English summary of the original verse fed to the model. The Reference Interpretation (Ground Truth) column contains the authoritative, expert-validated analysis, including correct literary context and attribution (e.g., Hafez). This is the ideal target interpretation. The Prediction (Before Fine-tuning) column shows the output from the initial large language model, which typically provides generic or contextually incorrect analysis. Finally, the Prediction (After Fine-tuning/LoRA) column demonstrates the improved capacity of our optimized model, which successfully incorporates and applies specific domain knowledge to yield accurate, nuanced literary interpretations, a key measure of the effectiveness of the adaptation method.

## 4.Conclusions

This research explored fine-tuning large language models, such as Google's Gemma, for interpreting Hafez's ghazals. Using optimizations like LoRA and quantization improved efficiency and reduced hardware needs, though limited Persian training data and model design constraints left room for improvement. Future work should expand datasets and develop Persian-specific LLMs for higher accuracy.

## 5.References

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 5998–6008.

[2] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog.

[3] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. ArXiv Preprint, arXiv:1907.11692.

[4] Howard, J., & Ruder, S. (2018). Universal Language Model Fine-tuning for Text Classification. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL), Melbourne, Australia, 328–339.

[5] Lin, C.-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. Proceedings of the ACL Workshop on Text Summarization Branches Out, Barcelona, Spain, 74–81.

[6] Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, PA, USA, 311–318.

[7] N. Resende and J. Hadley, "The Translator's Canvas: Using LLMs to Enhance Poetry Translation," Trinity Centre for Literary and Cultural Translation, Trinity College Dublin.

[8] S. Wang, D. F. Wong, J. Yao, and L. S. Chao, "What is the Best Way for ChatGPT to Translate Poetry?," NLP2CT Lab, University of Macau.

[9] www.hafez.top/divan-hafez.

[10] Beautiful Soup 4.

[11] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2021). LoRA: Low-Rank Adaptation of Large Language Models. Proceedings of the 38th International Conference on Machine Learning (ICML), 4568-4579.

[12] Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Hartwig, A., & Kalenichenko, D. (2017). Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 5323-5331.