Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

ViVerse-A1: Ascending Generating Vietnamese Poetry with Semantic and Cultural Fidelity

Phu Lam
Ho Chi Minh City University of Science
Hung Tran
Ho Chi Minh City University of Science
Tram Doan
Ho Chi Minh City University of Science
Binh Nguyen
Ho Chi Minh City University of Science
Son Huynh
htson@hcmus.edu.vn

Ho Chi Minh City University of Science

Research Article

Keywords: AI language processing, Vietnamese poem generation, fine-tuning, ViVerse-A1

Posted Date: April 7th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9115864/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

ViVerse-A1: Ascending Generating Vietnamese Poetry with Semantic and Cultural Fidelity

Phu G. Lam[1,2,3†], Hung P. Tran[1,2,3†], Tram T. Doan[1,2,3], Binh T. Nguyen[1,2,3], Son T. Huynh[1,2,3*]

[1*]Vietnam National University in Ho Chi Minh City, Vietnam.
[2]University of Science, Vietnam.
[3]AISIA Research Lab, Ho Chi Minh City, Vietnam.

*Corresponding author(s). E-mail(s): htson@hcmus.edu.vn;
†These authors contributed equally to this work.

Abstract

Despite their increasing sophistication in content generation, contemporary Large Language Models (LLMs) consistently fail to master the nuanced artistry of Vietnamese poetry, particularly its intricate semantic layers, figurative language, phonaesthetics, and deep-seated cultural resonance. This research confronts this challenge with the introduction of ViVerse-A1, a bespoke neural architecture optimized for Vietnamese poetic expression. ViVerse-A1's novelty lies in its hybrid training methodology, which synergizes targeted fine-tuning with advanced auxiliary frameworks, including a Masked Language Model and our proposed Masked-Token Reasoning Language Model. This design empowers the model to develop a granular understanding of the structural, prosodic, lexical, and diverse conventions of Vietnamese literature. Rigorous comparative analyses confirm that ViVerse-A1's performance surpasses that of leading state-of-the-art models, demonstrating a unique ability to generate poetry that is both technically proficient and expressively profound. The success of ViVerse-A1 signals a promising trajectory for AI-driven systems designed for deep semantic and cultural intelligence. To facilitate further research, the source code and all related materials are provided publicly at https://github.com/tph-kds/ViVerse_A1

Keywords: AI language processing, Vietnamese poem generation, fine-tuning, ViVerse-A1

1 Introduction

Automatic poetry generation, a field at the intersection of literary arts and artificial intelligence, aims to create works that not only adhere to strict prosodic rules but also possess significant expressive and cultural value. The evolution of machine learning and deep learning techniques has driven considerable advancements in this domain, enabling the generation of poems across various genres with increasing fidelity. However, current methodologies still face inherent limitations. Early research employing Recurrent Neural Networks (RNNs) [1, 2] demonstrated potential in maintaining the fundamental coherence and semantics of poetry. Nevertheless, these models were often constrained by limited training data, resulting in a lack of creative depth and linguistic sophistication.

To address the challenge of data scarcity in the domain of poetry generation, methods based on Retrieval-Augmented Generation (RAG) [3], have been proposed. By retrieving information from external knowledge sources, these models have demonstrated the ability to maintain poetic style and theme consistently. However, ensuring structural integrity and adherence to formal poetic prosody remains a significant challenge. Subsequent research building upon the RAG framework, such as FLARE [4], RegaVAE [5], G-Retriever [6], and METRAG [7], has further advanced performance in specialized tasks. These methods primarily focus on mitigating the phenomenon of "hallucination" [8] and enhancing the fidelity of the generated information. Despite these advancements, these state-of-the-art techniques exhibit several inherent limitations, including complex methodologies and substantial resource requirements. Many involve intricate computational steps and require intensive supervision from Large Language Models, as highlighted in METRAG [7].

Furthermore, their performance is predominantly validated on small-scale datasets, and they often lack comprehensive pre-training on large corpora, a noted drawback of RegaVAE [5]. This reliance on limited corpora consequently results in outputs that lack semantic diversity and depth. The aforementioned limitations present significant barriers to applying these models to morphologically rich and culturally nuanced languages such as Vietnamese. Specifically, existing models struggle to fully capture the subtle semantic shades, distinctive local linguistic features, and unique stylistic conventions of traditional Vietnamese poetry. This is precisely the research gap our study aims to address.

The advent of the Transformer architecture [9] and subsequent Large Language Models (LLMs) has inaugurated a new era for poetry generation tasks. Numerous studies have explored the potential of LLMs for various languages. For instance, GPoeT [10] utilized synthetic data to control rhyme and rhythm structures but struggled to expand lexical diversity and semantic depth. PoeLM [11] achieved high efficacy in controlling poetic form without large-scale labeled data, yet this was often at the expense of the work's artistic and semantic value. TPPoet [12] applied advanced decoding techniques to enhance coherence but was limited in creativity and thematic consistency. Meanwhile, AnchiBERT [13] focused on generating works aligned with historical-cultural contexts but showed limited adaptability to contemporary settings and lacked objective evaluation metrics. Other works [14–18] have also contributed to

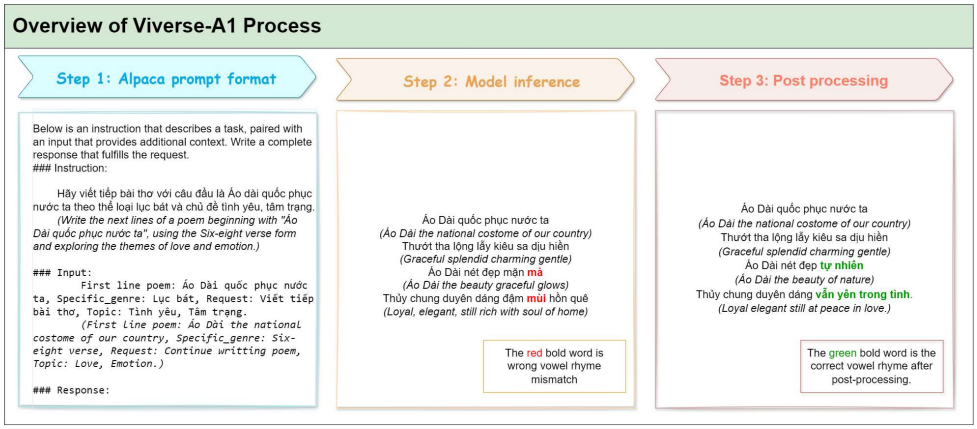


Fig. 1: Showcase the three key stages of the ViVerse-A1 process, providing an example for each step along with English translations. The process begins by formatting the Alpaca prompt, followed by generating a poem based on that input and reviewing the model's output. Finally, the poem undergoes post-processing to enhance the score.

improving thematic diversity and coherence, but their complex pipelines sometimes diminished the nuances and semantic specificity of the generated poems.

These challenges underscore the persistent difficulties in automatic poetry generation, particularly for Vietnamese, a language characterized by a complex tonal system and intricate poetic laws. To overcome these limitations, this paper proposes a holistic approach (see Figure 1), a model designed to comply with rigorous poetic standards while enhancing semantic diversity and harmony across various themes, ensuring that balancing adoptability between words and phrases aligns with the target topic. We recognize the potential of fine-tuning state-of-the-art LLMs, specifically the Qwen2.5-7B-Instruct model [19, 20] . This model is engineered for scalability across different poetic genres through strategic prompt engineering and a sufficiently large, curated dataset for fine-tuning. Our primary contributions are summarized as follows:

- We introduce ViVerse-A1 model, an end-to-end Vietnamese poetry generation model that integrates a fine-tuned LLM (Qwen2.5-7B-Instruct) with advanced *Post-preprocessing techniques*. Unlike conventional methods, our model leverages a powerful, state-of-the-art foundation model and a rigorous fine-tuning process to bolster the naturalness, flexibility, and semantic diversity of the generated works.
- The construction and public release of a large-scale Vietnamese poetry dataset, comprising over 200,000 poems that have been meticulously filtered and standardized from multiple reputable sources.
- A comprehensive, multi-faceted evaluation was conducted to provide a holistic assessment of the quality of the output poetry. Experimental results demonstrate

that ViVerse-A1 achieves superior performance compared to recent baseline models, as evidenced by metrics measuring prosodic accuracy and compositional stability.

2 Related work

2.1 CoT And Its Various Variants

A fundamental limitation of Large Language Models (LLMs) is the absence of robust mechanisms for knowledge retrieval and rigorous logical reasoning. This deficiency means that once a model initiates an erroneous line of reasoning, it lacks the ability to self-correct, leading to error propagation and a subsequent reduction in the reliability of its outputs. Therefore, developing methods to enhance the reasoning capabilities and deep comprehension of LLMs is a critical research imperative.

By tackling this limitation, Chain-of-Thought (CoT) prompting [21] was proposed as a seminal approach, providing the foundational rationale for many subsequent methods. By conditioning the model on a few step-by-step reasoning examples, this technique enables LLMs to generate intermediate logical steps, thereby significantly improving performance on complex reasoning tasks. A key advantage of CoT is its ability to enhance model interpretability and transparency without requiring computationally expensive processes like fine-tuning. To further optimize this process, sophisticated prompting strategies such as "Least-to-Most" [22] were introduced. These methods guide the model to decompose complex problems into simpler, sequential sub-problems, enabling it to handle tasks of greater difficulty.

Beyond improving reasoning, another major challenge is mitigating "hallucination," a phenomenon where the model generates information not present in the source data. The "Chain-of-Verification" (CoVe) method [23] was developed to address this by introducing a structured self-verification process, allowing the model to review and correct its own outputs systematically. However, a common drawback of these advanced methods, including CoVe [23], RAP [24], and various multimodal reasoning techniques [25], is their high computational overhead.

A comprehensive survey in [26] synthesizes and analyzes a range of prompting techniques, including Zero-Shot Prompting, Self-Consistency, Rephrase and Respond (RaR), and Contrastive Chain-of-Thought (CCoT). These techniques demonstrate considerable flexibility and effectiveness in improving model accuracy, robustness, and interpretability. Nevertheless, their performance is highly contingent upon the quality and relevance of the examples provided in the prompt; the selection of inappropriate knowledge can lead to unintended and erroneous outcomes.

Having to overcome the reliance on static, pre-supplied knowledge within prompts, the "Chain-of-Knowledge" (CoK) framework [27] proposes a novel approach. This method allows the model to dynamically and flexibly retrieve knowledge from heterogeneous sources during the reasoning process, which helps to mitigate error propagation. However, CoK confronts a different challenge: the reliability of the

external knowledge sources. Retrieving inaccurate information can lead to flawed conclusions, underscoring the fundamental concerns of reliability and veracity that persist in contemporary LLM systems.

Informed by the analysis of these related works, we identified Chain-of-Thought (CoT) as a robust methodology with potential for application extending beyond conventional abstract reasoning tasks. Accordingly, within the landscape of this study, we have selected and integrated a semi-structured variant of CoT into the core operational pipeline of our ViVerse-A1 model. The objective of this integration is twofold: to effectively leverage the substantial benefits offered by the CoT method, while simultaneously and proactively controlling for the inherent limitations that could adversely affect output quality in the poetry composition task.

2.2 SP-GPT2

Transformers, especially GPT-2 [28], transformed text production through their robust attention mechanisms and auto-regressive modeling. GPT-2, developed by Radford et al. [28], demonstrates proficiency in producing coherent and contextually appropriate prose. SP-GPT2 [29], based on GPT-2, was created for “Luc Bat" (Eight-line Form) poetry. This strategy used LSTM-based contextual vector modeling to ensure semantic consistency between verses to capture long-term dependencies inside stanzas. A new loss function was introduced by combining cross-entropy loss for structural learning with mean squared error loss for maintaining semantic coherence in rhyming verse pairs. This split loss function significantly improved the model's ability to write Luc Bat-compliant poetry.

2.3 Poem-Frame CharPoet

A significant instance is the CharPoet system created by Yu et al. [30], which employs an innovative methodology for generating ancient Chinese poetry. CharPoet employs a token-free language model, circumventing conventional tokenization, and implements a structure-sensitive method that encapsulates the complexities of classical forms such as SHI and CI. This approach closely aligns with our methodology of utilizing format-specific cues during model training to enhance structural conformity and creative fluency.

2.4 RAG-Viverse

RAG-Viverse[31] is a combination of the Retrieval-augmented generation (RAG) [32] technique and traditional model generation for Vietnamese poetry. RAG-Viverse offers a unique training approach for poem generation, utilizing two layers of long short-term memory (LSTM) [33] to create a Masked Poem, retrieving a similar poem in the database for reference, and then employing a large language model with a personalized prompt to replace and rephrase the words in the Masked Poem according to its context and rules. With that architecture, RAG-Viverse can handle every Vietnamese poem structure.

2.5 Fine-Tuning Large Language Models for Specialized Use Cases

Recent advancements in large language models have demonstrated their adaptability in natural language processing. General-purpose LLMs frequently falter in domain-specific or structurally constrained tasks due to their deficiency in linguistic, stylistic, and functional alignment, as noted in Fine-Tuning Large Language Models for Specialized Use Cases (2024) [34].

The research shows that fine-tuning can make large language models better at some tasks. Fine-tuning a model needs more than just data from the same field. Also, consider how the model is constructed, its intended purpose, and the testing process. The authors say that fine-tuning helps models understand language that is specific to a particular field, follow rules that are specific to a certain task, and reduce hallucinations and confusion. Still, they warn against overfitting and the problems that come with adapting to specific tasks when we have a lot of data and computing power.

The paper does not focus on generative creativity tasks like writing poetry. However, its results are still helpful for tasks that require linguistically and stylistically consistent outputs. The idea of constrained fine-tuning, which involves adding structural, lexical, or semantic properties to the training process, can be directly applied to tasks that need control over rhyme, rhythm, and syntax.

3 Methodology

This research centers on the development of ViVerse-A1, a comprehensive, end-to-end architecture for language processing, as illustrated in Figure 1. The foundation of this architecture is the Qwen2.5 Large Language Model (LLM), which underwent a specialized fine-tuning process using an extensive Vietnamese text corpus. This procedure was designed to bolster the model's coherence and robustness, thereby maintaining semantic integrity and minimizing inaccuracies during the processing of the Vietnamese language. The ViVerse-A1 architecture is structured into three core modules, as shown in Figure 2: (1) Data Selection and Pre-processing, where prompt engineering techniques are applied; (2) Model Inference; and (3) Post-processing technologies. The fine-tuning approach, in particular, is instrumental to the overall success of the model. The specifics are as follows:

3.1 Fine-tuning Strategy

To make the model more adaptable to structured and stylistically limited Vietnamese poetry, a parameter-efficient fine-tuning strategy through the processing pipeline illustrated in Figure 2. This method strikes a balance between making the computer work and ensuring the results have the right style.

In this study, we chose the Qwen2.5-7B-Instruct model [35] as our main generation model. Instruction-based fine-tuning [36] is done with Low-Rank Adaptation (LoRA) [37], which makes adaptation more efficient by adding trainable adapters to important projection layers while keeping the base model frozen. The Unsloth [38] library is

used to quantize the model to 4 bits of precision, which saves memory and makes deployment easier.

Training samples are organized according to the Alpaca prompt format [39], which integrates an instruction, contextual input (such as an initial poetic line, desired verse form, and theme), and a reference output. This instruction-based framework allows the model to acquire both poetic form limitations and thematic coherence.

The resultant fine-tuning pipeline is computationally efficient, scalable across various poetic forms, and adaptable for future improvements such as tone-aware decoding or reinforcement of structural constraints.

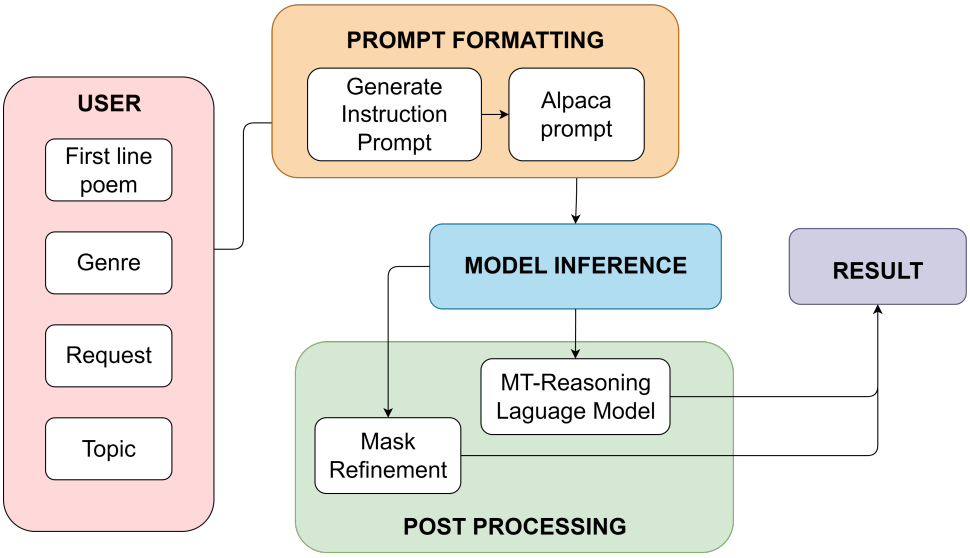


Fig. 2: ViVerse-A1 process for poetry generation, integrating user inputs with prompt formatting, masked-token reasoning, and post-processing to produce structured poetic outputs.

3.2 Data Selection and Preprocessing

In creating the *ViVerse-A1* dataset, we start by collecting raw poems from the original source[1], and then apply the scoring method proposed by Nguyen et al. [29] to filter for high-quality samples. Only poems with a score greater than 90 are retained for further processing.

To prepare the dataset for instruction-based generation, we adopt the Alpaca format, which consists of three columns: **Input**, **Instruction**, and **Output**. Specifically:

- **Input**: This field contains the contextual information necessary for poem generation. It includes the opening line of a poem, the specific poetic form (e.g.,

[1] Vietnamese Poetry Corpus on Hugging Face

Six-eight verse, *seven-character regulated verse form*), and the thematic category (e.g., love, family, nature). Additionally, a *Request* tag indicates whether the goal is to continue a given first line or to generate a poem from scratch.

- **Instruction**: This column follows a consistent Vietnamese template:

  Write the following lines of a poem beginning with <first_line>, using the <specific_genre> form and exploring the themes of <topic>.

- **Output**: This field contains the remainder of the poem, excluding the first line, which is already given in the Input. In cases where the request is for random generation, the Output contains the whole poem.

Upon completion, the dataset is structured according to the Alpaca schema from Alpaca Stanford and prepared for subsequent training of instruction-tuned generative models (Prompt Formatting step in Figure 1). This step ensures compatibility with modern instruction-following fine-tuning pipelines and allows seamless integration into the overall poem generation framework.

3.3 Model Inference

When the model archives the input, it generates a poem and then checks the poem output against the poem score rule [29]. Suppose the resulting score falls below the threshold of 100. In that case, the poem will highlight positions with incorrect vowel rhythm, then post-process to increase the score by replacing those highlighted positions with other words that have better context in the poem and follow the poem's rules.

3.4 Post-preprocessing Technologies

3.4.1 Mask Refinement

In this post-processing technique, fine-tune the PhoBert model [40] as a Masked Language model to replace the wrong vowel rhythm words in the poem.

Fine-tune PhoBert for poem post-processing

The dataset for the post-editing task is constructed using the same training data employed for the main poem generation model. Each poem is decomposed into individual lines, and each line is annotated with its corresponding poetic genre (e.g., *lục bát*, *thất ngôn bát cú*).

To facilitate rule-based correction, a new column is generated wherein specific positions within each line are obscured using the special token `[MASK]`. The structural limitations of each poetic form determine these positions:

- For **lục bát** (six-eight) verse:
  - Mask the 6^{th} syllable in the 6-syllable line.
  - Mask the 6^{th} and 8^{th} syllables in the 8-syllable line.
- For **thất ngôn bát cú** (seven-word regulated verse):
  - Mask the 7^{th} syllable in each line.

Once the masked dataset is prepared, the post-editing procedure proceeds as follows:

Algorithm 1 Rule-Constrained Poem with PhoBert

Input: $P_0 \leftarrow$ Generated Poem Line, Genre $\mathcal{G}$
Initialize: MaskToken $\leftarrow$ "[MASK]", NoMatchToken $\leftarrow$ "[NO_MATCH]"
PositionsToMask $\leftarrow$ GETMASKPOSITIONS(P_0, $\mathcal{G}$)
$P_1 \leftarrow$ REPLACEMASK(P_0, PositionsToMask, MaskToken)
$P_2 \leftarrow$ PHOBERTPREDICT(P_1)
NoMatchPositions $\leftarrow$ CHECKRULES(P_2, $\mathcal{G}$)
while NoMatchPositions $\neq \emptyset$ **and** attempts $<$ MaxAttempts **do**
    $P_2 \leftarrow$ RESAMPLEFROMINPUT(P_2, NoMatchPositions)
    $P_2 \leftarrow$ PHOBERTPREDICT(P_2)
    NoMatchPositions $\leftarrow$ CHECKRULES(P_2, $\mathcal{G}$)
end while
if NoMatchPositions $\neq \emptyset$ **then**
    Output: $P_{\text{final}} \leftarrow P_0$ ▷ Fallback to original line
else
    Output: $P_{\text{final}} \leftarrow P_2$
end if

In Algorithm 1, the process begins by evaluating the generated poem line against the structural rules of its corresponding genre. Any positions that violate the vowel rhythm constraints are replaced with the token `[MASK]`.

In the initial step, PhoBERT is used to predict suitable replacements for these `[MASK]` positions. Subsequently, the updated poem is checked for any remaining `[NO_MATCH]` tokens. If none are found, the algorithm terminates successfully.

If `[NO_MATCH]` tokens are present, the algorithm identifies the original position that caused the violation and performs a resampling operation via `RESAMPLEFROMINPUT`. For instance, in a Lục bát poem with the following rhyme structure:

$$\begin{array}{rl} _\ _\ _\ _\ _ & R_1 \\ _\ _\ _\ _\ _ & R_1\ _\ R_2 \\ _\ _\ _\ _\ _ & R_2 \\ _\ _\ _\ _\ _ & R_2\ _\ R_3 \end{array}$$

In this structure, each R_x denotes a position that must satisfy the same vowel rhythm constraint across all occurrences of the same index x. The underscores (_) represent freely chosen words not constrained by rhythmic rules.

Suppose the second or third occurrence of R_2 is marked as `[NO_MATCH]`. In that case, the algorithm traces back to the first occurrence of R_2, randomly resamples a word at that position, and reinitiates the correction process.

Algorithm 1 continues to verify the rule compliance of the poem after each resampling. If any `[NO_MATCH]` tokens remain, the process restarts until either the maximum number of attempts is reached or no `[NO_MATCH]` tokens persist.

Finally, if the poem still contains unresolved `[NO_MATCH]` tokens, the algorithm returns the original input as the output. Otherwise, the post-processed poem is accepted as the final result.

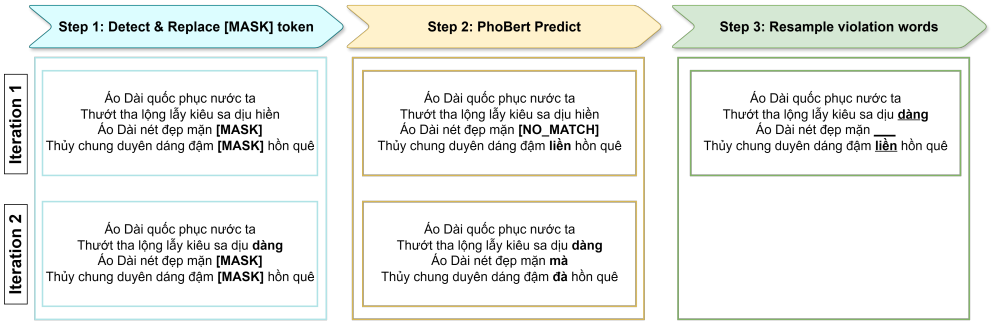


Fig. 3: Masked token refinement pipeline for structured verse correction

Here we present an example of generating a lục-bát poem beginning with "Áo Dài quốc phục nước ta." which is shown through Figure 3 and it The process is divided into three main steps:

1. **Detect & Replace [MASK] Tokens.** Scan the current draft and identify any words that violate the required rhyme or tone pattern; replace each violation with the special placeholder token `[MASK]`.
2. **PhoBERT Prediction.** Feed the masked draft into PhoBERT, which predicts an appropriate replacement for each `[MASK]`. If no suitable candidate is found that satisfies the poetic constraints, the placeholder is converted to `[NO_MATCH]`.
3. **Resample Violation Words.** For every `[NO_MATCH]` token produced in Step 2, randomly sample an alternative word from a curated candidate list that fits the rhyme and tone requirements, replacing the placeholder with a valid poetic word.

As illustrated in Figure 3, during *Iteration 1*, Step 1 replaces out-of-pattern words with `[MASK]`; Step 2 fills most masks but leaves at least one `[NO_MATCH]`; Step 3 resamples a correct word for that slot. The revised poem then enters *Iteration 2*, where Steps 1–2 find no further violations and fully populate any remaining masks, at which point the process terminates with a fully compliant lục-bát poem.

3.4.2 Masked-Token Reasoning Language Model method (M-TRLM method)

Within the framework of the ViVerse-A1 model, it is a pivotal strategy in advancing enrichment post-processing methodology, the workflow of which is illustrated in Figure 1. In contrast to conventional methods that focus on local word substitutions (e.g., PhoBERT), our approach leverages the reasoning capabilities of contemporary Large Language Models (LLMs) [19], [41], [42]. This enables the model to perform step-by-step interpretation and inference, thereby achieving a profound understanding of the task's context and objectives. Consequently, the post-processing stage not only corrects errors but also enhances the semantic quality and artistic merit of the generated poetry. The process is structured into three principal stages: *(1) Input Acquisition and Pre-processing*, *(2) Calling an External LLM*, and *(3) Confidence-aware Result Prioritization*.

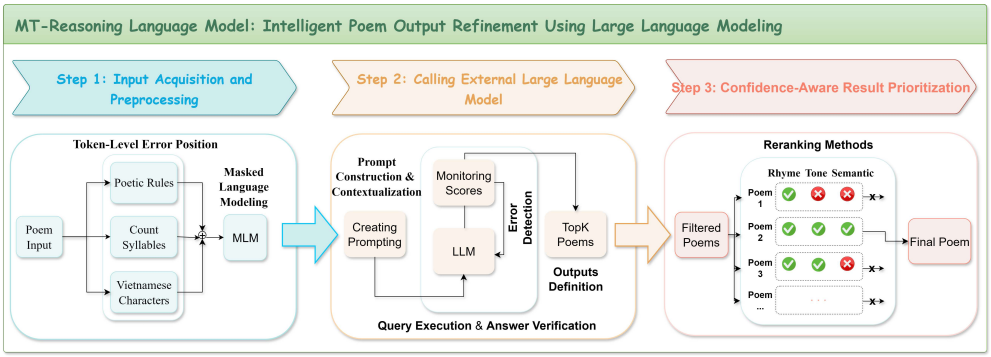


Fig. 4: System Architecture of Masked-Token Reasoning Language Model: Unified Multi-Stage Processing for Advanced Language Reasoning Tasks

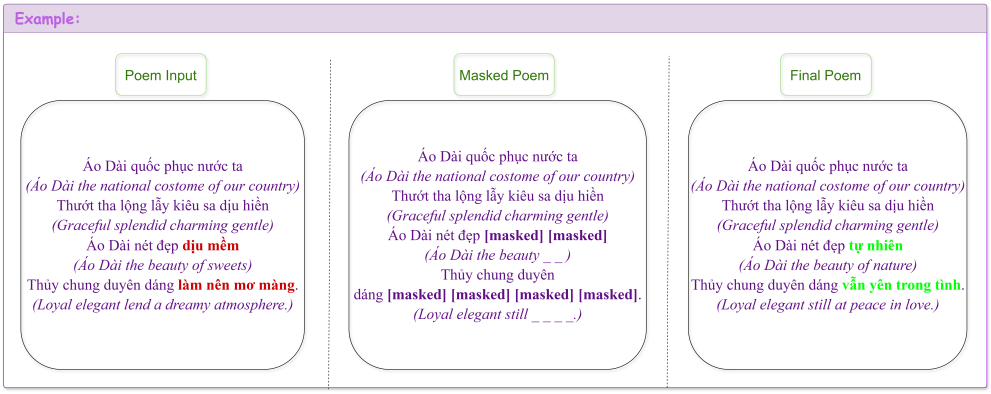


Fig. 5: Step-by-Step Illustration of the MT-Reasoning Language Model Process

Input Acquisition and Pre-processing: This initial stage involves passing the input poem through a Token-Level Error Localization Module, corresponding to step 1 in Figure 4. Meanwhile, provide illustrative examples for each corresponding step to enhance clarity in Figure 5. The function of this module is to identify and flag potential error locations. The types of errors analyzed include:

- *Poetic Rule Violations (Poetic Conformity Score (PCS)):* This checks the poem's adherence to the specific rules of its designated genre (detailed in Section 3.5.1).
- *Syllable Count Errors:* This identifies lines containing a surplus or deficit of syllables relative to the poetic form's requirements.
- *Vietnamese Character Errors:* This detects characters outside the standard Vietnamese alphabet (e.g., pictographs) or inappropriate vocabulary generated by the base model.

Algorithm 2 Token-Level Error Positions Fulfillment

Input: Draft Poem `P0`, Number of Error Categories `E` (default = 3)
Output: Refined Poem P_{final}
Initialize: `L` $\leftarrow [\,]$, `textToken` $\leftarrow$ "masked"
for $e = 1$ to E **do** ▷ Iterate through all error categories
    $I_e \leftarrow \textsc{CheckErrors}(P_0)$ ▷ Detect error positions
    Update L by including I_e into the set
end for
$P \leftarrow \textsc{MaskAndRefine}(P_0, L, \texttt{textToken})$ ▷ Generate refined poem
Return: $P_{\text{final}} \leftarrow P$

Once errors are localized, we apply a Masked Language Modeling (MLM) technique, as described in Algorithm 2. Words or phrases at the identified error positions are excised and replaced with a special *[masked]* token, as shown in an example in Figure 4. A key innovation in our methodology is rhythm-aware masking. Instead of masking tokens arbitrarily, this process strictly adheres to the characteristic rhythmic patterns of Vietnamese poetry (e.g., the 4/3 rhythm in Seven-syllable eight-line poetry ("Thất Ngôn Bát Cú" form), or the 2/2/2 rhythm for 6-syllable lines and 4/4 for 8-syllable lines in the "Lục Bát" form). Preserving this rhythmic structure guides the LLM to generate replacement candidates that are not only semantically correct but also prosodically harmonious, thereby significantly increasing the value of the post-processing stage.

Calling External Large Language Model: In this stage, a comprehensive prompt is constructed. This prompt endows the LLM with all requisite knowledge, including (1) the context regarding the rules and structure of the poetic form and (2) the masked poem content from the preceding stage. We utilize LLMs with robust reasoning abilities, such as Qwen3 [41], or DeepSeek-R1 [42]. The LLM returns a set of k candidates (top-k, where k defaults to 5). To handle instances where the output is truncated due to the model's token limit, a recall mechanism is activated to ensure a

complete set of k candidates is collected. All k candidates then undergo a preliminary filtering step to discard any poems that fail to comply with fundamental rules. A notable point is that the selection of DeepSeek-R1 [42] as the primary external LLM is based on a detailed comparative performance analysis against other models, including PhoBert. These empirical findings are presented clearly in Section 3.4.1.

Confidence-aware Result Prioritization After obtaining a set of valid candidates, the process proceeds to rerank them based on a triad of quantitative evaluation metrics: *(1) Rhyme:* The degree of rhyme compatibility between lines, *(2) Tone:* The conformity to tonal patterns as dictated by poetic law, *(3) Semantics:* The logical coherence and thematic cohesion of the entire poem. Each candidate's poem is scored against these three metrics. The final score is calculated as the average of the individual metric scores. The poem with the highest average score is selected as the final output of the post-processing pipeline.

3.5 Evaluation Metrics

3.5.1 Overall

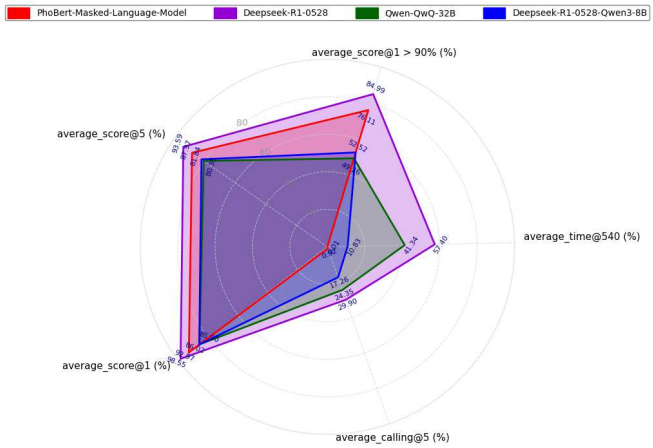
The evaluation of machine-generated poetry, particularly within the context of the Vietnamese language, constitutes a task of considerable complexity, requiring profound domain expertise in literature. The quality of a poem is not merely contingent upon its formal correctness but is also determined by its semantic coherence, artistic merit, and profound cultural-historical value. Consequently, no single evaluation method can holistically encompass these facets.

To address this challenge, we propose a multi-dimensional evaluation framework designed to provide a comprehensive and objective assessment of the generated poems. This framework comprises three principal dimensions: (1) Formal and Semantic Integrity, (2) Coherence and Interpretability, and (3) Stylistic Novelty and Imaginativeness.

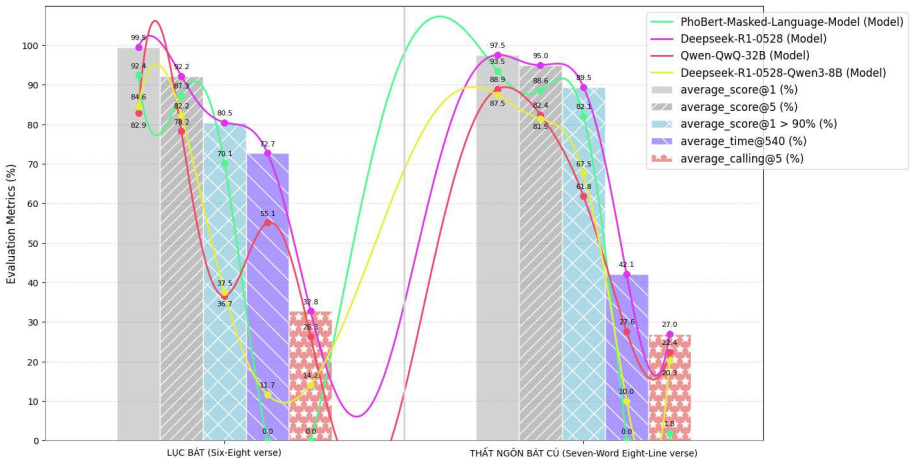
Formal and Semantic Integrity: This dimension is designed to quantify the generated text's adherence to the structural and prosodic rules of a specific poetic genre. It serves as a primary differentiator between a well-formed poem and a mere prose passage. Therefore, this is measured using the PCS (Poetic Rule Score or Poetic Conformity Score) metric [1] for both the "Lục Bát" and "Thất Ngôn Bát Cú" forms. An elevated PCS score indicates greater accuracy and better alignment with the rules of the respective poetic style.

Coherence and Interpretability: After establishing formal correctness, this set of metrics focuses on assessing the poem's linguistic and semantic quality. These metrics measure fluency, logical consistency, lexical richness, and semantic complexity. We employ a suite of standard metrics from the field of Natural Language Processing (NLP), including:

- Overlap-based Metrics: BLEU [43] and ROUGE [44] scores are used to compare the generated text against one or more reference texts, where available.
- Lexical diversity is a critical criterion for evaluating the quality of text generated by language models. In this study, we quantify a model's capacity for generating



(a) Graphical summary of averaged performance metrics of four candidate models in the Post-Processing Stage in ViVerse-A1 utilizing radar visualization.



(b) Evaluation metrics reveal the feasibility trends between 'Lục Bát' Style (Six-Eight Verse) and 'Thất Ngôn Bát Cú' Style (Seven-Word Eight-Line verse).

Fig. 6: Overview of results comparing the PhoBert-Masked-Language-Model and the M-Reasoning Language Model evaluated in the having-reasoning state-of-the-art model baselines. Both charts illustrate the overperformance achievement of the **DeepSeek-R1-0528 model** in not only implementing *masked-token* tasks effectively but also sustaining the inclusive semantics and emotional flow in enhancing Vietnamese poetry.

lexically rich and varied text using three primary metrics: the Lexical Richness [45] metric, the Maas index [46], and Simpson's D-index ($D_{Simpson}$) [47]. The evaluation is performed by comparing these metrics between model-generated poems and their original counterparts, which serve as the gold standard. To establish a single, composite comparative score, we propose the integration of the Gaussian Distance [48] formula into our scoring methodology. This approach transforms the disparity in lexical diversity into a single, normalized score within the [0,1] interval, thereby providing an explicit measure of the similarity in vocabulary richness between the generated and reference texts.

- Model Perplexity [49], [50], [51]: This metric measures the model's "surprise" when predicting the text; a lower value typically corresponds to a more fluent and natural-sounding text. These indices provide a quantitative assessment of the poem's overall expressive capability and semantic structure.

Stylistic Novelty and Imaginativeness: While the preceding metrics measure correctness and coherence, they often fall short of capturing the creative dimension or the "poetic essence" of a work. This dimension is therefore proposed to address this gap by measuring the ability to generate novel ideas, unique word combinations, and diversity in the use of semantic fields. Rather than focusing on structure, this metric analyzes the poem's thematic scope and semantic depth. Specifically, we evaluate the diversity of the semantic fields employed, which reflects the model's depth of "thought" and creative capacity.

3.5.2 Post-Processing Models Stage

To facilitate an objective and comprehensive comparison, we evaluated the performance of various post-processing models on the task of refining Vietnamese poetry. The performance of each model was quantified using five key metrics, as visually presented in Figure 6:

- Accuracy@1 ~ average_score@1 (%): The percentage of top-1 predictions that strictly adhere to the established poetic rules.
- Average Accuracy@5 ~ average_score@5 (%): The mean accuracy of the top-5 predictions.
- High-Confidence Rate@1 ~ average_score@1 $>$ 90% (%): The percentage of top-1 predictions achieving a poetic rule compliance score exceeding 90%.
- Latency Efficiency@540s ~ average_time@540 (%): The percentage of tasks completed within a 540-second threshold, where a higher value signifies faster processing.
- API Call Efficiency ~ average_calling@5 (%): A measure of resource utilization based on the number of model invocations. A higher value corresponds to fewer recalls, indicating the model's ability to generate high-quality results in its initial attempts.

The experimental procedure was conducted on two prevalent Vietnamese poetic forms: "Thất Ngôn Bát Cú" (a seven-word, eight-line verse) and "Lục Bát" (a six-eight word-pair verse). The analysis of the results reveals a distinct trade-off between output accuracy, processing efficiency, and resource utilization.

According to Figure 6, the evidence indicates that the DeepSeek-R1 model demonstrated unequivocal superiority in terms of output quality. Specifically, its Top-1 Accuracy reached exceptional levels for both genres: 99.5% for Lục bát and 97.5% for Thất ngôn bát cú, as shown in Figure 6b. Moreover, its rate of high-confidence results was substantially higher than that of the other evaluated models.

However, this exceptional performance is achieved at the cost of increased latency and resource consumption. DeepSeek-R1's efficiency metrics were comparatively lower, indicating that it requires longer processing times and a greater number of model invocations to achieve optimal outcomes. In stark contrast, the PhoBert model emerged as a highly efficient alternative. Although its accuracy did not match that of DeepSeek-R1 (achieving 92.4% for Lục bát and 93.5% for Thất ngôn bát cú), PhoBert completed tasks with near-instantaneous speed and minimal API recalls. The radar chart depicted in Figure 6a corroborates this finding, showing PhoBert's Latency Efficiency@540s and API Call Efficiency metrics approaching maximum values at 99.99% and 99.08%, respectively.

Within the scope of this research, the foremost objective is to maximize the quality and formal correctness of the generated poetry. While PhoBert represents an effective solution for applications where speed is a critical requirement, the superior accuracy of DeepSeek-R1 is more congruent with our research goals. Consequently, DeepSeek-R1 has been selected as the external Large Language Model (LLM) for the post-processing module of the ViVerse-A1 system. All evaluation metrics reported in this paper are based on this chosen model.

4 Experiments

This section presents a detailed, multifaceted analysis of the experimental outcomes achieved by the ViVerse-A1 model in Vietnamese poetry generation. The results reported herein provide compelling evidence of the model's performance and serve to highlight its superior capabilities.

4.1 Datasets and Experimental Setup

The fine-tuning of the Qwen2.5-7B model was conducted using experimental parameters analogous to those in the cited work [52]. Specifically, we employed a combination of Quantization [53] and Supervised Fine-Tuning (SFT) [54], [55], [56] on a dataset comprising approximately 200,000 poetry samples. This process required approximately 72 hours to complete a single epoch. We executed the fine-tuning procedure for a total of five epochs and saved the optimal model weights for the subsequent generation and evaluation phases. Further technical details regarding the fine-tuning process are provided in the supplementary document.

From the fine-tuned ViVerse-A1 model, we generated an output dataset of approximately 600 poems for evaluation. This dataset encompasses two primary poetic forms:

"Thất Ngôn Bát Cú" (seven-word, eight-line verse) and "Lục Bát" (six-eight verse). Thematically, the generated works are highly diverse, spanning from classical and traditional subjects imbued with national cultural identity to modern themes such as patriotism and the values of contemporary life.

Table 1: Quantitative Comparison of Formal and Semantic Integrity Across Distinct Vietnamese Poetic Styles

Models	**"Lục bát" Style**		**"Thất ngôn bát cú" Style**	
	PCS	PCS $>$ 90%	PCS	PCS $>$ 90%
GPT4o (Direct)	81.62	26.25	75.72	16.42
RAG-ViVerse	x	x	81.33	37.81
ViVerse-A1 (Ours)	**92.80** ↑	**100.00** ↑	**99.02** ↑	**97.22** ↑

4.2 Evaluation Metrics

Adopting the methodology of related research, we performed a comprehensive comparative analysis to ascertain the superior performance and artistic merit of the poems generated by ViVerse-A1. Our proposed model was benchmarked against the following baseline models: (1) GPT-4o: A state-of-the-art large language model, (2) RAG-ViVerse: A variant utilizing the Retrieval-Augmented Generation (RAG) methodology to enhance factual accuracy and semantic coherence, (3) ViVerse-A1: The primary model proposed in this study. The evaluation was conducted based on four principal criteria. The precise definitions and measurement methodologies for each criterion have been detailed in Section 3.5.1. Prior to delving into intricate specifics, it is important to note that the RAG-ViVerse model was specifically engineered for the Seven-word, Eight-line ("Thất Ngôn Bát Cú") poetic form. Consequently, a comparative performance evaluation of this model on the Six-eight ("Lục Bát") form was not conducted within the scope of this research. Non-evaluated items are denoted by an *"x"* in the corresponding results tables.

4.2.1 Formal and Semantic Integrity

To verify the model's adherence to established poetic conventions, we conducted a comparative performance evaluation of ViVerse-A1 against the baseline models. The results in Table 1, were measured using the Poetic Rule Score (PCS) (detailed in Section 3.5.1) and the percentage of outputs with a PCS exceeding 90%. The analysis reveals that the ViVerse-A1 model demonstrates superior performance in both poetic forms. In the "Lục Bát" form, for instance, our model (ViVerse-A1) achieved a considerably higher average PCS than the direct application of GPT4o, which reached a maximum PCS of only 81.61% with just 26.25% of its outputs scoring above 90%. This superiority was even more pronounced in the "Thất Ngôn Bát Cú" form, where ViVerse-A1 attained a near-perfect PCS of 99.02%. This score is approximately 24%

(a) Comparison of Coherence and Interpretability Metrics for Baselines and Proposed Models.

Metrics	**"Lục bát" Style**			**"Thất ngôn bát cú" Style**		
	GPT4o (Direct)	RAG-ViVerse	ViVerse-A1 (Ours)	GPT4o (Direct)	RAG-ViVerse	ViVerse-A1 (Ours)
BLEU-1	0.153	x	**0.193**	0.146	0.169	**0.193**
BLEU-2	0.098	x	**0.135**	0.106	0.112	**0.125**
BLEU-3	0.076	x	**0.113**	0.090	0.091	**0.125**
BLEU-4	0.057	x	**0.093**	0.073	0.072	**0.108**
ROUGE-1	0.165	x	**0.222**	0.164	0.175	**0.199**
ROUGE-2	0.065	x	**0.106**	0.079	0.077	**0.107**
ROUGE-L	0.143	x	**0.188**	0.153	0.145	**0.172**
Interpretability throughout Diversity and Complexity Rate						
Lexical Richness	0.974	x	**0.994**	0.965	0.986	**0.988**
Maas Index	0.994	x	**0.998**	**0.997**	0.996	**0.997**
Simpson's D	0.995	x	**0.999**	0.995	0.997	**0.998**
Perplexity	11.770	x	**8.948**	10.300	14.301	**8.291**

(b) Performance Comparison on Stylistic Novelty and Imaginativeness Across Poetic Forms and Models

Vocab Fields	**"Lục bát" Style**			**"Thất ngôn bát cú" Style**		
	GPT4o (Direct)	RAG-ViVerse	ViVerse-A1 (Ours)	GPT4o (Direct)	RAG-ViVerse	ViVerse-A1 (Ours)
N - Common noun	**19.854**	x	17.552	**20.105**	18.159	19.672
Nc - Noun Classifier	**1.304**	x	1.025	0.798	**1.015**	0.873
Ny - Noun abbreviation	0.000	x	0.000	0.000	0.035	0.000
Np - Proper noun	**0.275**	x	0.037	0.175	**1.592**	0.052
Nu - Unit noun	0.000	x	**0.004**	**0.026**	0.005	**0.026**
P - Pronoun	1.004	x	**1.440**	1.930	**2.557**	1.590
V - Verb	**11.871**	x	11.560	**14.456**	14.328	14.004
A - Adjective	**6.900**	x	5.751	5.009	4.716	**5.900**
I - Interjection	0.000	x	0.000	0.000	**0.005**	0.000
R - Adverb	1.683	x	**2.730**	3.316	**4.980**	3.786
C - Coordinating conjunction	0.633	x	**0.776**	0.746	**1.090**	0.616
S - Subordinating conjunction	0.000	x	0.000	0.000	0.000	0.000
E - Preposition	1.546	x	**1.838**	1.395	**1.766**	1.725
L - Determiner	0.271	x	**0.365**	**0.430**	0.403	0.345
M - Numeral	0.571	x	**0.593**	**0.860**	0.816	0.808
T - Auxiliary, modal words	**0.167**	x	0.124	0.088	**0.269**	0.083
X - Unknown	0.067	x	**0.079**	**0.263**	0.075	0.092
F - Filtered out (punctuation)	0.000	x	0.000	0.000	0.000	0.000

higher than GPT-4o (Direct) and 18% higher than RAG-ViVerse. Notably, 97.22% of the poems generated by ViVerse-A1 scored above 90% on the PCS, a rate three times higher than that of RAG-ViVerse and nearly four times higher than GPT-4o (Direct). These findings affirm ViVerse-A1's capability to generate Vietnamese poetry that strictly adheres to the structural and prosodic rules of both investigated forms.

4.2.2 Coherence and Interpretability

The statistics are shown as follows in Table 2a and indicate that ViVerse-A1 surpasses the baseline models in terms of coherence and interpretability. The inferior scores of GPT-4o and RAG-ViVerse highlight the superior stability of our proposed model in generating highly connected and creative verses. The ViVerse-A1 model not only preserves the core semantics and sentiment of the poetic theme but also demonstrates significant lexical diversity. The Lexical Richness, Maas Index, and Simpson's D metrics particularly highlight phonological vocabulary and abundance, demonstrating the capacity for creativity in adapting complex structures while simultaneously instilling and consistently preserving core values related to semantics, the linguistic context of poetry, and the morphology of word connections within a poem. This ability is evident in the "Lục Bát" forms, which exhibit scores of 0.994, 0.998, and 0.999 for the three aforementioned metrics, respectively. Moreover, all "Thất Ngôn Bát Cú" forms surpass the other models, attaining around 0.988 for Lexical Richness, 0.997 (equivalent to GPT-4o Direct), and 0.998 for the Maas Index and Simpson's D metrics. Critically, its Perplexity (PPL) scores are exceptionally low, at 8.948 and 8.921 for the "Lục Bát" and "Thất Ngôn Bát Cú" forms, respectively. These values are substantially lower than those of other models (all with PPL > 10), indicating that the outputs of ViVerse-A1 possess greater fluency and predictability, thereby minimizing randomness and incomprehensibility.

4.2.3 Stylistic Novelty and Imaginativeness

Evaluating stylistic novelty and imaginativeness, these are displayed across the analysis of the Part-of-Speech (POS) distribution of the generated poems in the Vietnamese language, with the results detailed in Table 2b. The analysis revealed that GPT-4o tends to concentrate on primary word classes such as nouns, verbs, and adjectives when generating "Lục Bát" poetry. In contrast, ViVerse-A1 demonstrates a more balanced POS distribution, making effective use of auxiliary word classes like adverbs and conjunctions, which facilitates smoother semantic transitions and enhances expressive depth. Similarly, in the context of the "Thất Ngôn Bát Cú" form, ViVerse-A1's POS distribution strikes an effective balance between that of RAG-ViVerse and GPT-4o. By avoiding over-concentration on any specific word class and instead leveraging a broad vocabulary spectrum, the model enriches the stylistic texture of the poetry while preserving the characteristic prosody of the form. Collectively, this analysis indicates that ViVerse-A1 is capable of producing poetic works that are not only metrically compliant but also lexically diverse and expressively nuanced.

5 Discussion and conclusion

Model Scope and Specialization. ViVerse-A1, a large language model trained only on annotated Vietnamese poetry, consistently outperforms both GPT-4o and RAG-ViVerse (which adds external retrieval to generation) across all evaluation dimensions.

Following the rules of poetry. ViVerse-A1 achieved a mean Poetic Conformity Score (PCS) of 99.02% in the "Thất Ngôn Bát Cú" test. In the "Lục Bát" setting, 97.22% of its outputs were better than the 90% PCS threshold. These results show that ViVerse-A1 is very good at using meter, rhyme schemes, and structural conventions in both types of poetry.

Consistency and predictability. The verses that ViVerse-A1 generates are obvious and easy to understand. For example, "Lục Bát" got a perplexity score of 8.948, and "Thất Ngôn Bát Cú" got a score of 8.921. On the other hand, other models recorded values over 10. The lower perplexity indicates that ViVerse-A1's outputs are semantically stable and exhibit minimal randomness.

Stylistic Diversity and Expressiveness. ViVerse-A1 has a wider range of styles. Part-of-speech analysis reveals that nouns, verbs, adjectives, and auxiliary classes (such as adverbs and conjunctions) are used in a balanced manner, with a tendency to favor nouns within the poem's context. Such usage makes semantic transitions smoother and gives the writing more depth. On the other hand, GPT-4o likes core word classes more, and RAG-ViVerse is in the middle of the two.

In short, ViVerse-A1 gets the best results in Vietnamese poetry generation by fine-tuning a general-purpose LLM on carefully chosen poetic data. These results show how important it is to adapt to a specific domain for creative tasks with strict formal requirements. They also point to promising areas for further research in text generation that is culturally grounded.

6 Limitations

While the proposed model has achieved commendable results, a thorough examination of its inherent limitations is necessary for a comprehensive perspective. A primary challenge of the model lies in its computational cost and resource requirements. The architecture of ViVerse-A1, which integrates a fine-tuning stage on a specialized dataset and builds upon the RAG-ViVerse foundation, demands substantially greater hardware resources and processing time compared to preceding methods. Furthermore, the fine-tuning costs increase linearly with the expansion of the knowledge base, posing a significant challenge to the model's long-term scalability.

The model's performance is also directly contingent upon the diversity of the corpus used for training. Consequently, ViVerse-A1's generalization capability may be constrained when applied to poetic forms not represented in its training data. The present study focuses exclusively on two representative Vietnamese forms, "Lục Bát" and "Thất Ngôn Bát Cú," and therefore, the model's efficacy on other poetic genres remains unverified.

Additionally, the post-processing stage of the pipeline is heavily reliant on the capabilities of the underlying reasoning-based model. The use of a reasoning model

that is not optimized for the Vietnamese language could severely degrade the quality and validity of the final output, thereby rendering this critical stage ineffective.

Finally, the current evaluation methodology is limited by its incomplete integration of metrics from human evaluation as noted in [57]. The assessment of poetic quality, particularly for Vietnamese poetry, is a highly subjective endeavor that requires evaluators with a profound understanding of the prosody and specific characteristics of each genre. This subjectivity complicates the construction of an entirely objective and reliable set of evaluation criteria.

Acknowledgements

Son T. Huynh was funded by the PhD Scholarship Programme of Vingroup Innovation Foundation (VINIF), code VINIF.2025.TS.02.

Statements and Declarations

Conflict of Interest
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding
Son T. Huynh was funded by the PhD Scholarship Programme of Vingroup Innovation Foundation (VINIF), code VINIF.2025.TS.02.

Ethics Approval and Consent to Participate
This research does not involve human participants or animal subjects. Therefore, ethical approval and consent to participate were not required.

Consent for publication
Not applicable.

Data availability
The datasets analyzed during the current study are publicly available from the resources cited within this manuscript.

Code availability
The source code used to implement the proposed methods will be made publicly available in a designated project repository upon the publication of this paper.

Author contribution
All authors contributed to the study. Specific contributions are as follows:

Phu G. Lam: Conceptualization, Methodology, Data curation, Software, Writing – Original Draft.
Hung P. Tran: Conceptualization, Data Analysis, Data Preprocessing, Methodology, Software, Visualization, Writing – Original Draft.
Tram T. Doan and **Binh T. Nguyen:** Validation, Review, Supervision.

Son T. Huynh: Conceptualization, Methodology, Validation, Review, Supervision.

Declaration of Third-Party Generative AI in the Writing Process
Generative AI tools were used during the preparation of this manuscript to assist with language refinement and clarity. The authors reviewed and revised the output as necessary and assume full responsibility for the accuracy and integrity of the content.

Appendix A Examples of Vietnamese Poetry Generation

This section A presents a comparative analysis of the final outputs produced by various models when generating poetry in two traditional Vietnamese poetic forms: "Thất ngôn bát cú" in Table A1 and "Lục bát" in Table A2.

Note: The positions highlighted in red indicate incorrect rhymes in the verses.

A.1 "Thất ngôn bát cú" genre

As shown in Table A1, we compare three approaches for generating Vietnamese poetry in the seven-word eight-line form. The table presents complete poem outputs along with their respective Poetic Rule Scores, where ViVerse-A1 achieves a perfect score of 100, significantly outperforming both RAG-Viverse (88.88) and Direct GPT-4o (62.22).

Table A1: Example Comparison of ViVerse-A1, RAG-Viverse, and Direct GPT-4o in Vietnamese Poetry Generation in "Thất ngôn bát cú" genre

Task: Hãy viết tiếp bài thơ với câu đầu là **Quê tôi khóm cúc vẫn lên hoa** theo thể loại **thất ngôn bát cú** và chủ đề **tình yêu, tâm trạng**. *(Write the next lines of a poem beginning with **"Back home, chrysanthemums still rise in bloom"**, using the **Seven-Word Eight-Line Poem** and exploring the themes of **love and emotion**.)*
ViVerse-A1 output: Poetic Rule Score = 100

Quê tôi khóm cúc vẫn lên hoa
(Back home, the chrysanthemums still bloom)
Ngọn gió đưa hương khắp mọi nhà
(The breeze carries fragrance to every home)
Thơm mát lòng người hưng hửng nắng
(Fragrant and cool, the sunshine lifts the heart)
Tươi cười cảnh vật xốn xang hoa
(Nature beams with joy, flowers flutter with delight)
Đường vui nhộn nhịp người về chợ
(The roads bustle cheerfully as people head to the market)
Phố đẹp xinh tươi khách đón hoa
(The lovely streets greet guests with blooming flowers)
Cô lữ tình duyên dìu bước ngọc
(A lone girl, guided by love, walks with graceful steps)
Quê ơi thắm mãi một cành hoa
(Oh my homeland, forever bright with a single flower)

RAG-Viverse output:
Poetic Rule Score = 88,88

Quê tôi ngàn khóm cúc ngàn hoa
(Back home, a thousand chrysanthemums still bloom)
Đượm hồn thu thấm sắc vàng hoa
(Autumn's soul imbues, saturates the golden hue of blossoms)
Gió lộng vương hương lạc bước ngàn
(The wind carries fragrance adrift for a thousand paces)
Mờ nắng phiêu phiếu vẹn bờ hoa
(Dim sunlight drifts over the intact banks of flowers)
Tim sầu khẽ thổn thức nhớ ngươi
(My sorrowful heart gently throbs, remembering you)
Lệ rơi chạnh lòng vấn vương hoa
(Tears fall, my heart yearns for the blossoms)
Đêm khuya thả giấc mộng nghìn sao
(Late at night, I release dreams into a thousand stars)
Quê hương lưu bóng ngọt ngào hoa
(My homeland preserves the flowers' sweet essence)

Direct GPT-4o output:
Poetic Rule Score = 62,22

Quê tôi khóm cúc vẫn lên hoa,
(Back home, the chrysanthemums still bloom bright)
Tỏa sắc vàng tươi mùi hương tỏa.
(Their golden hue spreads across the land)
Mưa đổ giọt nhẹ trên lá cỏ,
(Rain falls gently upon the blades of grass)
Gió quyện hương thơm khắp lá hoa.
(The wind carries fragrance through every leaf and home)
Cây đa gốc thắm tình sông núi,
(The old banyan tree holds love for land and river)
Lúa chín rực rỡ hát mình ơi.
(The ripe rice fields sing in radiant joy)
Nhớ về dĩ vãng thời gian ấy,
(I recall the past from those olden days)
Thương mến quê mình mãi thiết tha.
(Forever I love my dear homeland with deep affection)

A.2 "Lục bát" genre

Table A2 provides a comparison of model performance on the "Lục bát" verse form. Notably, RAG-Viverse demonstrates limitations in this genre due to structural challenges in masked poem generation, while ViVerse-A1 maintains perfect adherence to poetic rules with a score of 100, compared to Direct GPT-4o's score of 53.31.

Table A2: Example Comparison of ViVerse-A1, RAG-Viverse, and Direct GPT-4o in Vietnamese Poetry Generation in "Lục bát" genre

Task: Hãy viết tiếp bài thơ với câu đầu là **cho tôi mượn nửa bờ vai** theo thể loại **lục bát** và chủ đề **tình yêu, tâm trạng**. *(Write the next lines of a poem beginning with **"let me borrow just half your shoulder"**, using the **Six-eight verse form** and exploring the themes of **love and emotion**.)*
ViVerse-A1 output: Poetic Rule Score = 100

cho tôi mượn nửa bờ vai
(let me borrow just half your shoulder)
xin về gối mái tóc dài đợi trao
(to rest on your hair, where dreams softly gather)
bờ vai rất đỗi ngọt ngào
(your shoulder—tender, sweet as ever)
vai em thao thức gầy hao tháng ngày
(worn thin through nights and drifting weather)
để tôi nghe tiếng lòng này
(so I may hear your heart's deep sigh)
tiếng em da diết như say mơ chờ
(your voice—like longing in a dream gone by)
tình mình một thuở còn thơ
(our love once bloomed when we were young)
thời gian vùn vụt mãi chờ vơi đầy
(time rushed past—still waiting, full then gone)

RAG-Viverse may perform poorly in this genre, likely because the Masked Poem generation fails to conform to the structural rules of poetry.

Direct GPT-4o output:
Poetic Rule Score = 53.31

Cho tôi mượn nửa bờ vai,
(let me borrow just half your shoulder,)
Để tôi dựa nhẹ những ngày mệt nhoài.
(So I may lean on it through weary days.)
Giữa đời sóng gió mênh mông,
(In this vast life of tempests and waves,)
Xin cho tôi chút ấm lòng, bạn ơi.
(Grant me some warmth, dear friend, I pray.)
Nửa vai để ngả giấc mơ,
(Half a shoulder to rest my dreams on,)
Cùng nhau san sẻ nỗi bơ vơ gầy.
(To share the frailty of our lone days.)
Tình bạn đâu cần thật nhiều,
(Friendship needs not wealth or show,)
Chỉ cần nửa bến che chiều bão giông.
(Just a small harbor to shield the storm's blow.)

References

[1] Zhang, X., Lapata, M.: Chinese poetry generation with recurrent neural networks. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 670–680. Association for Computational Linguistics, ??? (2014)

[2] Dang, N.H., Thinh, N.T.: Using of recurrent neural network with a refine model to compose vietnamese luc-bat poems for human-robot verbal interactions. Journal of Advances in Information Technology **14**(5) (2023)

[3] Chatzikyriakidis, S., Natsina, A.: Poetry in rags: Modern greek interwar poetry generation using rag and contrastive training. In: Proceedings of the 5th International Conference on Natural Language Processing for Digital Humanities, pp. 257–264 (2025)

[4] Jiang, Z., Xu, F.F., Gao, L., Sun, Z., Liu, Q., Dwivedi-Yu, J., Yang, Y., Callan, J., Neubig, G.: Active retrieval augmented generation. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 7969–7992 (2023)

[5] Deng, J., Pang, L., Shen, H., Cheng, X.: Regavae: A retrieval-augmented gaussian mixture variational auto-encoder for language modeling. arXiv preprint arXiv:2310.10567 (2023). arXiv:2310.10567

[6] He, X., Tian, Y., Sun, Y., Chawla, N., Laurent, T., LeCun, Y., Bresson, X., Hooi, B.: G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. Advances in Neural Information Processing Systems **37**, 132876–132907 (2024)

[7] Gan, C., Yang, D., Hu, B., Zhang, H., Li, S., Liu, Z., Shen, Y., Ju, L., Zhang, Z., Gu, J., et al.: Similarity is not all you need: Endowing retrieval augmented generation with multi-layered thoughts. arXiv preprint arXiv:2405.19893 (2024). arXiv:2405.19893

[8] Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y.J., Madotto, A., Fung, P.: Survey of hallucination in natural language generation. ACM Computing Surveys **55**(12), 1–38 (2023)

[9] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)

[10] Popescu-Belis, A., Atrio, A.R., Bernath, B., Boisson, É., Ferrari, T., Theimer-Lienhardt, X., Vernikos, G.: Gpoet: A language model trained for rhyme generation on synthetic data. In: Proceedings of the 7th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities

and Literature. Association for Computational Linguistics, ??? (2023)

[11] Ormazabal, A., Artetxe, M., Agirrezabal, M., Soroa, A., Agirre, E.: Poelm: A meter-and rhyme-controllable language model for unsupervised poetry generation. arXiv preprint arXiv:2205.12206 (2022). arXiv:2205.12206

[12] Panahandeh, A., Asemi, H., Nourani, E.: Tppoet: Transformer-based persian poem generation using minimal data and advanced decoding techniques. arXiv preprint arXiv:2312.02125 (2023). arXiv:2312.02125

[13] Tian, H., Yang, K., Liu, D., Lv, J.: Anchibert: A pre-trained model for ancient chinese language understanding and generation. In: 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE, ??? (2021)

[14] Beheitt, M.E.G., Ben Haj Hmida, M.: Automatic arabic poem generation with gpt-2. In: Proceedings of the 14th International Conference on Agents and Artificial Intelligence (ICAART), vol. 2, pp. 366–374 (2022)

[15] Huynh, T.M., Bao, Q.L.: Vietnamese poem generation & the prospect of cross-language poem-to-poem translation. arXiv preprint arXiv:2401.01078 (2024). arXiv:2401.01078

[16] Zou, X., Yin, D., Zhong, Q., Yang, H., Yang, Z., Tang, J.: Controllable generation from pre-trained language models via inverse prompting. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, pp. 2450–2460. ACM, ??? (2021)

[17] Tian, Y., Peng, N.: Zero-shot sonnet generation with discourse-level planning and aesthetics features. arXiv preprint arXiv:2205.01821 (2022). arXiv:2205.01821

[18] Belouadi, J., Eger, S.: Bygpt5: End-to-end style-conditioned poetry generation with token-free language models. arXiv preprint arXiv:2212.10474 (2022). arXiv:2212.10474

[19] Qwen Team: Qwen2.5: A Party of Foundation Models. Online; accessed 2024 (2024). https://qwenlm.github.io/blog/qwen2.5/

[20] Yang, A., Yang, B., Hui, B., Zheng, B., Yu, B., Zhou, C., Li, C., Li, C., Liu, D., Huang, F., Dong, G., Wei, H., Lin, H., Tang, J., Wang, J., Yang, J., Tu, J., Zhang, J., Ma, J., Xu, J., Zhou, J., Bai, J., He, J., Lin, J., Dang, K., Lu, K., Chen, K., Yang, K., Li, M., Xue, M., Ni, N., Zhang, P., Wang, P., Peng, R., Men, R., Gao, R., Lin, R., Wang, S., Bai, S., Tan, S., Zhu, T., Li, T., Liu, T., Ge, W., Deng, X., Zhou, X., Ren, X., Zhang, X., Wei, X., Ren, X., Fan, Y., Yao, Y., Zhang, Y., Wan, Y., Chu, Y., Liu, Y., Cui, Z., Zhang, Z., Fan, Z.: Qwen2 Technical Report. arXiv technical report / project technical note; see project page (2024)

[21] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou,

D., *et al.*: Chain-of-thought prompting elicits reasoning in large language models. In: Advances in Neural Information Processing Systems, vol. 35, pp. 24824–24837 (2022)

[22] Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Cui, C., Bousquet, O., Le, Q., et al.: Least-to-most prompting enables complex reasoning in large language models. arXiv preprint arXiv:2205.10625 (2022). arXiv:2205.10625

[23] Dhuliawala, S., Komeili, M., Xu, J., Raileanu, R., Li, X., Celikyilmaz, A., Weston, J.: Chain-of-verification reduces hallucination in large language models. arXiv preprint arXiv:2309.11495 (2023). arXiv:2309.11495

[24] Hao, S., Gu, Y., Ma, H., Hong, J.J., Wang, Z., Wang, D.Z., Hu, Z.: Reasoning with language model is planning with world model. arXiv preprint arXiv:2305.14992 (2023). arXiv:2305.14992

[25] Zhang, Z., Zhang, A., Li, M., Zhao, H., Karypis, G., Smola, A.: Multimodal chain-of-thought reasoning in language models. arXiv preprint arXiv:2302.00923 (2023). arXiv:2302.00923

[26] Debnath, T., Siddiky, M.N.A., Rahman, M.E., Das, P., Guha, A.K.: A comprehensive survey of prompt engineering techniques in large language models. TechRxiv (2025)

[27] Li, X., Zhao, R., Chia, Y.K., Ding, B., Joty, S., Poria, S., Bing, L.: Chain-of-knowledge: Grounding large language models via dynamic knowledge adapting over heterogeneous sources. arXiv preprint arXiv:2305.13269 (2023). arXiv:2305.13269

[28] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI Blog. OpenAI Blog, technical report (2019)

[29] Nguyen, T., Nguyen, P., Pham, H., Bui, T., Nguyen, T., Luong, D.: Sp-gpt2: Semantics improvement in vietnamese poetry generation. In: 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 1576–1581. IEEE, ??? (2021)

[30] Yu, C., Zang, L., Wang, J., Zhuang, C., Gu, J.: Charpoet: A chinese classical poetry generation system based on token-free llm. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pp. 315–325 (2024)

[31] Huynh, S.T., Lam, P.G., Tran, H.P., Nguyen, B.T.: Rag-viverse: Enhancing poetry generation with retrieval augmented generation and large language models. In: International Conference on Computational Collective Intelligence, pp.

215–227. Springer, ??? (2025)

[32] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-augmented generation for knowledge-intensive NLP tasks. Proceedings of the 34th International Conference on Neural Information Processing Systems. NeurIPS 2020 (2020)

[33] Hochreiter, S.: Long short-term memory. Neural Computation (1997). MIT Press

[34] Anisuzzaman, D.M., Malins, J.G., Friedman, P.A., Attia, Z.I.: Fine-tuning large language models for specialized use cases. Mayo Clinic Proceedings: Digital Health **3**(1), 100184 (2025) https://doi.org/10.1016/j.mcpdig.2024.11.005 . doi:10.1016/j.mcpdig.2024.11.005

[35] Qwen Team: Qwen2.5: A Party of Foundation Models. Online (2024). https://qwenlm.github.io/blog/qwen2.5/

[36] Kamble, K., Alshikh, W.: Palmyra-med: Instruction-based fine-tuning of llms enhancing medical domain performance. Palmyra-Med: Instruction-based fine-tuning of LLMs enhancing medical domain performance (2023)

[37] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021). arXiv:2106.09685

[38] Han, D., Han, M., Unsloth Team: Unsloth. Online (2023). https://github.com/unslothai/unsloth

[39] Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., Hashimoto, T.B.: Stanford Alpaca: An instruction-following LLaMA model. Online (2023). https://github.com/tatsu-lab/stanford_alpaca

[40] Nguyen, D.Q., Nguyen, A.T.: Phobert: Pre-trained language models for vietnamese. In: Findings of EMNLP (2020)

[41] Qwen Team: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025). arXiv:2505.09388

[42] DeepSeek-AI: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948 (2025). arXiv:2501.12948

[43] Tabassam, M.R., Waheed, H., Safder, I., Sarwar, R., Aljohani, N.R., Nawaz, R., Hassan, S.-U., Zaman, F., Ahsan, A.: Upon: Urdu poetry generation using deep learning: A novel approach and evaluation. ACM Transactions on Asian and Low-Resource Language Information Processing (2024). Retrieved from https://dl.acm.org/doi/pdf/10.1145/3708535

[44] Lin, C.-Y.: Rouge: A package for automatic evaluation of summaries. In: Text Summarization Branches Out, pp. 74–81 (2004). ACL Workshop W04-1013, https://aclanthology.org/W04-1013.pdf

[45] Siallagan, E.A., Alfina, I.: Poetry generation for indonesian pantun: Comparison between seqgan and gpt-2. Jurnal Ilmu Komputer dan Informasi **16**(1), 59–67 (2023)

[46] Hamat, A.: The language of ai and human poetry: A comparative lexicometric study. 3L, Language, Linguistics, Literature **30**(2) (2024)

[47] Gregorius, H.-R., Gillet, E.M.: Generalized simpson-diversity. Ecological Modelling **211**(1–2), 90–96 (2008)

[48] Wang, J., Xu, C., Yang, W., Yu, L.: A normalized gaussian wasserstein distance for tiny object detection. arXiv preprint arXiv:2110.13389 (2021). arXiv:2110.13389

[49] Shannon, C.E.: A mathematical theory of communication. The Bell System Technical Journal **27**(3), 379–423 (1948). Nokia Bell Labs

[50] Lo, K.-L., Ariss, R., Kurz, P.: Gpoet-2: A gpt-2 based poem generator. arXiv preprint arXiv:2205.08847 (2022). arXiv:2205.08847

[51] Meister, C., Cotterell, R.: Language model evaluation beyond perplexity. arXiv preprint arXiv:2106.00085 (2021). arXiv:2106.00085

[52] Hùng, T.P., Phú, L.G.: ViVerse-Datasets: Published datasets for evaluating performance of Vietnamese generative poetry model. Hugging Face dataset (2025). https://huggingface.co/datasets/PuLam/RAG-Viverse

[53] Li, Y., Yu, Y., Liang, C., He, P., Karampatziakis, N., Chen, W., Zhao, T.: Loftq: Lora-fine-tuning-aware quantization for large language models. arXiv preprint arXiv:2310.08659 (2023). arXiv:2310.08659

[54] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., *et al.*: Lora: Low-rank adaptation of large language models. In: International Conference on Learning Representations (ICLR) (2022). ICLR publication / preprint

[55] Dong, G., Yuan, H., Lu, K., Li, C., Xue, M., Liu, D., Wang, W., Yuan, Z., Zhou, C., Zhou, J.: How abilities in large language models are affected by supervised fine-tuning data composition. arXiv preprint arXiv:2310.05492 (2023). arXiv:2310.05492

[56] Jin, X., Ren, X.: Demystifying language model forgetting with low-rank example associations. arXiv preprint arXiv:2406.14026 (2024). arXiv:2406.14026

[57] Zhao, S., Zhou, Y., Ren, Y., Chen, Z., Jia, C., Fang, Z., Long, Z., Liu, S., Lan, M.: Fuxi: A benchmark for evaluating language models on ancient chinese text understanding and generation. arXiv preprint arXiv:2503.15837 (2025). arXiv:2503.15837