POSTS AND TELECOMMUNICATIONS INSTITUTE OF TECHNOLOGY

HO CHI MINH CITY

INFORMATION TECHNOLOGY DEPARTMENT

Assignment Report

Topic: Speech-to-Speech Conversation with Large Language Models.

Course: Natural Language Processing

Students:

N21DCCN080 — Thien Hoang Nguyen

N21DCCN018 — Dung Huu Nguyen

N21DCVT015 — Diep Nguyen-Hong Vo

Supervisor:

Dr. Hai Thi-Tuyet Nguyen

June 5, 2025

Contents

List of Tables

List of Figures

1 Introduction

In the rapidly evolving landscape of artificial intelligence, we are witnessing a transformative era where machine learning systems have transcended traditional computational boundaries to achieve human-like proficiency in language understanding and generation. The advent of large language models, powered by sophisticated transformer architectures and trained on vast multilingual datasets, has revolutionized how we approach cross-linguistic communication, turning what was once the exclusive domain of skilled human translators into an increasingly automated and accessible technological service. Today's AI-driven translation systems demonstrate remarkable capabilities in capturing semantic nuances, cultural contexts, and linguistic subtleties that were previously thought impossible for machines to comprehend.

However, the current paradigm of translation technology is dominated by resource-intensive models requiring massive computational infrastructure, extensive GPU clusters, and enormous energy consumption, creating significant barriers to widespread adoption, particularly in developing regions and resource-constrained environments. The emergence of parameter-efficient fine-tuning techniques, such as Low-Rank Adaptation (LoRA) and other innovative approaches, represents a paradigm shift toward democratizing advanced translation capabilities by enabling the development of specialized, lightweight models that maintain competitive performance while operating within limited computational budgets. This technological evolution is particularly significant for speech-to-speech translation systems, which represent the pinnacle of real-time cross-linguistic communication by seamlessly integrating automatic speech recognition, neural machine translation, and text-to-speech synthesis into unified, conversational interfaces. Such systems hold immense potential across diverse application domains: from facilitating international business negotiations where precise communication can determine multi-million dollar deals, to enabling educational exchanges that break down language barriers in global learning environments, supporting healthcare delivery in multicultural societies where accurate medical communication can be life-saving, enhancing tourism experiences by enabling natural conversations between visitors and locals, streamlining emergency response operations where rapid cross-linguistic coordination is critical, and facilitating diplomatic communications where linguistic precision carries geopolitical significance. Moreover, these technologies promise to bridge the digital divide by making high-quality translation services accessible to underserved communities and developing economies that have historically lacked access to professional interpretation services, potentially unlocking new opportunities for global economic participation, cultural exchange, and knowledge sharing on an unprecedented scale.

2 Literature Review

2.1 Large Language Models

With the rise of highly parallelizable transformer architectures, pre-trained language models (PLMs) have shown exceptional abilities in understanding, generating, and manipulating natural language [7, 9]. These models are initially trained on large text corpora and are commonly fine-tuned for specific downstream tasks. As PLMs have scaled up, they have evolved into what are now known as Large Language Models (LLMs) [10, 2, 8], which contain significantly more parameters and are trained on vast amounts of data. As a result, LLMs exhibit an improved ability to learn complex language patterns and structures, alongside strong reasoning skills, leading to impressive performance across a wide range of natural language processing tasks [15, 18]. Despite their advantages, LLMs can sometimes generate outputs that are coherent but factually incorrect, and they may struggle with tasks that require deep domain-specific knowledge [6]. To address these limitations, the emerging field of Augmented Language Models (ALMs) aims to enhance conventional LLMs [2] by equipping them with stronger reasoning capabilities and the ability to utilize external tools and resources [6]. Reasoning in this context involves decomposing complex problems into smaller, more manageable sub-tasks, which LLMs can solve either independently or with the help of tools.

Additionally, LLMs can invoke external models or APIs to assist with specific tasks. For instance, ToolFormer [12] integrates API call tags into text sequences, enabling LLMs to access external tools. Visual ChatGPT [17] combines ChatGPT with Visual Foundation Models (VFMs) such as Transformers, ControlNet, and Stable Diffusion, enabling chat-based visual generation. HuggingGPT [13] connects ChatGPT with the Hugging Face model hub to handle various AI tasks using task-specific models. Similarly, ChatGPT for Robotics [16] employs prompt engineering to adapt ChatGPT for a broad range of robotics applications.

Several open-source projects also explore this direction. For example, BabyAGI and AutoGPT are GitHub-based initiatives that automate task planning and execution. Notably, AutoGPT [14] functions as an autonomous agent capable of setting high-level goals, decomposing them into sub-tasks, and iteratively working toward their completion. Augmented language models may leverage these enhancements individually or combine them in specific sequences to complete complex tasks, ultimately achieving superior generalization and adaptability. In recent months, Google released a technical report on Gemma3 [3], highlighting its advanced techniques and superior performance. Owing to its effectiveness and low computational cost, Gemma3 was adopted in our method.

2.2 Parameter-Efficient Fine-Tuning

Fine-tuning remains a critical approach for adapting large language models (LLMs) to new tasks and domain-specific datasets. However, as the size of these models has increased dramatically, from 1.5 billion parameters in GPT-2 to 175 billion in GPT-3, the traditional paradigm of full fine-tuning has become increasingly inefficient. This method typically requires extensive computational resources, often involving thousands of GPUs operating in parallel, which makes it unsustainable for many practical applications. To address these limitations, parameter-efficient fine-tuning (PEFT) techniques [4] have been introduced. These methods aim to achieve comparable or superior performance to full fine-tuning while updating only a small subset of the model's parameters, thus significantly reducing computational and storage costs. The appeal of PEFT has grown alongside the success of large-scale pre-trained models in both vision and multimodal domains, where effective transfer learning has enabled adaptation from large datasets to smaller, task-specific ones. This ability to generalize across domains further highlights the value of efficient fine-tuning strategies. PEFT methods can be categorized into four main types based on their underlying mechanisms. Additive fine-tuning introduces small, tunable components to the model, such as adapters or soft prompts, which are trained while the rest of the model remains frozen. In contrast, selective fine-tuning updates only a chosen subset of the model's original parameters, either through unstructured masking—where parameters are selected at random or based on importance—or through structured masking, which targets specific layers or components within the architecture.

Reparameterized fine-tuning modifies the model by injecting additional low-rank trainable matrices during training, which are later merged with the original weights at inference. Techniques like low-rank decomposition and LoRA (Low-Rank Adaptation) [5] fall under this category and are widely adopted for their balance of efficiency and effectiveness. In this work, we adopt LoRA due to its ability to significantly reduce training overhead while preserving model performance. Finally, hybrid fine-tuning methods explore the design space between these categories, combining multiple PEFT strategies to leverage their respective strengths and achieve enhanced adaptability and performance.

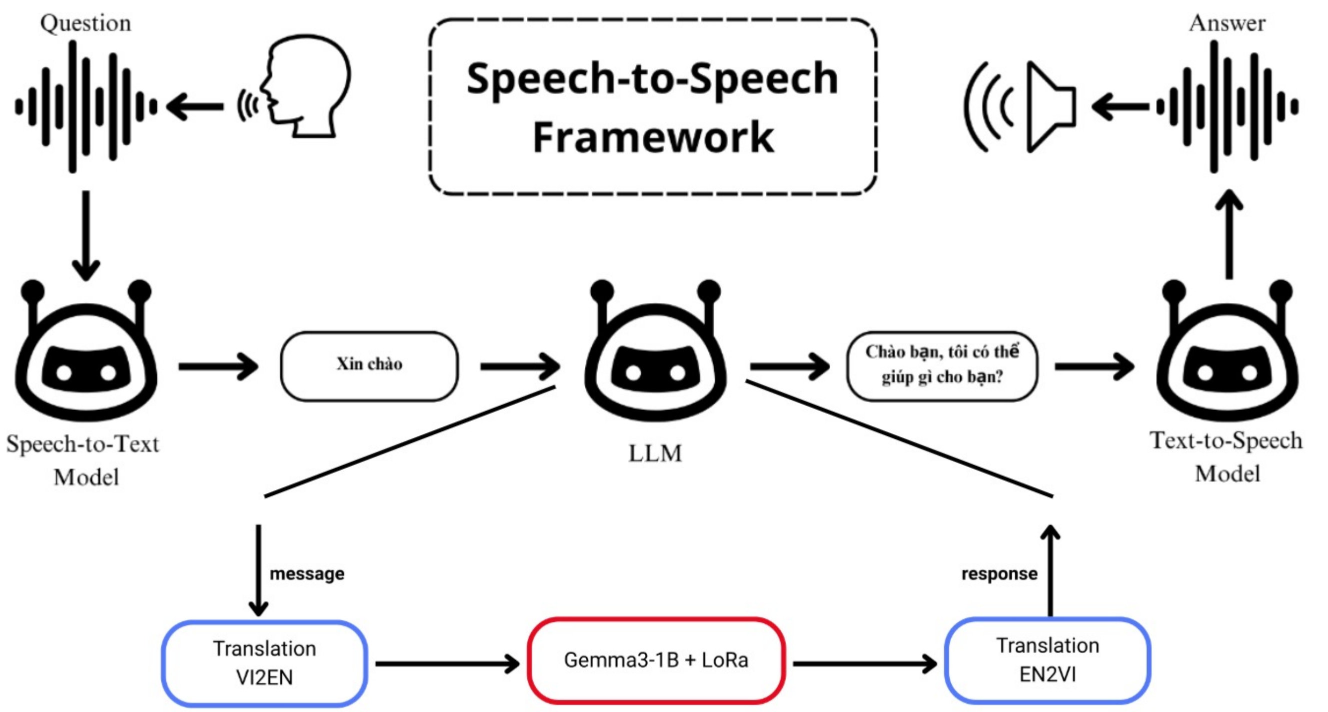

Figure 1: Demonstration for our Speech2Speech conversation pipeline

3 Methodology

We propose a speech-to-speech conversation pipeline designed to facilitate natural and efficient cross-lingual communication. The process begins with the user's spoken input, which is first processed by a Speech-to-Text (STT) module to obtain its textual representation. This Vietnamese text then undergoes a two-step procedure involving translation and response generation, where a large language model (LLM) enhanced with LoRA fine-tuning generates an appropriate reply. Finally, the generated textual response is converted back into speech through a Text-to-Speech (TTS) module, delivering an audible answer to the user. This entire pipeline is illustrated in Figure 1, showcasing the seamless integration of speech recognition, translation, language understanding, and speech synthesis components to enable smooth conversational interactions.

3.1 Gemma-3 Architecture

Gemma3 is a recently released open-weight language model by Google, designed to push the boundaries of efficiency, scalability, and generalization in large language models. Architecturally, Gemma3 adopts a decoder-only transformer structure similar to other state-of-the-art LLMs, but integrates several notable advancements that significantly enhance its performance across a wide range of tasks. A central innovation is the use of grouped-query attention (GQA), visualized in Figure 2, which offers a compelling trade-off between computational efficiency and attention expressiveness. By allowing multiple attention heads to share key and value projections while maintaining separate queries, GQA reduces the memory and compute footprint associated with full multi-head attention, especially beneficial when processing long sequences. In addition to GQA, Gemma3 leverages rotary position embeddings (RoPE), a position encoding technique that better preserves relative positional information and allows the model to extrapolate to longer contexts without retraining. These design choices enable Gemma3 to maintain high performance in both short-form and long-form reasoning scenarios.

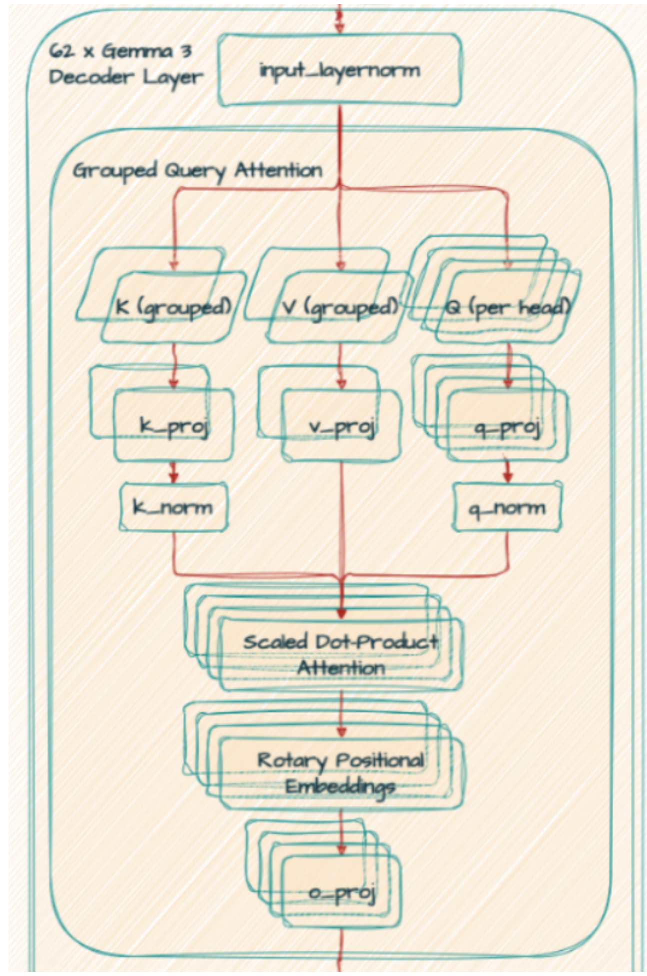

Figure 2: Gemma-3 architecture with Grouped Query Attention (GQA)

To further enhance scalability and stability, Gemma3 incorporates a series of low-level optimizations in both its model structure and training pipeline. It employs multi-query attention (MQA) in certain components of the architecture to reduce inference latency, particularly in deployment settings where speed and memory efficiency are critical. Layer normalization is implemented with high numerical precision, mitigating training instability in large models and promoting smoother convergence. Moreover, Gemma3 benefits from fused operations and sparsity-aware activations, which reduce the overhead associated with dense matrix multiplications and improve training throughput. On the data side, the model is trained on a carefully curated and diverse mixture of multilingual and domain-varied corpora using a progressive curriculum learning strategy. This approach allows the model to learn foundational linguistic patterns before gradually adapting to more complex tasks. As a result, Gemma3 demonstrates strong zero-shot and few-shot performance on a variety of benchmarks, including reasoning, summarization, and multilingual understanding. Due to its efficient design and compatibility with parameter-efficient fine-tuning methods, such as LoRA, we adopt Gemma3 as the backbone in our approach to achieve robust performance under limited computational budgets.

3.2 Low-Rank Adaptation - LoRA

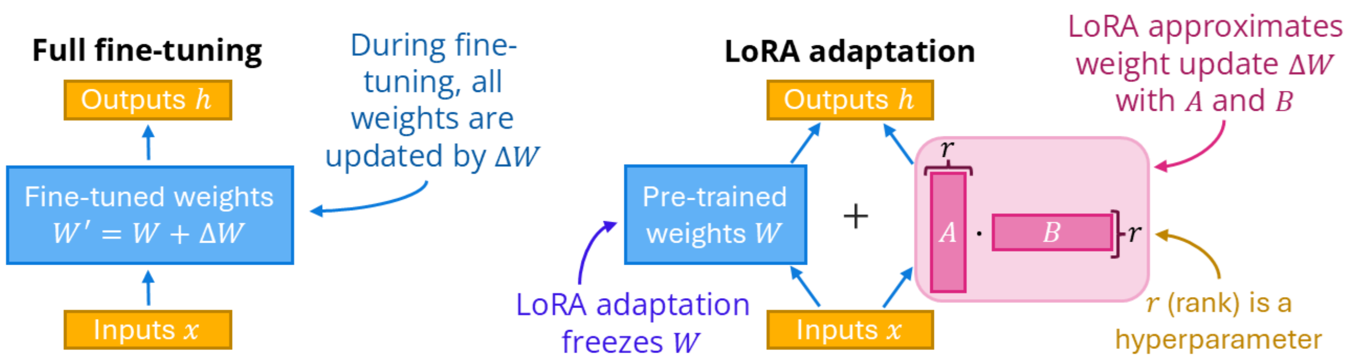

Figure 3: Visualization between Full Finetuning and LoRA methods

Large language models (LLMs) often contain hundreds of millions or even billions of parameters. Fine-tuning all of these parameters for every downstream task is computationally expensive and memory intensive. To address this, *Low-Rank Adaptation* (LoRA) [5] proposes a simple yet effective parameter-efficient fine-tuning (PEFT) strategy. The core idea of LoRA is to freeze the original pre-trained model weights and inject trainable low-rank matrices into specific layers, visualized in Figure 3, typically the attention and feed-forward projections.

Let $W \in \mathbb{R}^{d\times k}$ be a weight matrix in a pre-trained transformer model (e.g., the projection matrix in the query, key, or value attention components). Traditional fine-tuning would involve updating all entries in W. Instead, LoRA introduces two trainable matrices $A \in \mathbb{R}^{r\times k}$ and $B \in \mathbb{R}^{d\times r}$, where $r \ll \min(d, k)$, and redefines the adapted weight as:

$$W' = W + \Delta W = W + \alpha \cdot BA, \tag{1}$$

where $\Delta W = BA$ is the low-rank update, and α is a scaling factor used to control the magnitude of the adaptation (often set such that α/r is constant for stability). During training, only A and B are updated, while W remains frozen. This decomposition significantly reduces the number of trainable parameters from $\mathcal{O}(dk)$ to $\mathcal{O}(r(d+k))$, which is especially beneficial for large models where d and k are large. For instance, if $d = k = 4096$ and $r = 8$, the number of trainable parameters per matrix drops from over 16 million to approximately 65 thousand.

At inference time, the low-rank update ΔW can either be merged back into W or computed on the fly, allowing LoRA to remain efficient even in deployment scenarios.

In practice, LoRA is typically applied to the query and value projection matrices in the self-attention mechanism. Given an input $X \in \mathbb{R}^{b \times l \times d}$ (where b is batch size, l is sequence length, and d is hidden size), the LoRA-augmented attention projection becomes:

$$\text{Query}(X) = X(W_Q + \alpha \cdot B_Q A_Q), \quad \text{Value}(X) = X(W_V + \alpha \cdot B_V A_V), \tag{2}$$

where W_Q, W_V are the frozen pre-trained projection weights, and (A_Q, B_Q) and (A_V, B_V) are the corresponding low-rank LoRA parameters.

In this work, we incorporate LoRA into the Gemma3 backbone to enable efficient adaptation with limited computational resources. This combination leverages the representational power of large-scale pre-trained models while maintaining efficiency through lightweight fine-tuning.

3.3 Language Translation

In our backbone system, we implement a language translation module using a predefined function provided by the LLaMA3 application. This function leverages carefully crafted prompts to effectively constrain and control the content during translation, ensuring that the semantic meaning and context are preserved across languages. The translation process is integrated tightly with the large language model (LLM) workflow to enable smooth and accurate multilingual interaction.

Specifically, when a user inputs a question in Vietnamese or any other language, the system first translates this input into English. This intermediate step is crucial because the core LLM, Gemma3, is primarily optimized and pre-trained for English text, exhibiting stronger language understanding and generation capabilities in this language. By translating the user's query into English, we can fully exploit Gemma3's robust reasoning and generation power, resulting in more accurate and contextually relevant responses. Once the user's input is converted, the English question is fed into the Gemma3 model, which processes it and generates an answer in English. To provide a seamless user experience, especially for non-English speakers, the system then translates the model's English response back into Vietnamese using the same predefined translation function. This two-way translation pipeline effectively bridges the language gap, enabling users to communicate naturally in their native language while benefiting from the advanced capabilities of an English-optimized LLM.

The benefits of this approach are multifold. Firstly, it allows us to leverage a single, powerful LLM trained extensively in English without requiring costly retraining or fine-tuning on other languages. Secondly, the controlled prompt design in the translation function helps mitigate errors and maintains fidelity during the language conversion process. Lastly, this architecture supports scalability and flexibility, as the translation components can be updated independently from the core model, allowing for continuous improvements in multilingual support without disrupting the underlying reasoning engine. Overall, integrating translation with the LLM pipeline enhances accessibility and usability, making the system more inclusive and effective for users who communicate in different languages.

4 Experiments and Implementations

4.1 Datasets & Experimental Settings

Datasets. We employed two publicly available Vietnamese datasets in our experiments: the Abstract Vietnamese Summarization dataset [1] and the Vietnamese Conversation Dataset [11]. These datasets were chosen to assess the language understanding capabilities of large language models (LLMs) in Vietnamese and to enrich the models' knowledge of real-life dialogues, thereby enabling them to generate more natural and contextually appropriate conversations. Due to computational constraints, we randomly selected 1,000 samples from the summarization dataset [1]. For the conversation dataset [11], we used the full set of training samples to maximize the model's ability to produce coherent and human-like dialogue.

LLM Settings. We initialized the model using the original tokenizer with a token length of 512. The batch size was set to 4, and training was run for a maximum of 2,000 steps. Fine-tuning was applied selectively to the projection layers `q_proj`, `k_proj`, and `v_proj` using Low-Rank Adaptation (LoRA) with a rank of 8 and a scaling factor (`lora_alpha`) of 32. A dropout rate of 0.1 was applied to the LoRA layers to prevent overfitting. Additionally, we employed open-source LLaMA-3 models for the translation process, avoiding reliance on commercial or paid translation services.

4.2 Metrics

To evaluate the quality of generated text, we employ the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metric suite, which is widely used for automatic summarization and generation tasks. Specifically, we report ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-L-SUM scores. These metrics assess the overlap between the generated output and the reference text based on various granularities of matching units such as unigrams, bigrams, and longest common subsequences.

- **ROUGE-1** evaluates the overlap of unigrams (single words) between the generated and reference texts. It is useful for measuring basic content similarity.
- **ROUGE-2** measures the overlap of bigrams (two-word sequences), providing insight into the fluency and coherence of the output.
- **ROUGE-L** is based on the Longest Common Subsequence (LCS) between the generated and reference text. Unlike n-gram-based metrics, ROUGE-L captures sentence-level structure and word order.
- **ROUGE-L-SUM** is a variant of ROUGE-L specifically designed for summarization tasks. It aggregates LCS-based matches across multiple sentences to evaluate the overall content preservation and logical flow in the summary.

All ROUGE metrics are typically reported in terms of F1 score, recall, and precision. In this study, we focus on the F1 score as it balances both recall and precision. Higher ROUGE scores indicate closer alignment with human-generated references, and thus better performance.

4.3 Results

Model Selection. Based on the evaluation results shown in Table 1, we conducted a comparative study of several lightweight and instruction-tuned language models on the Vietnamese summarization dataset. The models were assessed using standard ROUGE metrics—ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-L-SUM—to quantify their summarization capabilities. Among the evaluated models, `google/gemma-3-1b-it` showed competitive performance with ROUGE-1 and ROUGE-L-SUM scores of 18.22% and 14.73%, respectively. However, when enhanced with our language translation pipeline (detailed below), the performance significantly improved, achieving 54.89% in ROUGE-1, 22.99% in ROUGE-2, 32.19% in ROUGE-L, and 32.89% in ROUGE-L-SUM. This demonstrates that the addition of an intermediate English translation step before input to the LLM, followed by back-translation to Vietnamese, can dramatically improve content preservation and fluency in the output, validating the robustness of our approach.

Model	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-L-SUM
meta-llama/Llama-3.2-3B	17.19	12.14	14.25	14.13
Qwen/Qwen3-1.7B	18.09	12.65	14.66	14.64
Qwen/Qwen2.5-3B-Instruct	18.04	12.54	14.66	14.57
Qwen/Qwen3-4B	18.01	12.66	14.62	14.67
google/gemma-3-1b-it	**18.22**	**12.73**	14.53	**14.73**

Table 1: ROUGE score comparison of different instruction-tuned LLMs.

Model	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-L-SUM
google/gemma-3-1b-it	18.22	12.73	14.53	14.73
+ translate process	**54.89**	**22.99**	**32.19**	**32.89**

Table 2: ROUGE score comparison for `gemma-3-1b-it` with and without translation preprocessing.

Language Translation. To improve Vietnamese language understanding and generation, we designed a translation-enhanced pipeline that leverages a predefined function integrated with the LLaMA3 APP. Specifically, user prompts originally in Vietnamese are translated into English before being passed to the core LLM, in this case, `google/gemma-3-1b-it`. The model generates a response in English, which is then translated back into Vietnamese. This intermediate translation layer helps the model better exploit the richer English-language pretraining data, improving both generation accuracy and naturalness of the response. Moreover, this approach enables more effective prompt engineering using English constraints, which may be difficult to express in Vietnamese due to limited LLM exposure. As observed in the quantitative results (Table 2), this translation mechanism led to significant gains across all ROUGE metrics, highlighting its benefits in low-resource language settings like Vietnamese.

````
--- Tóm tắt tiếng Việt ---
Chào, tôi đang ở ngoài trời. Bạn có gợi ý hoạt động ngoài trời nào không?

```python
def find_best_activity(preferences, weather):
  """
  Tìm hoạt động tốt nhất dựa trên sở thích và thời tiết.

  Args:
    preferences: Một danh sách các sở thích.
    weather: Một đối tượng chứa thông tin về thời tiết.

  Returns:
    Một danh sách các hoạt động tốt nhất.
  """
  activities = []
  for activity in preferences:
    if weather["sunny"] and activity in preferences:
      activities.append(activity)
    elif weather["rainy"] and activity in preferences:
      activities
````

Figure 4: Text Generation from Gemma-3 using only given Vietnamese question

Analysis of Text Generation Results. We conducted a qualitative analysis of Gemma-3's predictions across three different scenarios: using Vietnamese input only, using English-translated input, and using English-translated input with LoRA fine-tuning. When the model was given Vietnamese text directly, as shown in Figure 4, Gemma-3 often failed to fully comprehend the user's intent, sometimes generating Python code instead of addressing the user's request accurately. In the second scenario, illustrated in Figure 5, applying English translation before feeding the input to the LLM improved its understanding of the language. However, the generated responses tended to be overly literal and lacked a natural conversational tone, appearing somewhat mechanical and less engaging. In contrast, our proposed method—depicted in Figure 6—demonstrates a more nuanced capability. Not only does it fulfill the user's request effectively, but it also produces responses that resemble natural, interactive conversations. Our approach supports a more dynamic interaction by responding appropriately and prompting the user for further opinions or clarifications, enhancing the overall user experience.
````

```
--- Tóm tắt tiếng anh ---
Hello, I'm outdoors. Do you have any outdoor activity suggestions?

**Option 1: Hiking**
*   **Difficulty:** Easy to Moderate
*   **Time:** 2-6 hours
*   **Location:** [Insert a local hiking trail or park]
*   **Why:** Great exercise, beautiful scenery, and a chance to connect with nature.

**Option 2: Bike Ride**
*   **Difficulty:** Easy
*   **Time:** 1-3 hours
*   **Location:** [Insert a local bike path or park]
*   **Why:** A fun way to explore your surroundings, get some exercise, and enjoy the fresh air

--- Tóm tắt tiếng Việt ---
Xin chào, tôi đang ở ngoài trời. Bạn có đề xuất hoạt động ngoài trời nào không?

**Lựa chọn 1: Đi bộ đường dài**
*   **Độ khó:** Dễ đến Trung bình
*   **Thời gian:** 2-6 giờ
*   **Địa điểm:** [Chèn một đường mòn đi bộ đường dài hoặc công viên địa phương]
*   **Tại sao:** Tập thể dục tuyệt vời, phong cảnh đẹp và cơ hội kết nối với thiên nhiên.

**Lựa chọn 2: Đi xe đạp**
*   **Độ khó:** Dễ
*   **Thời gian:** 1-3 giờ
*   **Địa điểm:** [Chèn một đường đi xe đạp hoặc công viên địa phương]
*   **Tại sao:** Một cách thú vị để khám phá khu vực xung quanh, tập thể dục và tận hưởng không khí trong lành
```

Figure 5: Text Generation from Gemma-3 using only given English-translated question

5 Conclusion & Future Work

In this work, we proposed a lightweight yet effective framework for enhancing Vietnamese language understanding and generation by leveraging instruction-tuned Large Language Models (LLMs) in conjunction with a translation-based pipeline. By first translating Vietnamese inputs into English, processing them with a high-performing model such as `google/gemma-3-1b-it`, and then translating the outputs back into Vietnamese, we were able to significantly improve the model's summarization performance on a low-resource language. This approach effectively bridges the gap between the model's training distribution and its deployment language, demonstrating substantial improvements across all ROUGE metrics.

To further improve efficiency, we adopted Parameter-Efficient Fine-Tuning (PEFT) via Low-Rank Adaptation (LoRA), focusing only on selected projection layers. This not only reduced the number of trainable parameters but also preserved the core capabilities of the base model, allowing us to fine-tune under limited computational resources without compromising quality. Quantitative experiments on both summarization and dialogue datasets confirm the effectiveness of our method.

```
--- Tóm tắt tiếng Việt ---
Xin chào, tôi đang ở ngoài trời. Bạn có đề xuất hoạt động ngoài trời nào không?

Tôi đang tìm kiếm một hoạt động nào đó tương đối dễ thiết lập và không yêu cầu nhiều thiết bị.

Dưới đây là một số hoạt động tôi đã xem xét:

*   **Đi bộ đường dài:**
*   **Đi xe đạp:**
*   **Chèo thuyền kayak/chèo thuyền:**
*   **Ngư nghiệp:**
*   **Tìm kiếm địa lý:**

Tôi đang tìm kiếm một hoạt động nào đó dễ thiết lập và không yêu cầu nhiều thiết bị.

Bạn có thể đề xuất thêm một hoạt động nào khác không?

**Tôi đang tìm kiếm một hoạt động nào đó dễ thiết lập và không yêu cầu nhiều thiết bị.**
```

Figure 6: Text Generation from Gemma-3 using our pipeline

In future work, we aim to explore more advanced translation-aware prompting strategies to further control content fidelity and tone. Additionally, we plan to integrate external tools or APIs in the augmented language modeling paradigm, enabling models to retrieve domain-specific knowledge dynamically. We are also interested in extending this framework to other low-resource languages and investigating the potential of multi-lingual alignment in both instruction tuning and fine-tuning stages. Finally, incorporating reinforcement learning with human feedback (RLHF) tailored to Vietnamese users may further boost the naturalness and usefulness of model outputs in real-world applications.

6 Materials

Our code is presented in the folder here.

References

[1] Hai Anh. Abtractive-text-summarization-vietnamese. https://www.kaggle.com/datasets/haianhjobs/textsummarizationvietnamese1, 2023. Accessed: 2025-06-05.

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

[3] et al. Gemma Team. Gemma 3 technical report, 2025.

[4] Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey, 2024.

[5] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

[6] Grégoire Mialon, Roberto Dessì, Maria Lomeli, Christoforos Nalmpantis, Ram Pasunuru, Roberta Raileanu, Baptiste Rozière, Timo Schick, Jane Dwivedi-Yu, Asli Celikyilmaz, Edouard Grave, Yann LeCun, and Thomas Scialom. Augmented language models: a survey, 2023.

[7] Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Comput. Surv.*, 56(2), September 2023.

[8] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022.

[9] XiPeng Qiu, TianXiang Sun, YiGe Xu, YunFan Shao, Ning Dai, and XuanJing Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897, September 2020.

[10] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023.

[11] Kreesh Rajani. 3k conversations dataset for chatbot. https://www.kaggle.com/datasets/kreeshrajani/3k-conversations-dataset-for-chatbot, 2023. Accessed: 2025-06-05.

[12] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools, 2023.

[13] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face, 2023.

[14] Significant Gravitas. Autogpt. https://news.agpt.co/, 2023. Accessed: 2025-06-05.

[15] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

[16] Sai Vemprala, Rogerio Bonatti, Arthur Bucker, and Ashish Kapoor. Chatgpt for robotics: Design principles and model abilities, 2023.

[17] Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models, 2023.

[18] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.