Evaluating LLMs on Vietnamese Duration Question Answering

Nhat-Truong Dinh[1,2], Thanh-Trung Ngo[1,2], Quoc-Bao Trinh[1,2], Duc-Vu Nguyen[1,2]
[1]University of Information Technology, Ho Chi Minh City, Vietnam
[2]Vietnam National University, Ho Chi Minh City, Vietnam
{22521575, 22521560, 22520125}@gm.uit.edu.vn **Correspondence:** vund@uit.edu.vn

Abstract

Time cognition is crucial for both humans and machines, as understanding temporal information enables event interpretation and dynamic reasoning. This capability is essential for tasks such as temporal question answering, legal retrieval, and video captioning. To advance research in this area, Chu et al. (2023) introduced TimeBench, a benchmark covering nine temporal QA categories and evaluated with models such as GPT-4, Llama 2, and Mistral. However, Vietnamese temporal reasoning remains underexplored. In this study, we evaluate Qwen3 on the TimeBench Duration task in the VLSP 2025 Challenge on Temporal QA, using GPT-based generation to preserve original semantics. Our model achieved an F-score of 0.81 on the public test and 0.80 on the private test, ranking among the top solutions.

1 Introduction

Temporal question answering (Jia et al., 2018) is a fundamental task for large language models, enabling them to comprehend information within its temporal context and accurately extract time-based details to provide appropriate responses. Leveraging language models for temporal question answering can benefit related applications, such as document retrieval, by allowing systems to filter outdated laws or identify versions of legal documents that are not suitable for a given query, which semantic search alone may not effectively address. In the context of video captioning, temporal reasoning allows models to interpret continuous events and respond to user queries with greater efficiency and accuracy.

In the previous study, large language models (LLMs) were evaluated on the TimeBench dataset (Chu et al., 2023). The results indicated that LLMs were generally unable to handle most temporal questions efficiently, with scores ranging from 0.50 to 0.98. Notably, the DurationQA question type showed one of the lowest performance scores at 0.62, highlighting a significant gap between the capabilities of large language models and human performance.

Listing 1 Each training sample is represented as a dictionary with four keys: "ctx" (context), "opts" (duration options), "q" (question), and "label" (correct answers). In the test set, the "label" field is omitted.

```
sample = {
    "ctx": (
        "Trong một lớp học, "
        "các học sinh đang học "
        "về các chủ đề khác nhau. "
        "Một số em rất chăm chỉ "
        "và thường xuyên hoàn thành ..."
    ),
    "opts": [
        "1 tuần", "5 năm", "10 ngày", "10 giây"
    ],
    "q": (
        "Mất bao lâu để hoàn thành "
        "tất cả bài tập về nhà của lớp học?"
    ),
    "label": [
        "yes", "no", "yes", "no"
    ] # excluded in test set
}
```

Research in this domain remains limited, particularly for the Vietnamese language, where prior studies and available data are scarce. This study aims to address this gap by evaluating language model performance on DurationQA questions specifically in Vietnamese.

The Duration Question Answering (DurationQA) task is designed to evaluate a system's ability to answer questions related to the duration of an event or action based on a given context. Specifically, the system must extract explicit temporal information from the text or infer implicit durations by leveraging commonsense knowledge, thereby determining the plausibility of each answer option. The input consists of a context containing

temporal information, a duration-related question, and a set of candidate answers, and the output is a list of "yes" or "no" labels for each option, indicating whether it correctly represents the duration of the event or action. Formally, let the input be defined as

$$x = \{c, q, O\},$$

where c is the context, q is the duration-related question, and $O = \{o_1, o_2, \ldots, o_m\}$ is the set of answer options. The system learns a labeling function

$$f : (c, q, O) \rightarrow Y,$$

where $Y = \{y_1, y_2, \ldots, y_m\}$, and each $y_i \in$ {"yes", "no"}.

2 Related Work

TimeBench (Chu et al., 2023) is a comprehensive, hierarchical benchmark introduced to evaluate large language models' temporal reasoning across a wide spectrum of phenomena. The authors organize temporal reasoning into three levels: symbolic temporal reasoning, temporal commonsense, and event temporal reasoning. They assemble 10 datasets with 16 subtasks spanning four task formats (reading comprehension, NLI, constrained generation, and multi-select) to evaluate different temporal capabilities. The paper reports extensive zero-shot and few-shot experiments, including chain-of-thought prompting, on many contemporary models such as GPT-4 (Achiam et al., 2023), LLaMA3 (Dubey et al., 2024), and Mistral (Chaplot, 2023). The results show that while GPT-4 achieves the strongest performance, a substantial gap remains between state-of-the-art models and humans, and chain-of-thought prompting does not consistently improve temporal performance. TimeBench specifically includes symbolic tasks such as date arithmetic and TimeX inference; commonsense tasks covering duration, frequency, and typical time (e.g., MC-TACO, DurationQA, TimeDial); and event-level tasks requiring event–time and event–event reasoning (e.g., TimeQA, MenatQA, TempReason). Together, these tasks reveal model weaknesses in abstract time understanding, temporal relation modeling, and implicit temporal inference.

Tan et al. (2023) introduce TempReason, a benchmark designed to rigorously evaluate the temporal reasoning capabilities of large language models (LLMs). This benchmark assesses performance across three hierarchical levels: time–time, time–event, and event–event relations. It utilizes both synthetically generated and Wikidata-derived questions spanning a wide range of temporal intervals, from centuries to months. In contrast to previous datasets with limited temporal scope or reasoning complexity, TempReason includes closed-book, open-book, and reasoning-based question-answering settings. This design enables the evaluation of both memorization and inferential capabilities. Additionally, the authors present TempT5, a model trained with temporal span extraction pre-training and time-sensitive reinforcement learning. TempT5 demonstrates consistent improvements over strong baselines, particularly in reasoning-based question-answering scenarios, and reduces temporal bias toward recent time periods.

Virgo et al. (2022b) propose an effective method to enhance event duration question answering by explicitly bridging the gap between duration classification and question answering tasks. They automatically recast the event-duration annotations from UDS-T (Virgo et al., 2022a) into a question answering format similar to McTACO (Zhou et al., 2019), resulting in the creation of the UDST-DurationQA dataset. To effectively leverage this dataset, a two-stage fine-tuning strategy is employed: the model is first fine-tuned on UDST-DurationQA and subsequently on McTACO-duration, which (Zhou et al., 2019) demonstrates provides a substantial improvement of approximately 13% in Exact Match and 5% in F_1-score compared to a RoBERTa baseline. Their results underscore the importance of addressing task-format discrepancies between intermediary and target tasks when transferring duration knowledge, a key factor in improving duration reasoning across QA datasets.

The introduction of Qwen3 (Yang et al., 2025) represents a significant advancement in the development of large language models. This family of open-weight models encompasses a broad range of parameter sizes, from 0.6B to 235B, and incorporates both dense and Mixture-of-Experts (MoE) (Zhou et al., 2022) architectural designs. Notably, the flagship MoE variant activates only a subset of its total parameters during inference, effectively balancing computational efficiency with model performance. Qwen3 (Yang et al., 2025) models have demonstrated state-of-the-art results across multiple domains, highlighting the potential of scalable and efficient architectures for a wide array of downstream tasks.

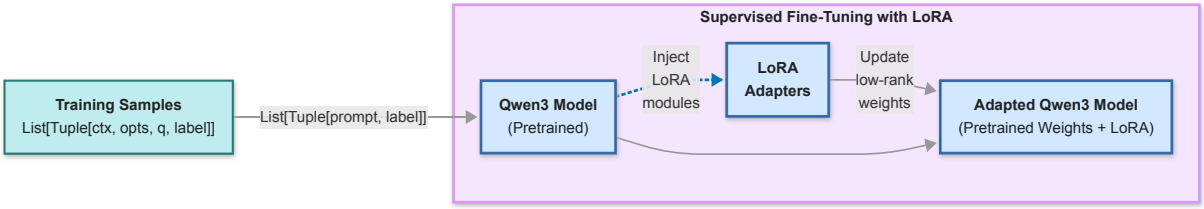

Figure 1: Overview of the training pipeline. Each sample (see Listing 1) is formatted into an experimentally designed prompt and used for supervised fine-tuning of the model.

3 Method

Overview, the training pipeline converts each sample into a prompt-label pair and fine-tunes a pretrained Qwen3 model via LoRA, updating only the adapter weights (Figure 1).

3.1 Prompt Design

To adapt large language models for the Duration Question Answering task, we designed a specialized prompt that explicitly guides the model to perform temporal and duration-related reasoning. The prompt frames the model as a temporal expert, instructing it to carefully read the context and question, analyze the candidate answers, and classify each option as correct or incorrect. The function is shown in Listing 2.

Listing 2 The function build_prompt constructs a duration reasoning prompt by combining four parts: a system instruction and three inputs ctx (context), opts (duration options), and q (question). These inputs correspond to the data sample shown in Listing 1.

```
def build_prompt(ctx, opts, q):
    """
    Build a duration QA prompt.
    """
    prompt = (
        f"Bạn là một chuyên gia về thời gian "
        f"và thời lượng. Nhiệm vụ của bạn "
        f"là đọc kỹ ngữ cảnh và câu hỏi, "
        f"sau đó phân tích để xác định "
        f"câu trả lời phù hợp "
        f"trong các lựa chọn được đưa ra. "
        f"Mỗi lựa chọn có thể đúng hoặc sai, "
        f"hãy đánh giá chính xác.\n\n"
        f"{ctx}\n\n"
        f"{opts}\n\n"
        f"{q}"
    )
    return prompt
```

This prompt is concatenated with the context, the duration-related question, and the candidate answer options to form the final input to the model. By explicitly framing the task in this way, we encourage the model to perform structured temporal reasoning rather than relying solely on superficial text matching.

3.2 Supervised Fine-tuning

The initial phase of training involves supervised fine-tuning to adapt the foundational language model Qwen to the specific requirements of the Vietnamese DurationQA dataset. At this stage, the model has no prior knowledge of the DurationQA task or its output format. Through supervised fine-tuning, it learns to accurately interpret prompts, process long-form documents and tabular data, and generate precise responses. Due to the substantial computational resources and time required for full-parameter fine-tuning, we employ LoRA-based fine-tuning (Hu et al., 2022) with hyperparameters set to rank $R = 64$ and LoRA alpha = 64. Furthermore, we use a quantized version of the Qwen3 model, reducing the parameter count from 32 billion to 24 billion to improve efficiency without sacrificing performance.

4 Experiment

4.1 Data Statistics

The dataset[1] comprises 1,490 samples, each linked to a document providing temporal context. Context lengths range from 7 to 82 tokens, with an average of 28 tokens and a median of 25 tokens. Fourteen distinct temporal units appear in the answer options, and their frequencies are summarized in Table 1. Shorter and more frequent units (hours, minutes, months, weeks) dominate the dataset, while broader time spans (decade, million, millennium) occur rarely. This indicates that the dataset primarily evaluates fine-grained temporal reasoning while still including occasional instances involving broader time spans.

[1] VLSP 2025 Challenge on Temporal QA: https://vlsp.org.vn/vlsp2025/eval/tempqa

Temporal Unit	Frequency
Week	684
Year	515
Day	602
Month	782
Hour	970
Minute	873
Second	265
Decade	1
Thập kỷ	108
Century	28
Million	2
Symbol "."	2
Million years	3
Millennium	1

Table 1: Distribution of temporal units within the dataset.

We randomly split the dataset into training (80%), development (10%), and test (10%) sets, resulting in 1,192 training samples, 149 development samples, and 149 test samples. The distribution of temporal units is consistent across all three splits, ensuring that data imbalance does not affect model training or evaluation.

4.2 Experimental Setup

We leverage the TRL[2] library to perform supervised fine-tuning (SFT) on the Qwen3-32B 4-bit model quantized using BNB[3]. We use SFTTrainer with a per-device batch size of 16 and gradient accumulation over 4 steps (for an effective batch size of 64), training over 10 epochs. The learning rate is set to 2×10^{-4} with a linear decay schedule and 5 warmup steps. We optimize using adamw_8bit (to reduce memory usage) with a weight decay of 0.01. A fixed random seed (3407) ensures reproducibility. Logging occurs each step, evaluation and checkpointing each epoch, with the best model chosen by development loss.

Fine-tuning was conducted on a single NVIDIA RTX 4090 GPU with 24 GB of VRAM, requiring approximately 1–2 hours to complete all 10 training epochs. For inference and leaderboard submission, a single NVIDIA A100-SMX4 GPU with 40 GB of VRAM was used, enabling efficient batch processing and stable evaluation throughput.

[2]Train Transformer Language Models with Reinforcement Learning: https://github.com/huggingface/trl

[3]Qwen3-32B 4-bit model, quantized using BNB: https://huggingface.co/unsloth/Qwen3-32B-bnb-4bit

4.3 Evaluation Metric

System performance is evaluated using standard metrics, including Exact Match, Precision, Recall, and F_1-score (Yacouby and Axman, 2020). Exact Match is applied specifically to Sub-Task 2 and measures whether the predicted label sequence exactly matches the ground-truth sequence, defined as

$$\text{EM} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\big(\hat{Y}^{(i)} = Y^{(i)}\big).$$

Precision is computed as the ratio of correctly predicted "yes" answers to the total number of "yes" predictions made by the system:

$$\text{P} = \frac{\text{TP}}{\text{TP} + \text{FP}},$$

while Recall is defined as the ratio of correctly predicted "yes" answers to the total number of actual "yes" answers in the ground truth:

$$\text{R} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

The F_1-score, calculated as the harmonic mean of Precision and Recall, is given by

$$\text{F}_1 = \frac{2 \cdot \text{P} \cdot \text{R}}{\text{P} + \text{R}}.$$

Evaluation is conducted separately for each subtask, and the final report provides both individual results and aggregate performance across all tasks, considering both "yes" and "no" labels.

4.4 Main Result

Test Set	EM	Precision	Recall	F_1
Public Test	0.48	0.75	0.88	0.81
Private Test	0.43	0.73	0.89	0.80

Table 2: Performance of our model on the public and private test sets of DurationQA.

Table 2 summarizes the model performance on both the public and private test sets of DurationQA. On the public test set, our model achieved strong overall performance with an F_1-score of 0.81, a precision of 0.75, and a recall of 0.88. However, the Exact Match (EM) score was lower at 0.48, indicating a gap between token-level classification and strict answer matching. On the private test set, we observed slightly lower but still competitive results, with an F_1-score of 0.80 and an EM of 0.43.

4.5 Error Analysis

To identify the limitations of our system, we manually analyzed 50 mispredicted samples. Of these, 31 involved the model incorrectly predicting "yes" for durations exceeding the maximum possible time indicated in the context, demonstrating a tendency to overestimate and accept implausibly long durations. In contrast, 29 samples involved predictions of "yes" for durations shorter than the minimum required to complete an event, revealing a weakness in reasoning about necessary temporal constraints and real-world feasibility. These results indicate a systematic challenge in grounding temporal reasoning within realistic boundaries. Although the model can extract explicit temporal expressions, it often fails to align them with contextual constraints, resulting in both overestimation and underestimation errors. Addressing this limitation may require integrating external temporal knowledge bases or augmenting the training data with counterfactual duration reasoning examples.

5 Discussion

Our pipeline reveals that a considerable performance gap persists in this task, suggesting significant opportunities for further enhancement. However, the current reward mechanism remains suboptimal, as it does not address all scenarios present in the dataset, particularly cases involving unit mismatches, which consequently constrain model performance. Moreover, implementing a robust mechanism to filter irrelevant factual information from the input is essential. Greater investment of time and computational resources is warranted for training the model with the GRPO approach, given its lengthy convergence period. It is also advisable to explore a range of augmentation techniques, such as introducing greater variability in numerical program transformations. Error analysis indicates that some calculation functions are missing from the training data, which may further impede model accuracy. Finally, alternative evaluation strategies should be pursued, as many mathematically valid reasoning programs may employ different calculation functions yet produce correct results, a nuance not fully captured by the current evaluation system.

6 Conclusion

In conclusion, our study highlights both the progress and the remaining challenges in temporal question answering for Vietnamese, particularly on the DurationQA task within the TimeBench benchmark. By tailoring prompts and applying supervised fine-tuning to the Qwen3 model, we achieved an F_1-score of 0.81 on the public test set and 0.80 on the private test set, demonstrating competitive performance against existing large language models. However, our analysis reveals persistent difficulties in aligning model predictions with contextual temporal constraints, leading to systematic overestimation and underestimation errors. These findings underscore the need for further research, including the integration of external temporal knowledge, the development of more sophisticated reward and evaluation mechanisms, and the adoption of advanced data augmentation strategies. Despite notable advancements, a significant gap remains between current model capabilities and the nuanced temporal reasoning required for real-world applications, indicating ample opportunity for future work in this domain.

Acknowledgement

This research was supported by The VNUHCM-University of Information Technology's Scientific Research Support Fund. We thank the anonymous reviewers for their time and helpful suggestions that improved the quality of the paper.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Devendra Singh Chaplot. 2023. Albert q. jiang, alexandre sablayrolles, arthur mensch, chris bamford, devendra singh chaplot, diego de las casas, florian bressand, gianna lengyel, guillaume lample, lucile saulnier, lélio renard lavaud, marie-anne lachaux, pierre stock, teven le scao, thibaut lavril, thomas wang, timothée lacroix, william el sayed. *arXiv preprint arXiv:2310.06825*, 3.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Haotian Wang, Ming Liu, and Bing Qin. 2023. Timebench: A comprehensive evaluation of temporal reasoning abilities in large language models. *arXiv preprint arXiv:2311.17667*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018. Tempquestions: A benchmark for temporal question answering. In *Companion Proceedings of the The Web Conference 2018*, pages 1057–1062.

Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023. Towards benchmarking and improving the temporal reasoning capability of large language models. In *arXiv preprint arXiv:2306.08952*.

Felix Virgo, Fei Cheng, and Sadao Kurohashi. 2022a. Improving event duration question answering by leveraging existing temporal information extraction data. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4451–4457.

Felix Giovanni Virgo, Fei Cheng, and Sadao Kurohashi. 2022b. Improving event duration question answering by leveraging existing temporal information extraction data. In *Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022)*, pages 4451–4457.

Reda Yacouby and Dustin Axman. 2020. Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models. In *Proceedings of the first workshop on evaluation and comparison of NLP systems*, pages 79–91.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. 2019. "going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3363–3369, Hong Kong, China. Association for Computational Linguistics.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, and 1 others. 2022. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, 35:7103–7114.