A Comparative Study on the Development of a Thai Legal QA Framework Using Large Language Models and Mixed Legal Datasets

Supachoke Hanwiboonwat[1], Chaichana Thavornthaveekul[2], Prachya Boonkwan[3], Apivadee Piyatumrong[4], and Peerapon Vateekul[1]

[1] Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Pathum Wan, Bangkok 10330, Thailand
[2] Data Wow Co., Ltd.
[3] School of ICT, Sirindhorn International Institute of Technology (SIIT), Thammasat University, Khlong Luang, Pathum Thani 12120, Thailand
[4] Big Data Institute (Public Organization).

Abstract. In the present day, large language models (LLMs) such as GPT-4o play a significant role in answering legal questions in the Thai language. However, creating a system for answering legal questions for the general public is a highly challenging and complex task. This is due to the fact that Thai legal code documents use complex language and are often lengthy. This research focuses on developing a framework for legal question answering (QA) targeted at the general public, with the aim of establishing best practices for creating effective legal QA systems. To enhance the system's performance, we constructed our own dataset and integrated data from a variety of sources. Intensive experiments were conducted to identify the most suitable LLM for legal applications in the Thai context. We proposed a method to improve QA performance in the legal domain with LLM by fine-tuning multiple task datasets. The entire process is thoroughly detailed, covering aspects such as fine-tuning and the use of retrieval-augmented generation (RAG), including techniques like keyword search and contextual search, as well as reranking processes. Additionally, this study compares various prompt formats to find the most effective one for answering legal questions for the general public. The results from our model are comparable to larger models with more parameters, such as GPT-4o, in legal examination tasks and perform better in answering legal questions tasks.

Keywords: Large Language Model · Thai Legal Documents · Question Answering.

1 Introduction

Artificial intelligence (AI) plays a significant role in the legal field [2,27,29], and there are applications that utilize AI in various legal aspects, such as legal question answering (QA) [30,6,34], legal judgment prediction [5], case prediction [7],

and contract review [21]. In terms of legal QA, the launch of ChatGPT [24] has played a crucial role in answering legal questions [13,15].

However, developing a legal QA system for the general public, who lack legal knowledge, poses several challenges. due to legal terminologies and lengthy idiomatic expressions [14]. These issues can result in large language models (LLMs) not performing well in answering legal questions for the general public. This aforementioned work addresses this problem by creating a legal story using LLMs to help the general public learn about the law in simple language. This contrasts with our work, which focuses on developing a framework for answering legal questions for the general public. Our work does not compare to this aforementioned study because we focus on measuring the accuracy of legal QA, while the other study evaluates the readability of the legal stories as output.

To address these issues, we have enhanced the performance of the LLM to effectively answer legal questions for the general public with the highest efficiency. We developed a retrieval-augmented generation (RAG) system that is most suitable for the Thai legal domain, enabling the LLM to identify which specific legal provisions should be used to answer questions, and adjusted the prompt to ensure the LLM can provide the best responses to legal questions for the general public.

This research aims to design best practices for legal QA, focusing on three main components: selecting the most effective Large Language Model for answering Thai legal questions, developing a highly accurate RAG system for extracting relevant legal codes from databases, and comparing various prompt formats to determine which most efficiently enhance the model's ability to answer legal questions for the general public. Additionally, we present methods to improve the performance of the LLM by utilizing datasets of various formats, which prove to be more effective than using a single-format dataset. Our research addresses the following research questions :

1. Which LLMs model is the best LLM for answering Thai legal questions for the general public?
2. In RAG system for retrieving Thai legal documents, which method is the most effective?
3. In improving the performance of LLM, which good method makes the LLMs most efficient?
4. What type of prompt is the most appropriate for answering Thai legal questions for the general public?

We compared our model with large LLM like GPT-4o in terms of providing appropriate legal answers and the accuracy of solving legal examination questions. Our results show that legal QA performance superior to larger models like GPT-4o and comparable efficiency in performing in bar exams with GPT-4o. In summary, our work aims to develop a best practice legal QA framework that efficiently and accurately answers legal questions in Thai.

2 Related work

2.1 Thai-Enabled Large Language Models

There are several attempts to enable Thai in existing LLMs via continual pretraining and finetuning. First, Typhoon 2 [26] is a collection of LLMs developed by SCB10X, featuring two models: typhoon2-qwen2.5-7b-instruct, based on the Qwen-v2.5-7B [33] with 7 billion parameters, and llama3.1-typhoon2-8b-instruct, based on llama-3.1-8B [11] with 8 billion parameters. Second, SeaLLMs-v3-7B-Chat [39], also known as Seallm3, is the latest in the SEALLM series, trained on Southeast Asian languages including English, Chinese, Indonesian, Vietnamese, Thai, and others, with QWEN2 [32] as its base. OpenThaiGPT 1.5-7B-Instruct, OpenThaiLLM-Prebuilt-7B, and PathummaLLM-Text-1.0.0, all developed by NECTEC, are notable Thai language models. Third, OpenThaiGPT 1.5-7B-Instruct [37] is a 7-billion-parameter Thai model, built on Qwen v2.5 and refined using over 2 million Thai instructional pairs. Fourth, OpenThaiLLM-Prebuilt-7B [22] is a multilingual model, also with 7 billion parameters, continuing from Qwen2.5-7B and optimized for applications like Retrieval-Augmented Generation and reasoning. Fifth, PathummaLLM-Text-1.0.0 [23], with 7 billion parameters, is fine-tuned from OpenThaiLLM-Prebuilt for practical use. Sixth, Sailor2-8B-Chat [9], based on Qwen-2.5 7B, supports Southeast Asian languages, including Thai.

2.2 Text Embedding and Reranking Models

Text embedding models convert input text into a vector representation for semantic similarity. With contextualized information, these vectors enable semantic search, where documents are retrieved by meanings rather than keywords. The paraphrase-multilingual-mpnet-base-v2 [28], also from 2019, by the sentence-transformers team, converts text into dense vectors for tasks like clustering and semantic search, benefiting from the strengths of both BERT and XLNet for comprehensive information capture. The Multilingual E5 [31], is designed for text embedding and provides a more accurate alternative to the older paraphrase-multilingual-mpnet-base-v2 model by utilizing a diverse dataset called CCPairs, which includes community QA content, CommonCrawl, and scientific papers. This leads to reliable embeddings across multiple languages. In contrast, the bge-reranker-v2-m3 [4], a reranking model, supports Thai by generating a similarity score for a given question and document, rather than embeddings. This relevance score is converted into a float value between 0 and 1 using a sigmoid function, enabling precise evaluation of document relevance. The main function of the reranking model is to rank the importance of documents retrieved from keyword search and contextual search.

2.3 Relevant Legal Works

Our work investigates several LLM-based approaches for legal text processing, focusing on adapting them to Thai legal documents. "Adapt-Retrieve-Revise"[30]

fine-tunes a Chinese legal LLM and uses GPT-4 for answer revision, but we do not compare it due to differences in document structures. "Legal Reasoning Prompts"[35] utilizes the IRAC framework to structure legal answers; in our work, we evaluate various IRAC-style prompts to determine the best fit for Thai legal contexts. "KELLER"[7] is a reranking model using expert summaries to improve retrieval, and we adopt a similar approach with positive and negative document identification in our own reranking model. SaulLM-7B[6] is a large language model designed specifically for the legal domain, but it is excluded from our comparisons as it lacks Thai training data and performs poorly with Thai legal text. Overall, our methodology adapts and customizes these techniques specifically for Thai legal tasks.

3 Methodlogy

3.1 Datasets

Large Language Model Enhancement Dataset To enhance the performance of a LLM specifically for question-answering tasks, we utilized three datasets: the Tamtanai dataset, the WangchanX-Legal-ThaiCCL-RAG dataset [1], and the Thai attorney exam dataset [16,17,18]. The Tamtanai dataset, used as the primary dataset for fine-tuning a LLM, comprises a comprehensive collection of official legal documents from a specialized Thai legal documentation website. To ensure the relevance and significance of the data, 31 pivotal documents frequently referenced in legal consultations were meticulously selected, guided by consultations with legal experts to address crucial areas relevant to the Thai legal system. To further enrich the dataset, GPT-4 was employed to generate additional training and testing data, with questions and answers designed to enable the model to effectively respond to legal inquiries from the public, thereby enhancing its ability to provide accessible legal information and assistance. The dataset consists of 4,534 entries, which were randomly divided into 3,296 for training, 825 for validation, and 413 for testing. The second dataset, WangchanX-Legal-ThaiCCL-RAG, is a Thai legal QA dataset selected to enhance LLM performance with additional legal data, comprising 409 entries distributed as 327 for training and 82 for validation, with no testing entries. Lastly, the Thai attorney exam dataset, containing multiple-choice questions from past attorney exams over 27 years, was processed to remove duplicates, ensuring no overlap between training and testing datasets. There are a total of 409 entries, consisting of 327 training entries, 82 validation entries, and 80 test entries. In this work, we divide both the WangchanX-Legal-ThaiCCL-RAG and the Thai attorney exam dataset in half, using 205 entries from each, for a total of 410 entries, to evaluate the improvement in the LLM's performance. When combined, the total size is approximately equal to that of the Thai attorney exam dataset.

Retrieval-Augmented Generation Enhancement Dataset Before entering the RAG process, queries are first classified to determine whether they belong to the legal domain. To fine-tune the query classification model, we utilized

two datasets: the Tamtanai dataset, representing legal domain queries (positive label), and the han-instruct-dataset-v2.0 dataset [25], representing non-legal domain queries (negative label). The han-instruct-dataset-v2.0 dataset contains 3,200 entries, divided into 2,324 for training, 556 for validation, and 320 for testing. We employed GPT-4o to label the han-instruct-dataset-v2.0 dataset, identifying which queries fall outside the legal domain and which ones are within the legal domain. Afterwards, a sample of the labels generated by GPT-4o will be randomly selected for human evaluation to determine whether the labels are appropriate. For fine-tuning the reranking model, we utilized the Tamtanai dataset to enhance and evaluate its performance in reranking legal code documents.

Thai Legal Document Retrieval-Augmented Generation Database After selecting 31 legal codes suitable for the everyday life of the general public to create the Tamtanai dataset, we also added 5 legal codes relevant for legal examinations. We then cleaned these 36 legal codes by removing unnecessary sections, such as articles that have been repealed and are no longer in use. Before integrating them into the database for RAG, We divided the documents into subsections based on specific subtopics, with each subsection forming a chunk. Each chunk corresponds to a chapter defined in the legal code, where each chapter contains several sections focused on a single topic. In total, we created 617 chunks. Additionally, we stored the cleaned and reformatted data along with metadata indicating which legal sections are included in each chunk and which sections are still currently in use.

3.2 Overall Framework

An overview of the framework is shown in Fig. 1. Firstly, when receiving a user query, we first classify whether the incoming query belongs to the legal domain. If it does not belong to the legal domain, we use the base model to respond. However, if it is within the legal domain, it enters the RAG process. To enhance the efficiency of our RAG system, we implemented a two-stage process involving both keyword search and contextual search, followed by a reranking step. For the keyword search, once a user query is received, we extract important keywords and eliminate unnecessary ones by removing those with low IDF values, which helps avoid the retrieval of documents with extraneous terms. After that, we use the remaining keywords to search for documents containing those keywords. For the contextual search, we use the user query to find documents with a similarity score greater than or equal a set threshold. Then, we combine the documents obtained from both the keyword search and the contextual search and apply a reranking model to rerank the documents. After obtaining documents from the reranking model, these documents are inserted into a prompt template for the LLM to generate a response.

In our end-to-end evaluation, after integrating all modules, we investigate how many legal documents, chosen by the reranking model, should be included in the prompt to maximize the LLM's effectiveness in answering legal questions.

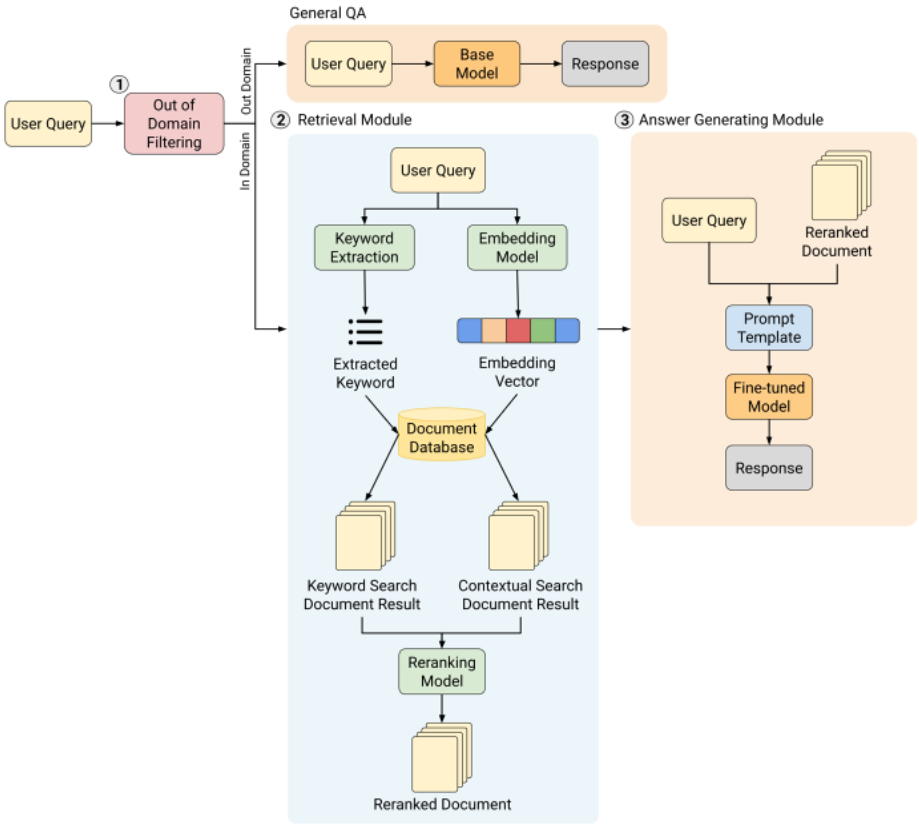

Fig. 1. Overview of the Framework.

3.3 Large Language Model Selection and Query Classifier Model

In selecting the most suitable LLM for our experiments, we focused on models that support the Thai language and have a size of 7B-8B. These models were fine-tuned and evaluated with the Tamtanai dataset. The model that achieved the highest BERTScore (details in Section 4.2) was selected as the best model. For the query classifier model, we fine-tuned the paraphrase-multilingual-mpnet-base-v2 using the han-instruct-dataset-v2.0 dataset and the Tamtanai dataset to effectively classify queries within the legal domain.

3.4 Retrieval-Augmented Generation System

In the contextual search segment, we first determine the most suitable contextual search model using the mean reciprocal rank (MRR) metric. For the reranking process, we selected the bge-reranker-v2-m3 model, which was fine-tuned using

the Tamtanai dataset. During the fine-tuning, questions in the dataset were used as queries, the most relevant legal document served as the positive document, and five other documents acted as negative documents. These negative documents were retrieved from contextual search to identify which ones are similar but not identical to the positive document. Lastly, we identify the most effective method by comparing the accuracy of document retrieval across different approaches: (1) contextual search, (2) keyword search and (3) reranking, contextual search and reranking, and a combination of keyword search, contextual search, and reranking to determine which yields the highest efficiency. For this experiment, we do not report using keyword search alone because it doesn't provide a clear ranking. Keyword search only checks for the presence of keywords, which can result in less relevant documents being ranked equally with more relevant ones if they have the same number of matching keywords. Thus, it must always be followed by a ranking process.

3.5 Enhancing Large Language Model

In this study, we fine-tuned the LLMs with two datasets: a legal question-and-answer dataset and a legal multiple choice exam dataset. To enhance the LLM, we primarily used the Tamtanai dataset for fine-tuning. We also explored multi-task fine-tuning to assess improvements in question-answering performance when combined with other datasets, comparing this approach to single-task fine-tuning on similar datasets. The Thai attorney exam dataset was used for this experiment, while the WangchanX-Legal-ThaiCCL-RAG dataset served as the similar dataset for comparison. We used QLoRA [8] to fine-tune the model. For the fine-tuning process, we configured the hyperparameters as follows: We trained the model for 3 epochs with a per-device training batch size of 3 and a per-device evaluation batch size of 4. We chose PagedAdamW32bit [20] as the optimizer and set the learning rate to 2.5e-4. Additionally, the temperature was set to 0.1.

3.6 Prompt Optimization

In this study, we aim to identify the most effective prompt for answering legal questions for the general public and the one best suited for performing well on legal examinations. This involves comparing different types of prompts: the normal instruction prompt, the chain of thought (COT) prompt, and the IRAC prompt [35,36]. The COT prompt instructs the model to first review the sub-sections and critical parts of the legal code before answering the question. The IRAC prompt, on the other hand, is based on a legal reasoning framework represented by the acronym: I for issue – identify the legal issue by examining the question clearly; R for rule – thoroughly review the information provided and apply the relevant laws or principles; A for application/analysis – analyze and apply the rule to the given information while examining all sub-sections if applicable; and C for conclusion – provide a summarized answer based on the appropriate legal principles.

4 Implementation Details and Evaluation Metrics

4.1 Implementation Details

For all LLM experiments, including fine-tuning and inference, we used four A100 GPUs. For the RAG experiments, we utilized FAISS [10] as a vector store for vector search and used LangChain for processing legal data to store in the vector database, employing four A100 GPUs as well. For the GPT-4o settings, we selected the model "gpt-4o-2024-08-06" and adjusted the temperature to 0 and max tokens to 2048. For calculating BERTScore, we used the deberta-xlarge-mnli [12] model as the scorer.

4.2 Large Language Model Evaluation Metrics

To evaluate the performance of LLMs on text summarization tasks, we utilize BERTScore [38] and ROUGE Score [19] as key metrics. BERTScore measures the similarity between the candidate and reference texts [3], using pre-trained contextual embeddings from BERT to match words in the sentences through cosine similarity. The ROUGE Score includes ROUGE-N and ROUGE-L. ROUGE-N, such as ROUGE-1 and ROUGE-2, calculates precision, recall, and F1-score based on n-gram overlaps between the candidate and reference texts. ROUGE-L focuses on the longest common subsequence (LCS) shared between the texts, calculating precision, recall, and F1-score based on the LCS length. This approach is particularly useful for judging semantic similarity and content coverage by considering sequences of words that match, regardless of their order [3].

4.3 Retrieval-Augmented Generation Evaluation Metrics

In evaluating the RAG process for selecting and assessing the contextual search model, we use MRR. However, we also assess the entire RAG process using metrics such as recall, accuracy@1, @2, @5, @10, MRR, and time. We do not report precision when evaluating the entire RAG process. because the ground truth consists of only one subsection from the code of law. This would result in a very low precision value, making precision an unsuitable metric for evaluating the entire RAG process. In this study, the ground truth contains only a single subsection because the small-sized LLM used here does not perform well when multiple subsections are included; providing multiple subsections may confuse the model and lead to errors.

5 Results

5.1 Results of Large Language Model Selection

This experiment aims to select the most effective LLM for use in the answer-generating module in Fig. 1. Table 1 demonstrates that the typhoon2-qwen2.5-7b-instruct model outperforms other Thai-supporting LLMs when fine-tuned

with the Tamtanai training dataset and tested on the Tamtanai test dataset. Not only does it achieve the highest BERTScore, but the Technical Report for Qwen2.5 [33], which serves as the base model for this version, shows that it also outperforms Qwen-2 and Llama 3.1 across various tasks. Furthermore, the Technical Report of typhoon2-qwen2.5-7b-instruct [26] indicates that the model's performance has been enhanced by fine-tuning with long context prompts, enabling it to analyze legal code documents effectively, even when fed with extensive legal texts. For these reasons, typhoon2-qwen2.5-7b-instruct has been chosen as the primary LLM for these experiments.

Table 1. BERTScore F1 for each model when fine-tuned on the Tamtanai training dataset and tested on the Tamtanai test dataset. Bold values indicate the winner.

Model	BERTScore F1 (↑)
typhoon2-qwen2.5-7b-instruct	**0.8979**
SeaLLMs-v3-7B-Chat	0.8965
Openthaigpt1.5-7B-instruct	0.8960
OpenThaiLLM-Prebuilt-7B	0.8912
Pathumma-llm-text-1.0.0	0.8900
Sailor2-8B-Chat	0.8898
llama3.1-typhoon2-8b-instruct	0.8888

5.2 Results of Contextual Search Model Selection

This experiment aims to select the most effective contextual search model for integration into the retrieval module illustrated in Fig 1. Table 2 presents the mean reciprocal rank (MRR) for each model when evaluated on the Tamtanai test set, providing a clear comparison of their retrieval effectiveness. The multilingual-e5-large model achieved the highest MRR score among all models in our experiment, indicating a superior ability to accurately rank and retrieve relevant documents compared to other candidates. This strong performance suggests that multilingual-e5-large can better distinguish relevant texts from irrelevant ones across diverse legal queries. Consequently, we have chosen the multilingual-e5-large as the primary contextual search model for our study. To further optimize retrieval accuracy, we fine-tuned its similarity threshold parameter to 0.55 and set the top k value to 50, ensuring the most relevant results are consistently retrieved.

Table 2. MRR and execution time (seconds) for each model tested on the Tamtanai test dataset. Bold values indicate the winner.

Model	MRR (↑)	Time (↓)
multilingual-e5-large	**0.7079**	0.0152
multilingual-e5-large-instruct	0.6719	0.0157
multilingual-e5-base	0.6345	0.0096
multilingual-e5-small	0.6348	**0.0095**

5.3 Results of Fine Tuning Reranking Model

This experiment aims to enhance the performance of the reranking model in the retrieval module in Fig.1. In Table3, it is shown that when tested on the Tamtanai test dataset, the Top-1 Accuracy after fine-tuning is 12.49% higher than before fine-tuning. This increase demonstrates the substantial impact that fine-tuning has on the model's ability to accurately rank relevant legal documents and improve overall retrieval effectiveness.

Table 3. Top-1 accuracy for both base model and fine-tuned model tested on the Tamtanai test dataset. Bold values indicate the winner.

bge-reranker-v2-m3	Accuracy@1 (↑)
Base Model	0.8198
Fine-Tuned Model	**0.9222**

5.4 Results of Retrieval-Augmented Generation Techniques

This experiment aims to compare RAG methods in the legal domain to identify the most effective RAG method for the retrieval module in Fig. 1. In Table 4, it can be seen that using a combination of keyword search, contextual search, and reranking provides the best performance in retrieving documents related to the code of law because each method offers unique strengths that complement one another. Keyword search is effective for matching specific terms, contextual search captures the meaning behind queries, and reranking refines the results to boost relevance. This combination creates a comprehensive search process where keywords help identify all potential matches, context ensures these matches are meaningful, and reranking optimizes the final results. Given the complex language and specific terminology often used in legal documents, this approach ensures both precise term matching and the consideration of legal context, thereby reducing the retrieval of irrelevant documents.

Table 4. Accuracy@1, @2, @5, @10, MRR, and retrieval time (seconds) for each RAG methods tested on the Tamtanai test dataset. Bold values indicate the winner.

Metrics	C	K+R	C+R	**K+C+R**
Retrieval Size (↓)	**44.68**	177.85	**44.68**	195.42
Recall (↑)	0.9782	0.9709	0.9782	**1.0000**
Accuracy@1 (↑)	0.6053	0.9370	0.9540	**0.9637**
Accuracy@2 (↑)	0.7094	0.9370	0.9685	**0.9879**
Accuracy@5 (↑)	0.8402	0.9661	0.9758	**0.9952**
Accuracy@10 (↑)	0.9056	0.9709	0.9782	**1.0000**
MRR (↑)	0.7064	0.9525	0.9633	**0.9784**
Retrieval Time (↓)	**0.0143**	0.3161	0.1246	0.2906

K stands for Keyword search, C is Contextual search, and R stands for Reranking.

5.5 Results of Large Language Model Enhancement

This experiment systematically compares fine-tuning methods for enhancing LLM performance in the answer-generating module illustrated in Fig 1, focusing on both legal question answering (QA) tasks and legal exam performance. As shown in Table 5, our results demonstrate that fine-tuning with both QA and multiple-choice exam datasets consistently yields better outcomes than using a QA dataset alone. Adding legal exam questions to training provides a clear improvement in the model's performance on both QA and exam-style tasks—a benefit greater than just increasing the QA dataset size. Even when QA and exam datasets are combined equally, performance exceeds that of the QA-only approach, though it falls just short of results from increasing exam data alone. This suggests that including exam-specific questions adds valuable knowledge for diverse legal benchmarks. Overall, our experiments confirm that the best results come from fine-tuning with both dataset types: the resulting LLM outperforms GPT-4o on legal QA tasks and matches its performance on legal exams, highlighting the value of diverse, task-relevant data for domain-specific optimization.

Table 5. BERTScore F1, ROUGE Scores, and multiple-choice exam scores for each model tested on the Tamtanai test dataset (2^{nd} - 5^{th} columns) and the Thai Attorney Exam test dataset (last column). Bold values indicate the winner.

Model	F1 (↑)	R-1 (↑)	R-2 (↑)	R-L (↑)	Exam (↑)
GPT-4o	0.8591	0.6590	0.6037	0.6256	47/80
LLM	0.8979	0.7719	0.7231	0.7402	34/80
LLM+EX	0.9159	0.8201	0.7717	0.7852	38/80
LLM+QA	0.9133	0.8076	0.7693	0.7820	31/80
LLM+QA+EX	**0.9367**	**0.8589**	**0.8273**	**0.8376**	**46/80**
LLM+QA(H)+EX(H)	0.9137	0.8109	0.7719	0.7853	36/80

LLM refers to the typhoon2-qwen2.5-7b-instruct model fine-tuned with the Tamtanai training dataset. QA denotes fine-tuning using the WangchanX-Legal-ThaiCCL-RAG dataset. EX involves fine-tuning with the Thai attorney exam dataset. The (H) indicates using half of that dataset. F1 means BERTScore F1, R-1 means ROUGE-1, R-2 means ROUGE-2, R-L means ROUGE-L, and Exam refers to the multiple-choice exam scores.

5.6 Results of Prompt Optimization

This experiment aims to identify the most effective prompt template for answering legal questions for the general public in the answer-generating module in Fig. 1. Table 6 demonstrates that using a normal instruction prompt yields the best results for answering legal questions. This is because the Tamtanai dataset is specifically designed for general question-answering related to legal inquiries for the public, rather than being structured for IRAC or Chain of Thought (CoT) formats. Therefore, when the model is fine-tuned with the Tamtanai dataset and

inference is conducted using a normal instruction prompt, it performs better than when using prompts designed for legal reasoning. Additionally, in terms of time efficiency, the normal instruction prompt requires the least amount of processing time compared to other prompt formats, likely because it is less complex and demands fewer computational resources, allowing for faster response times.

In Table 7, although the COT prompt shows the highest efficiency in legal exams, the score differences across the three prompt formats are minimal and not statistically significant. This likely stems from random processes during inference in LLMs, which can naturally introduce slight variations in results. Thus, any prompt format can be effectively used for legal exams; however, for scenarios where time efficiency is crucial, a normal instruction prompt offers significant practical benefits due to its simplicity and speed.

Table 6. BERTScore F1, ROUGE scores, and response time (seconds) for testing with di!erent prompt formats on the Tamtanai test dataset. Bold values indicate the winner.

Prompt	BERTScore F1 (↑)	ROUGE-1 (↑)	ROUGE-2 (↑)	ROUGE-L (↑)	Time (↓)
Normal	**0.9367**	**0.8589**	**0.8273**	**0.8376**	**37.41**
COT	0.9336	0.8541	0.8229	0.8345	37.95
IRAC	0.9346	0.8535	0.8233	0.8333	38.03

Table 7. Multiple-choice exam scores and response time (seconds) for testing with different prompt formats on the Thai Attorney Exam test dataset. Bold values indicate the winner.

Prompt	Exam Scores (↑)	Time (↓)
Normal	46/80	**37.74**
COT	**47/80**	38.50
IRAC	45/80	38.93

5.7 End-to-End Question and Answering Evaluation

The purpose of this experiment is to determine the optimal number of legal documents, selected by the reranking model, that should be provided to the LLM for effective end-to-end question answering. After integrating the most effective RAG approach, the best-performing LLM, and the optimal prompt format into our legal QA framework, Table 8 demonstrates that including only one legal document in the prompt improves QA performance compared to using multiple documents. This is likely because presenting too much legal text at once can make the LLM uncertain about which specific article to reference when constructing an answer, thus reducing the overall effectiveness and reliability of the legal QA system.

Table 8. BERTScore F1, ROUGE scores, and response time (seconds) for end-to-end evaluation tested on the Tamtanai test dataset. Bold values indicate the winner.

N#	BERTScore F1 (↑)	ROUGE-1 (↑)	ROUGE-2 (↑)	ROUGE-L (↑)	Time (↓)
1	**0.9052**	**0.7818**	**0.7398**	**0.7558**	**41.89**
2	0.8924	0.7481	0.7002	0.7155	42.16
5	0.8681	0.6867	0.6252	0.6453	48.15
10	0.8487	0.6221	0.5378	0.5669	70.36

N# is legal document input in the prompt.

6 Conclusion

This research presents a Thai legal question-answering (QA) framework for the general public. We evaluated several large language models (LLMs) and enhanced performance using a multitask dataset. For the retrieval-augmented generation (RAG) component, we optimized prompt templates, retrieval strategies, and document count. Experiments on the Thai Attorney Exam and our dataset showed that Typhoon2-qwen2.5-7b-instruct achieved the highest BERTScore. We used paraphrase-multilingual-mpnet-base-v2 to filter non-legal queries, handled by the base model. The RAG pipeline combined multilingual-e5-large for retrieval and bge-reranker-v2-m3 for reranking, outperforming single-method approaches. Training with QA and multiple-choice data improved results, and instruction-based prompts yielded the best performance when using only one retrieved document.

For future work, we plan to extend the framework to serve both general users and legal experts. Additionally, we aim to deploy the system in real-world settings to enhance accessibility and reproducibility. Furthermore, we will continue to optimize the inference time for improved performance.

Acknowledgments. This project was made possible through the support of the National Science and Technology Development Agency (NSTDA), which provided access to the LANTA HPC computing resources. We would also like to express our gratitude to the Chula Computer Engineering Graduate Scholarship for their financial support.

References

1. Akarajaradwong, P., Pothavorn, P., Chaksangchaichot, C., Tasawong, P., Nopparatbundit, T., Nutanong, S.: Nitibench: A comprehensive studies of llm frameworks capabilities for thai legal question answering. arXiv preprint arXiv:2502.10868 (2025)
2. ARIAI, F., DEMARTINI, G.: Natural language processing for the legal domain: A survey of tasks, datasets, models, and challenges. ACM Comput. Surv **1**(1) (2024)
3. Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., et al.: A survey on evaluation of large language models. ACM transactions on intelligent systems and technology **15**(3), 1–45 (2024)

4. Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., Liu, Z.: M3-embedding: Multilinguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In: Findings of the Association for Computational Linguistics ACL 2024. pp. 2318–2335 (2024)
5. Chlapanis, O., Androutsopoulos, I., Galanis, D.: Archimedes-aueb at semeval-2024 task 5: Llm explains civil procedure. In: Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024). pp. 1607–1622 (2024)
6. Colombo, P., Pires, T.P., Boudiaf, M., Culver, D., Melo, R., Corro, C., Martins, A.F., Esposito, F., Raposo, V.L., Morgado, S., et al.: Saullm-7b: A pioneering large language model for law. arXiv preprint arXiv:2403.03883 (2024)
7. Deng, C., Mao, K., Dou, Z.: Learning interpretable legal case retrieval via knowledge-guided case reformulation. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. pp. 1253–1265 (2024)
8. Dettmers, T., Pagnoni, A., Holtzman, A., Zettlemoyer, L.: Qlora: Efficient finetuning of quantized llms. Advances in neural information processing systems **36**, 10088–10115 (2023)
9. Dou, L., Liu, Q., Zhou, F., Chen, C., Wang, Z., Jin, Z., Liu, Z., Zhu, T., Du, C., Yang, P., et al.: Sailor2: Sailing in south-east asia with inclusive multilingual llms. arXiv preprint arXiv:2502.12982 (2025)
10. Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P.E., Lomeli, M., Hosseini, L., Jégou, H.: The faiss library. arXiv preprint arXiv:2401.08281 (2024)
11. Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Yang, A., Fan, A., et al.: The llama 3 herd of models. arXiv preprint arXiv:2407.21783 (2024)
12. He, P., Liu, X., Gao, J., Chen, W.: Deberta: Decoding-enhanced bert with disentangled attention. In: International Conference on Learning Representations
13. Iu, K.Y., Wong, V.M.Y.: Chatgpt by openai: The end of litigation lawyers? Available at SSRN 4339839 (2023)
14. Jiang, H., Zhang, X., Mahari, R., Kessler, D., Ma, E., August, T., Li, I., Pentland, A., Kim, Y., Roy, D., et al.: Leveraging large language models for learning complex legal concepts through storytelling. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 7194–7219 (2024)
15. Lai, J., Gan, W., Wu, J., Qi, Z., Philip, S.Y.: Large language models in law: A survey. AI Open (2024)
16. Lawyers Council Under The Royal Patronage: Compilation of exam questions and answer guidelines for the theoretical training in advocacy from session 16 to session 54. Dharmasarn Printing, Nonthaburi (2021)
17. Lawyers Council Under The Royal Patronage: Compilation of exam questions and answer guidelines for the theoretical training in advocacy from session 55 to session 59. Dharmasarn Printing, Nonthaburi (2021)
18. Lawyers Council Under The Royal Patronage: Compilation of exam questions and answer guidelines for the theoretical training in advocacy from session 60 to session 61. Dharmasarn Printing, Nonthaburi (2021)
19. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text summarization branches out. pp. 74–81 (2004)
20. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations
21. Mamooler, S., Lebret, R., Massonnet, S., Aberer, K.: An efficient active learning pipeline for legal text classification. NLLP 2022 **2022**, 345–358 (2022)

22. NECTEC: Openthaillm-prebuilt-7b (2025), https://medium.com/nectec/openthaillm-prebuilt-release-f1b0e22be6a5, retrieved February 18, 2025
23. NECTEC: Pathummallm-text-v 1.0.0 release (2025), https://medium.com/nectec/pathummallm-v-1-0-0-release-6a098ddfe276, retrieved February 18, 2025
24. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A.: Training language models to follow instructions with human feedback. Advances in neural information processing systems **35**, 27730–27744 (2022)
25. Phatthiyaphaibun, W.: Han instruct dataset (Apr 2024). https://doi.org/10.5281/zenodo.10935857, https://doi.org/10.5281/zenodo.10935857
26. Pipatanakul, K., Manakul, P., Nitarach, N., Sirichotedumrong, W., Nonesung, S., Jaknamon, T., Pengpun, P., Taveekitworachai, P., Na-Thalang, A., Sripaisarnmongkol, S., et al.: Typhoon 2: A family of open text and multimodal thai large language models. arXiv preprint arXiv:2412.13702 (2024)
27. Rodrigues, R.: Legal and human rights issues of ai: Gaps, challenges and vulnerabilities. Journal of Responsible Technology **4**, 100005 (2020)
28. Song, K., Tan, X., Qin, T., Lu, J., Liu, T.Y.: Mpnet: Masked and permuted pre-training for language understanding. Advances in neural information processing systems **33**, 16857–16867 (2020)
29. Sourdin, T.: Judge v robot?: Artificial intelligence and judicial decision-making. University of New South Wales Law Journal, The **41**(4), 1114–1133 (2018)
30. Wan, Z., Zhang, Y., Wang, Y., Cheng, F., Kurohashi, S.: Reformulating domain adaptation of large language models as adapt-retrieve-revise: A case study on chinese legal domain. In: Findings of the Association for Computational Linguistics ACL 2024. pp. 5030–5041 (2024)
31. Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., Wei, F.: Multilingual e5 text embeddings: A technical report. arXiv preprint arXiv:2402.05672 (2024)
32. Yang, A., Yang, B., Hui, B., Zheng, B., Yu, B., Zhou, C., Li, C., Li, C., Liu, D., Huang, F.: Qwen2 technical report. arXiv preprint arXiv:2407.10671 (2024)
33. Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., et al.: Qwen2. 5 technical report. arXiv preprint arXiv:2412.15115 (2024)
34. Yao, R., Wu, Y., Wang, C., Xiong, J., Wang, F., Liu, X.: Elevating legal llm responses: Harnessing trainable logical structures and semantic knowledge with legal reasoning. arXiv preprint arXiv:2502.07912 (2025)
35. Yu, F., Quartey, L., Schilder, F.: Legal prompting: Teaching a language model to think like a lawyer. arXiv preprint arXiv:2212.01326 (2022)
36. Yu, F., Quartey, L., Schilder, F.: Exploring the effectiveness of prompt engineering for legal reasoning tasks. In: Findings of the association for computational linguistics: ACL 2023. pp. 13582–13596 (2023)
37. Yuenyong, S., Viriyayudhakorn, K., Piyatumrong, A., Jaroenkantasima, J.: Openthaigpt 1.5: A thai-centric open source large language model. arXiv preprint arXiv:2411.07238 (2024)
38. Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: Bertscore: Evaluating text generation with bert. In: International Conference on Learning Representations
39. Zhang, W., Chan, H.P., Zhao, Y., Aljunied, M., Wang, J., Liu, C., Deng, Y., Hu, Z., Xu, W., Chia, Y.K., et al.: Seallms 3: Open foundation and chat multilingual large language models for southeast asian languages. arXiv preprint arXiv:2407.19672 (2024)