Taming the Babel of Queries: Enhancing RAG Systems for Intellectual Property via Synthetic Multi-Perspective Fine-tuning

Anonymous ACL submission

Abstract

NLP systems of the Intellectual property (IP) field face significant challenges due to the diverse ways in which users express queries, such as colloquial language and ambiguous terms. These issues hinder the effectiveness of Retrieval-Augmented Generation (RAG) systems in IP filed. In this paper, we propose a novel Multi-Angle Question Generation and Retrieval Fine-Tuning Method (MQG-RFM) that leverages large language models (LLMs) as agents to simulate diverse user queries. By generating multiple variations of queries and fine-tuning the retrieval model with hard negative mining, MQG-RFM improves the retrieval accuracy and answer generation quality in patent-related Q&A scenarios. MQG-RFM offers a simple and generalizable solution that does not require complex architectural changes, making it an efficient and scalable method for personalized deployment in small and medium-sized IP agencies. Experimental results on a Taiwan patent Q&A dataset show 185.62% improvement in retrieval accuracy on the Patent Consultation dataset and 262.26% improvement on the Novel Patent Technology Report dataset, with 14.22% and 53.58% improvements in generation quality,respectively, over the baselines.

1 Introduction

> *"The same truth can be questioned in countless ways."* — Thomas Aquinas

Half a year ago, at a patent agency in China, a lawyer tried to retrieve the legal status of a new invention from the database. However, after entering a misspelled English term, the system only returned a few irrelevant documents. Later, another user asked in a relatively complex colloquial manner, "I want to know if this device is still under review," but received a completely unrelated old patent document instead. These cases frequently appear in logs. Our further analysis of approximately 50,00

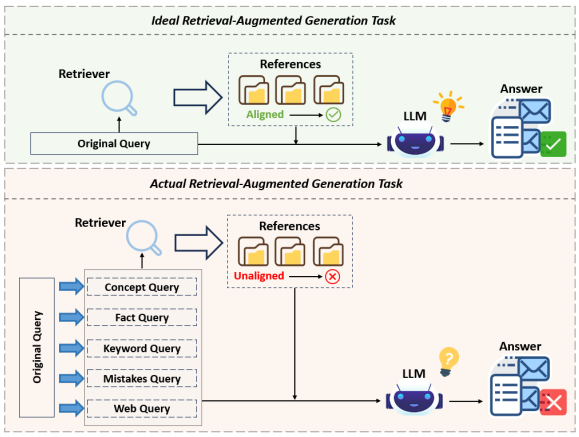

Figure 1: RAG system usage scenarios in the IP field

real user query records revealed that over 30% of requests involved spelling errors, colloquial expressions, or vague keywords. These examples indicate that users often pose questions in a variety of ways, as Hegel (1977) said: *the same entity can be interpreted by different languages, cultures, and backgrounds.* Similarly, the same answer can be asked through multiple different questions. This art of language offers insights for the system design in the field of intellectual property (IP). The diversity of language, spelling mistakes, colloquial expressions, and even ambiguous keywords can prevent natural language processing (NLP) systems from correctly understanding user needs (Joshi et al., 2020; Ranta and Goutte, 2021). From a system development perspective (Kelly, 2009), this not only affects the user experience but can also lead to inaccurate information retrieval, ultimately preventing users from obtaining correct answers. Therefore, how to address the impact of these diverse expressions on search results has become a core issue that needs to be solved in system design for IP field.

Although current large language models (LLMs) and retrieval technologies have made progress in handling user queries, these technologies still

face some limitations when applied to the IP field (Kirchhübel and Brown, 2024; Shalaby and Zadrozny, 2019). First, LLMs may suffer from hallucination issues when dealing with highly specialized content like patent law and technical terms (Dahl et al., 2024). Second, existing retrieval models like BM25, while improving retrieval performance through keyword matching and synonym substitution, cannot effectively capture the deeper semantics in user queries (Zhao et al., 2024), our subsequent experimental results also proved this phenomenon. In practical applications, the same question is often expressed by users in multiple different ways (e.g., keyword-based query or concept query) (Moffat and Zobel, 1996; Gavankar et al., 2016). For example, users may ask: "What is the legal status of the invention?" "Patent review of the invention," or "Has the invention been approved?" These queries are semantically similar but essentially the same, but due to the different ways of expression, the vector retrieval model may treat them as different questions instead of the same query (Ai et al., 2016). In other words, in the vector space, the embedding model cannot accurately capture the same intent behind them, causing them to be misunderstood as different queries and thus fail to match relevant information.

In addition to the diversity of inquiries, users may also raise intersectional questions that involve multiple fields in a single query (Whalen, 2018). For example, a query not only pertains to the patent review status but also involves legal terms related to patent protection and technical details.

In the above context, the Retrieval-Augmented Generation (RAG) could serve as an initial solution. For example, when a user asks how to query the status of a confidential patent, the RAG system can not only provide guidance on the review status, but also give relevant patent protection terms, thereby providing the user with a comprehensive answer covering multiple dimensions of information such as legal background, technical implementation and review status. However, the success of RAG is closely tied to the accuracy of the retrieval step. If the retrieval results are inaccurate, the generated answer may contain incorrect legal or technical basis. As shown in Figure 1, the ideal RAG system is designed to retrieve corresponding references. However, various queries often arise in practical use leading to retrieval failures. Thus, how to accurately understand the intention of user queries and ensure that the RAG system in the IP field can correctly respond to users' diverse queries is the primary motivation of this paper. In addition, our secondary motivation is to promote the usability of RAG in the IP field by proposing a simple and generalizable methodology, which facilitates rapid personalized deployment by agencies and solves the practical problem of inaccurate retrieval.

Currently, academia has proposed a variety of methods to optimize the performance of RAG systems. Based on the differences in technical paths, the existing paradigm of method can be grouped into three types: (1) *Data-to-Tune*: This paradigm does not alter the model architecture but focuses on prompt engineering or fine-tuning the model to optimize performance. For example, Kim et al. (2023) fine-tunes LLMs by constructing a CoT (Chain of Thought) dataset to explicitly guide the reasoning process, helping models exhibit clearer and more understandable reasoning. (2) *Flow-to-Run*: This paradigm modifies the external retrieval-generation interaction process without changing the model's internal parameters or structure, such as Adaptive-RAG (Jeong et al., 2024). (3) *Build-to-Learn*: This paradigm modifies the model's internal network architecture or mathematical functions to enhance capabilities, like mixture of experts (MoE) (Artetxe et al., 2022). These paradigms represent different optimization tracks for innovation: *Data-to-Tune* focuses on parameter tuning, *Flow-to-Run* emphasizes process design, and *Build-to-Learn* pursues algorithmic refinement. It is important to note that innovation is not limited to mathematical breakthroughs—choosing the appropriate technical path to meet the demands of different domains is crucial. For example, when aiming for answer diversity in general domains, *Build-to-Learn* might be preferred; however, in some niche fields, where rapid personalized deployment and low-cost fine-tuning are vital for small and medium agencies, *Data-to-Tune* may be the preferred choice.

In this paper, we propose a Multi-Angle Question Generation and Retrieval Fine-Tuning Method (MQG-RFM) using *Data-to-Tune* paradigm. Specifically, we use LLMs as agents to simulate different users to generate inquiries with different preferences. Then, we use hard negative mining to use generated inquiries as labels to fine-tune the retrieval model to improve the retrieval ability of the RAG system. Notably, we do not rely on complex mathematical constructions or reshaping the network architecture. Instead, we combine prompt engineering with fine-tuning, providing

a straightforward yet effective method to solve difficult problems in the Q&A scenario of the IP field. Through our method, the retrieval model learns how to map different questions to the same answer in the vector space, enhancing its adaptability to diverse queries. Experimental results show that MQG-RFM significantly improves retrieval accuracy and answer generation quality on the Taiwan patent Q&A dataset. This paper makes the following contributions:

(1) **A novel methodology MQG-RFM** is proposed to utilize LLMs simulate diverse user questions to generate data for fine-tuning, solving the problem of existing RAG systems in IP filed being unable to handle multiple question expressions.

(2) **A simple and generalizable solution** is provided for without complex architectural changes, our approach offers a practical, low-cost, and highly efficient solution that can be rapidly adapted to various real-world scenarios in the IP field.

(2) **Improves the retrieval model's ability** to map semantically similar queries expressed in various ways, thereby enhancing the effectiveness of RAG systems in matching user queries with relevant patent information.

2 Related Work

According to the hierarchical analysis theory of model optimization in the field of machine learning (Houlsby et al., 2019), the improvement of model performance can be achieved by adjusting parameters, optimizing processes or reconstructing architectures. Based on the hierarchical analysis theory, from a perspective of technical paths, the optimization of RAG can also be summarized into these three paradigms: *Data-to-Tune*, *Flow-to-Run*, and *Build-to-Learn*. These paradigms improve RAG performance from the perspectives of parameter tuning, process design, and architectural transformation. The following will introduce the definition, related works, and pros and cons of each paradigm, and compare their essential differences and industrial adaptability, finally reveal the suitable paradigm in the IP field.

2.1 Data-to-Tune

The paradigm of *Data-to-Tune* emphasizes optimizing models through data augmentation, hint engineering, or parameter fine-tuning without changing the model's internal architecture or external interaction process. The *Data-to-Tune* approach benefits from its simplicity and flexibility as it does not require structural changes to the model. For example, Yu et al. (2023) train the augmentation-adapted retriever (AAR) using preference learning on the source language model to better adapt to a target model (e.g., migrating from Flan-T5 to InstructGPT) to reduce the adaptation cost of heterogeneous models. Similarly, Mao et al. (2024) proposed a query rewriting method named RaFe based on reranker feedback to optimize RAG. By first fine-tuning the model to generate rewritten queries and then using reranker scores for feedback training, their fine-tuning method and prompt engineering enhances the model's ability to generate queries better aligned with retrieval targets (Mao et al., 2024). RaFe does not require changes to the model architecture or external processes, and only optimizes the model through data enhancement and feedback training, which demonstrates the effectiveness of the *Data-to-Tune* paradigm for RAG system enhancement. Another example is Self-Knowledge guided Retrieval augmentation (SKR) proposed by Wang et al. (2023) that trains a mechanism to adaptively decide whether to use the retriever and identify the most similar queries from the training data based on the input query, thereby reducing the interference of irrelevant retrieval on the generated results to optimize RAG.

2.2 Flow-to-Run

Flow-to-Run focuses on the workflow design and inter-module collaboration strategies to optimize RAG effects. Rather than modifying the network architecture of models, this paradigm enhances performance by improving how modules interact. As Jeong et al. (2024) emphasized, the generation module and the retrieval module do not operate independently, but are collaborative work. Jeong et al. (2024) believe that a good RAG workflow is that generation module can dynamically decide whether to call the retrieval module based on task requirements, while the retrieval module provides customized information based on the requirements of the generation model to optimize the generation results. Therefore, Jeong et al. (2024) proposed Adaptive-RAG, which optimizes the RAG system by dynamically selecting retrieval strategies and collaborating with the generation module through a classifier. In the specific application, Louis et al. (2024) proposed the Retrieve-Then-Read (R2R) process to apply the RAG to the legal field. They effectively verified the feasibility of RAG in ver-

Aspect	Data-to-Tune	Flow-to-Run	Build-to-Learn
Object	Parameter/Data	Process/Workflow	Architecture/Function
Cost	Low (data-centric)	Moderate (workflow design)	High (structural redesign)
Adaptability	Fast domain adaptation	Dynamic query handling	High-precision tasks
Limitation	Data coverage dependency	Latency-accuracy tradeoff	Deployment scalability

Table 1: Comparison of Paradigms and Their Suitability for IP

tical fields. Another noteworthy work is the summarized retrieval (SuRE) framework by Kim et al. (2024), which enhances module synergy through a structured process that includes candidate answer generation, document summarization, and answer verification. Specifically, the SuRE framework enables each module to play a specific role in the workflow through the process design of candidate answer generation, retrieval document summary, and answer verification.

2.3 Build-to-Learn

The *Build-to-Learn* paradigm refers to modifying the neural network architecture or related function within the model to make the model more capable of learning. The *Build-to-Learn* paradigm can significantly enhance model capabilities, but it often requires complex architectural design and massive computational resources for training and validating. OPEN-RAG (Islam et al., 2024) is a representative work of the *Build-to-Learn* paradigm, which converts LLMs into a parameter-efficient sparse MoE and enhances the generation ability by introducing special reflection tokens. Similarly, Asai et al. (2024) proposed a method named Self-RAG that introduced reflection tokens inside the model to modify the generation process, allowing the model to perform self-evaluation at each generation step and adjust the generation strategy based on the evaluation results. In addition, the REPLUG method by Shi et al. (2024) innovates the retriever's training process by calculating the similarity between each retrieved document and the query during the training phase, and then uses the Softmax method to calculate the selection probability of K documents. REPLUG has made structural innovations for the retriever, enabling it to select the best document.

2.4 Comparison and Suitability for IP

Each of the three paradigms offers unique advantages and limitations. The paradigm of *Data-to-Tune* is ideal for scenarios where model architecture does not require changes, and prompt engineering or fine-tuning with high-quality data is feasible. *Flow to Run* paradigm can design workflow for LLM interactions based on the specific needs of the scenario. Finally, the *Build-to-Learn* paradigm provides innovation in neural network architecture but at the cost of increased complexity and resource requirements to train and validate the feasibility of the reconstructed network.

In the context of IP, several core requirements of NLP applications have been identified through extensive research based on practical considerations (Gossen and Nürnberger, 2013; Aristodemou and Tietze, 2018; Ilin and Kelli, 2024), including (1) Handling expression diversity: Most user queries contain spelling errors or colloquial expressions; (2) Traceability of legal basis: The answer must be strictly related to the provisions of the Patent Law; (3) Feasibility of small and medium-sized institutions: Efficient deployment needs to be achieved with limited computing power. These requirements are illustrated in Table 1, which compares the three paradigms in terms of their adaptability to the IP field. Similarly, Aristodemou and Tietze (2018) also pointed out that NLP systems in the IP field require processing complex legal language and terminology, which places high demands on the adaptability and accuracy of models. In addition, Ilin and Kelli (2024) explored the need to consider computational resource limitations and efficient deployment when deploying LLMs in small enterprises.

Therefore, it is crucial to choose the appropriate paradigm for different scenarios. Among the three paradigms, *Data to Tune* is the optimal choice due to its low cost, high flexibility, and fast iteration capability. Although there are many existing methods for the *Data to Tune* paradigm, such as AAR and RaFe (Yu et al., 2023; Mao et al., 2024), their strong dependence on data coverage has not been effectively addressed in the field of IP. Existing methods often cannot cover all relevant term variants, resulting in inaccurate results and missing information in RAG system queries. Therefore, we propose the MQG-RFM method in this paper, which inherits the efficiency advantages of the *Data*

to Tune paradigm and fine-tunes the retrieval model by simulating multidimensional problems to solve these gaps.

3 Method

The methodology of MQG-RFM involves leveraging LLMs as agents simulating various users in IP field for query generation, hard negative mining for data augmentation, and fine-tuning retrieval model, as illustrated in figure 2.

3.1 Role Play

Our approach begins by utilizing a LLM as an agent to generate multiple queries based on the original query q_{orig} in the logs, which goal is to simulate different user preferences by generating various types of queries. Specifically, for each original query q_{o}, the agent generate a set of k queries $Q_{\text{gen}} = \{q^1_{\text{type}_i}, q^2_{\text{type}_i}, \ldots, q^k_{\text{type}_i}\}$ for query types with different preferences T_i separately. We define the set of query types $T = \{T_1, T_2, \ldots, T_k\}$, where each type corresponds to a different seeking behavior.

Each query type T_i is used to generate queries q_{gen_i} based on the original query q_{o}:

$$q^{T_i}_{\text{gen}_j} = LLM(q_{\text{o}}, T_i)$$

where $q^{T_i}_{\text{gen}_i}$ represents the i-th generated query of type T. By generating k queries per query type T_i, we ensure that each user intent is well-represented.

3.2 Data Augmentation

After generating the diverse queries, we proceed to the data augmentation phase using hard negative mining. This process involves pairing each generated query $q^{T_i}_{\text{gen}_j}$ with both positive and negative answers to create augmented training examples.

Positive Example: The answer a_{o} corresponding to the original query q_{o} is used as the positive example for each generated query, which is assumed to be correct and relevant for the generated query:

$$Pos^{T_i}_{\text{gen}_j} = (q_{\text{o}}, a_{\text{o}}) \quad \text{for each} \quad q^{T_i}_{\text{gen}_j}$$

Negative Example: For each generated query $q^{T_i}_{\text{gen}_j}$, we create a negative example by selecting an answer a_{neg} that do not correspond to $q^{T_i}_{\text{gen}_j}$. To ensure the difficulty of the negative examples, we randomly select a_{neg} from the set of answers that are not the correct answer for $q^{T_i}_{\text{gen}_j}$:

$$Neg^{T_i}_{\text{gen}_j} = (q_{\text{o}}, a_{\text{neg}}) \quad \text{for each} \quad q^{T_i}_{\text{gen}_j}$$

By generating positive and negative examples, we get a augmented dataset $(Pos^{T_i}_{\text{gen}_j}, Neg^{T_i}_{\text{gen}_j})$ that help the model distinguish between correct and incorrect answers.

3.3 Fine-Tuning the Retrieval Model

The final stage in MQG-RFM involves using the augmented dataset $(Pos^{T_i}_{\text{gen}_j}, Neg^{T_i}_{\text{gen}_j})$ to fine-tune the retriever. The positive and negative examples generated from the previous step are incorporated into the model's training set. The goal of this phase is to fine-tune the retrieval model $R(\theta)$, parameterized by θ, to correctly retrieve answers. The loss function $\mathcal{L}(\theta)$ is defined for fine-tuning as:

$$\mathcal{L}(\theta) = -\frac{1}{n}\sum_{j=1}^{n} \log \frac{\exp(\text{sim}(q^{T_i}_{\text{gen}_j}, Pos^{T_i}_{\text{gen}_j}))}{\sum_{j=1}^{B} \exp \Delta(\text{sim}(q^{T_i}_{\text{gen}_j}, Neg^{T_i}_{\text{gen}_j}))}$$

where B is the batch size. The loss function encourages the model to maximize the probability of retrieving the correct answer and minimize the probability of retrieving the incorrect answer. This leads to a model that is better able to distinguish between relevant and irrelevant answers, improving the quality of the retrieval process for RAG system.

4 Experimental Setup

4.1 Dateset

In order to simulate a realistic query scenario for IP, we use the real dataset related to patents and IP provided by the Taiwan government:

Patent Consultation (PC) Q&A This dataset is from the frequent Q&A of the Taiwan Patent Service Center. The content includes basic knowledge of patents, patent procedures, formal examination, change of application, annual fees, changes in patent rights, correction of patent rights, patent retrieval, patent attorney management and other Q&A related to IP business.

Novel Patent Technology Report (NPTR) Q&A This dataset contains frequent Q&A about novel patent technology reports in Taiwan, including the legal basis, acceptance and comparison of novel patent technology reports, and relevant regulations.

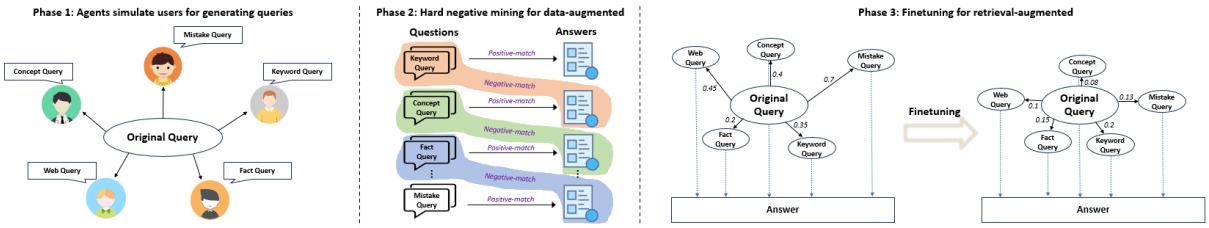

Figure 2: Implementation Process of MQG-RFM

4.2 Data-Augmented Analysis

We developed a platform [1] to facilitate data management for institutions shown in figure 12, enabling them to efficiently manage logs and perform analysis. The platform allows administrators to leverage LLM to play different user roles, thereby generating a range of queries for analysis based on specific user preferences.

In our study, we employed GPT-4 as an agent to simulate different user query preferences. To demonstrate the diversity of the generated queries compared to the original ones, we visualized the semantic distance between the generated and original queries using t-SNE. As shown in figure 3, the t-SNE plot maps these queries to a vector space, with each query positioned according to its semantic meaning. From the plot, it is evident that each generated query maintains a certain distance from the original queries, confirming that the model successfully produces diverse queries that differ from the initial set.

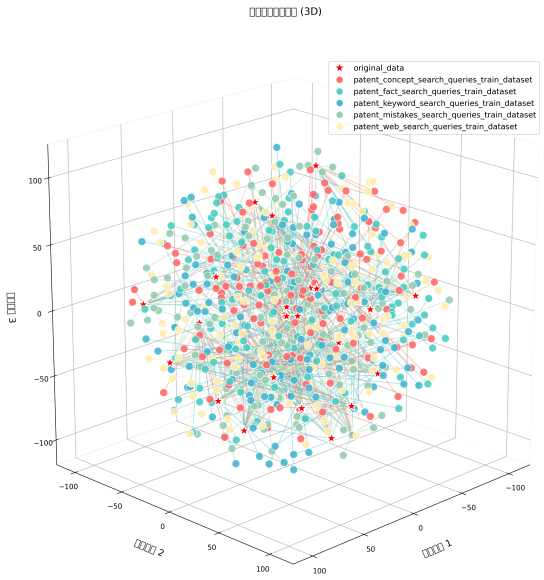

Figure 3: Spatial distance between the generated query and the original query

4.3 Metrics

We evaluate the performance of RAG methods using the following quantitative metrics:

ROUGE is used to evaluate the quality of generated answers by comparing them to references (Lin, 2004).

BLEU is used for evaluating generated text by comparing n-gram precision between the generated output and reference texts (Papineni et al., 2002).

BERT Score is calculated based on the cosine similarity between the embeddings of the predicted text and the reference text (Zhang et al.). For this paper, BERT-P measures how much of the predicted text matches the reference in terms of contextual embeddings, while BERT-R measures how much of the reference text is captured by the predicted text. The BERT-F1 is the harmonic mean of BERT-P and BERT-R, which balances both metrics.

NDCG measures the ranking quality of retrieved documents by considering both the relevance of the documents and their position in the ranked list.

Hit evaluates whether the system can successfully retrieve relevant items within the specified number of top results.

Precision measures the proportion of relevant documents among the retrieved documents.

4.4 Baselines

In our work, we compare our proposed MQG-RFM with several state-of-the-art methods that adopt the *Data-to-Tune*, *Flow-to-Run*, and *Build-to-Learn* paradigms. These methods include: **AAR** (Yu et al., 2023), **SKR** (Wang et al., 2023), **Self-RAG** Asai et al. (2024), **Adaptive-RAG** (Jeong et al., 2024), and **SuRE** Kim et al. (2024).

[1] https://anonymous.4open.science/r/ACL2025 − 6020/

4.5 Experimental Setup

For MQG-RFM, we employed the Chuxin-Embedding [2] after finetuning with 5 epoch as the retriever. The max input length of generator model is set to 4096. For approaches not utilizing custom-defined prompts, we applied a connected prompt, which is shown in the appendix. The methods in the baselines adopt the same settings and hyperparameters used by (Jin et al., 2024) in their work. All experiments are carried out on 4 NVIDIA 4090 GPUs.

5 Evaluation

In this section, we evaluate the performance of MQG-RFM in comparison to state-of-the-art retrieval models and RAG methods on both retrieval and generation tasks about the IP field.

5.1 Retrieval Model Comparison

We compare the retrieval performance of various embedding models bge-large-zh-v1.5[3], Dmeta-embedding-zh[4], stella-base-zh-v3-1792d[5], PatentSBERTa [6] on PC and NPTR datasets, as shown in Table 2.

5.2 Method Comparison

In order to compare the advancement of the proposed method in RAG, we also compared it with various state-of-the-art methods in terms of retrieval and generation. The results are shown in Tables 3 and 4.

5.3 Ablation Study

To investigate the effectiveness of our strategy in MQG-RFM, we compare method without fine-tuning on both generation and retrieval metrics, as shown in figure 4 and 5.

5.4 Robustness Test

To test the robustness of our approach, we evaluated our method under different generation models, specifically using Qwen14B [7] as the underlying generation model. This test demonstrates that our method performs excellently even when using a different generation model, further confirming its generalizability. The results are summarized in Table 5, which shows the performance across several evaluation metrics.

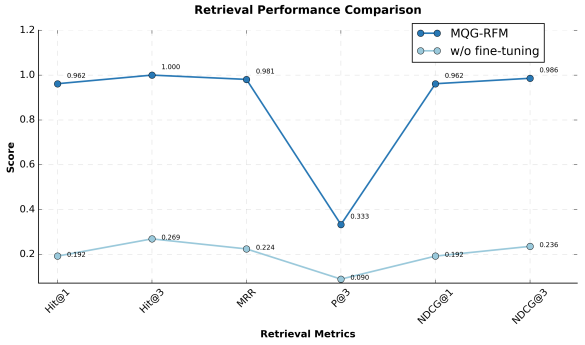

Figure 4: Ablation Experiment about Retrieval

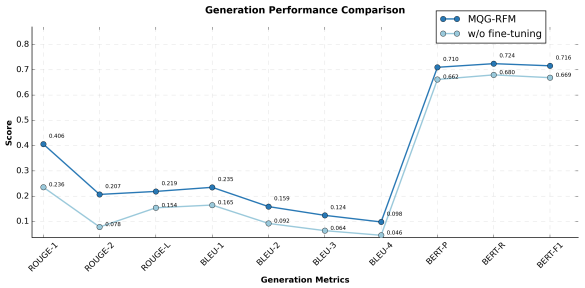

Figure 5: Ablation Experiment about Generation

6 Conclusion

In this paper, we addressed the critical challenge of handling diverse user queries in the IP field, where spelling errors, colloquial expressions, and ambiguous keywords often hinder accurate information retrieval. By focusing on the Data-to-Tune paradigm, our method enhances the ability of RAG systems to map semantically similar queries expressed in various ways to the same answer, thereby improving retrieval accuracy and answer generation quality. Our experimental results on the Taiwan patent dataset demonstrate that MQG-RFM significantly outperforms existing methods in handling diverse query expressions, providing a robust solution for real-world IP retrieval scenarios. The proposed method not only addresses the limitations of current RAG systems in the IP field but also offers a simple, generalizable, and cost-effective approach that can be rapidly deployed by small and medium-sized agencies. By combining prompt engineering with fine-tuning, MQG-RFM bridges the gap between user intent and system understanding, ultimately enhancing the usability and effectiveness of RAG systems in the IP domain.

[2] https://huggingface.co/chuxin-llm/Chuxin-Embedding
[3] https://huggingface.co/BAAI/bge-large-zh-v1.5
[4] https://huggingface.co/DMetaSoul/Dmeta-embedding-zh
[5] https://huggingface.co/infgrad/stella-base-zh-v3-1792d
[6] https://huggingface.co/AI-Growth-Lab/PatentSBERTa
[7] https://huggingface.co/Qwen/Qwen2.5-14B-Instruct

	Model	Hit@1	Hit@3	MRR	P@3	NDCG@1	NDCG@3
PC	bge-large-zh	0.563	0.751	0.673	0.250	0.563	0.675
	stella-base-zh	0.571	0.734	0.670	0.244	0.571	0.668
	Dmeta-embedding	0.621	0.779	0.712	0.259	0.621	0.713
	PatentSBERTa	0.148	0.252	0.218	0.084	0.148	**0.807**
	MQG-RFM (Ours)	**0.663**	**0.857**	**0.749**	**0.288**	**0.663**	0.777
NPTR	bge-large-zh	0.576	0.769	0.685	0.256	0.576	0.693
	stella-base-zh	0.538	0.846	0.675	0.282	0.538	0.707
	Dmeta-embedding	0.615	0.807	0.722	0.269	0.615	0.726
	PatentSBERTa	0.307	0.423	0.429	0.141	0.307	0.380
	MQG-RFM (Ours)	**0.961**	**1**	**0.980**	**0.333**	**0.961**	**0.985**

Table 2: Comparison of the retrieval performance of different retrievers

Dataset	Method	Hit@1	Hit@3	MRR	P@3	NDCG@1	NDCG@3
PC	ARR	0.184	0.038	0.089	0.082	0.036	0.022
	SKR	0.173	0.038	0.089	0.082	0.036	0.022
	SuRe	0.320	0.505	0.400	0.168	0.320	0.467
	Self-RAG	0.249	0.113	0.185	0.108	0.076	0.061
	Adaptive-RAG	0.232	0.082	0.128	0.121	0.071	0.049
	MQG-RFM (Ours)	**0.663**	**0.857**	**0.749**	**0.288**	**0.663**	**0.777**
NPTR	AAR	0.265	0.098	0.150	0.166	0.093	0.064
	SKR	0.206	0.056	0.141	0.157	0.082	0.055
	SuRe	0.192	0.269	0.224	0.089	0.192	0.235
	Self-RAG	0.225	0.099	0.152	0.046	0.035	0.028
	Adaptive-RAG	0.235	0.078	0.154	0.165	0.092	0.063
	MQG-RFM (Ours)	**0.961**	**1**	**0.980**	**0.333**	**0.961**	**0.985**

Table 3: Comparison of the retrieval performance of different RAG methods

Dataset	Method	ROUGE-1	ROUGE-2	ROUGE-L	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BERT-P	BERT-R	BERT-F1
PC	ARR	0.184	0.038	0.089	0.082	0.036	0.022	0.014	0.578	0.655	0.612
	SKR	0.173	0.038	0.089	0.082	0.036	0.022	0.014	0.578	0.655	0.612
	SuRe	0.033	0.011	0.026	0.007	0.006	0.006	0.005	0.501	0.357	0.413
	Self-RAG	0.249	**0.113**	**0.185**	0.108	0.076	0.061	0.050	**0.701**	0.629	0.661
	Adaptive-RAG	0.232	0.082	0.128	0.121	0.071	0.049	0.035	0.617	0.704	0.655
	MQG-RFM (Ours)	**0.265**	0.106	0.140	**0.132**	**0.084**	**0.061**	**0.052**	0.629	**0.725**	**0.671**
NPTR	AAR	0.265	0.098	0.150	0.166	0.093	0.064	0.044	0.646	0.682	0.661
	SKR	0.206	0.056	0.141	0.157	0.082	0.055	0.037	0.639	0.664	0.649
	SuRe	0.026	0.009	0.020	0.005	0.004	0.003	0.002	0.499	0.342	0.404
	Self-RAG	0.225	0.099	0.152	0.046	0.035	0.028	0.022	0.720	0.606	0.657
	Adaptive-RAG	0.235	0.078	0.154	0.165	0.092	0.063	0.045	0.661	0.679	0.668
	MQG-RFM (Ours)	**0.406**	**0.207**	**0.219**	**0.235**	**0.159**	**0.124**	**0.098**	**0.723**	**0.724**	**0.715**

Table 4: Comparison of the performance of different RAG methods in generation using DeepSeek

7 Limitations

While MQG-RFM demonstrates promising results, several limitations should be acknowledged: The effectiveness of MQG-RFM relies heavily on the quality of the LLM used for generating diverse user inquiries. If the LLM fails to capture the nuances of user queries or generates low-quality synthetic data, the fine-tuning process may be compromised; While the method is cost-effective for small and medium-sized agencies, scaling it to larger datasets or more complex retrieval tasks may require additional computational resources and optimization.

References

Qingyao Ai, Liu Yang, Jiafeng Guo, and W Bruce Croft. 2016. Analysis of the paragraph vector model for information retrieval. In *Proceedings of the 2016 ACM international conference on the theory of information retrieval*, pages 133–142.

Leonidas Aristodemou and Frank Tietze. 2018. The state-of-the-art on intellectual property analytics (ipa): A literature review on artificial intelligence, machine learning and deep learning methods for analysing intellectual property (ip) data. *World Patent Information*, 55:37–51.

Mikel Artetxe, Shruti Bhosale, Naman Goyal, Todor Mihaylov, Myle Ott, Sam Shleifer, Xi Victoria Lin, Jingfei Du, Srinivasan Iyer, Ramakanth Pasunuru, Giridharan Anantharaman, Xian Li, Shuohui Chen, Halil Akin, Mandeep Baines, Louis Martin, Xing Zhou, Punit Singh Koura, Brian O'Horo, Jeffrey Wang, Luke Zettlemoyer, Mona Diab, Zornitsa Kozareva, and Veselin Stoyanov. 2022. Efficient large scale language modeling with mixtures of experts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11699–11732, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E Ho. 2024. Large legal fictions: Profiling legal hallucinations in large language models. *Journal of Legal Analysis*, 16(1):64–93.

Chetana Gavankar, Yuan-Fang Li, and Ganesh Ramakrishnan. 2016. Explicit query interpretation and diversification for context-driven concept search across ontologies. In *The Semantic Web–ISWC 2016: 15th International Semantic Web Conference, Kobe, Japan, October 17–21, 2016, Proceedings, Part I 15*, pages 271–288. Springer.

Tatiana Gossen and Andreas Nürnberger. 2013. Specifics of information retrieval for young users: A survey. *Information Processing & Management*, 49(4):739–756.

Georg Wilhelm Friedrich Hegel. 1977. Phenomenology of spirit. *Trans. AV Miller/Oxford: Oxford UP*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR.

Ilya Ilin and Aleksei Kelli. 2024. Natural language, legal hurdles: Navigating the complexities in natural language processing development and application. *Journal of the University of Latvia. Law*, 17:44–67.

Shayekh Bin Islam, Md Asib Rahman, K S M Tozammel Hossain, Enamul Hoque, Shafiq Joty, and Md Rizwan Parvez. 2024. Open-RAG: Enhanced retrieval augmented reasoning with open-source large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14231–14244, Miami, Florida, USA. Association for Computational Linguistics.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2024. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7036–7050, Mexico City, Mexico. Association for Computational Linguistics.

Jiajie Jin, Yutao Zhu, Xinyu Yang, Chenghao Zhang, and Zhicheng Dou. 2024. Flashrag: A modular toolkit for efficient retrieval-augmented generation research. *arXiv preprint arXiv:2405.13576*.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Diane Kelly. 2009. [link].

Jaehyung Kim, Jaehyun Nam, Sangwoo Mo, Jongjin Park, Sang-Woo Lee, Minjoon Seo, Jung-Woo Ha, and Jinwoo Shin. 2024. Sure: Summarizing retrievals using answer candidates for open-domain qa of llms. *arXiv preprint arXiv:2404.13081*.

Seungone Kim, Se Joo, Doyoung Kim, Joel Jang, Seonghyeon Ye, Jamin Shin, and Minjoon Seo. 2023. The CoT collection: Improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning. In *Proceedings of the 2023*

Conference on Empirical Methods in Natural Language Processing, pages 12685–12708, Singapore. Association for Computational Linguistics.

Christin Kirchhübel and Georgina Brown. 2024. Intellectual property rights at the training, development and generation stages of large language models. In *Proceedings of the Workshop on Legal and Ethical Issues in Human Language Technologies @ LREC-COLING 2024*, pages 13–18, Torino, Italia. ELRA and ICCL.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. 2024. Interpretable long-form legal question answering with retrieval-augmented large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22266–22275.

Shengyu Mao, Yong Jiang, Boli Chen, Xiao Li, Peng Wang, Xinyu Wang, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2024. Rafe: Ranking feedback improves query rewriting for rag. *arXiv preprint arXiv:2405.14431*.

Alistair Moffat and Justin Zobel. 1996. Self-indexing inverted files for fast text retrieval. *ACM Transactions on Information Systems (TOIS)*, 14(4):349–379.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Aarne Ranta and Cyril Goutte. 2021. Linguistic diversity in natural language processing. *Traitement Automatique des Langues*, 62(3):7–11.

Walid Shalaby and Wlodek Zadrozny. 2019. Patent retrieval: a literature review. *Knowledge and Information Systems*, 61:631–660.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. REPLUG: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8371–8384, Mexico City, Mexico. Association for Computational Linguistics.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023. Self-knowledge guided retrieval augmentation for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10303–10315, Singapore. Association for Computational Linguistics.

Ryan Whalen. 2018. Boundary spanning innovation and the patent system: Interdisciplinary challenges for a specialized examination system. *Research Policy*, 47(7):1334–1343.

Zichun Yu, Chenyan Xiong, Shi Yu, and Zhiyuan Liu. 2023. Augmentation-adapted retriever improves generalization of language models as generic plug-in. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2421–2436, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. 2024. Dense text retrieval based on pretrained language models: A survey. *ACM Transactions on Information Systems*, 42(4):1–60.

A Example Appendix

Prompt of generating concept seeking query

下面是有关台湾便利民眾查詢及各項申請準備參考使用的專利服務平台平日答詢之常見問答：

{context_str}

你的任務是仿照原題目的問法，生成 {K} 個依舊能夠對應原答案的概念尋求查詢的題目。概念尋求查詢的定义：需要多個句子來回答的抽象問題。

Translation:

The following are frequent Q&A about Taiwan's patent service platform, which is used by the public for inquiries and reference in the preparation of various applications:

{context_str}

Your task is to generate {K} concept search queries that still correspond to the original answer, following the method used in the original question. The definition of concept seeking query: an abstract question that requires multiple sentences to answer

Figure 6: Prompt of generating concept seeking query

Prompt of generating fact seeking query

下面是有关台湾便利民眾查詢及各項申請準備參考使用的專利服務平台平日答詢之常見問答：

{context_str}

你的任務是仿照原題目的問法，生成 {K} 個依舊能夠對應原答案的事實尋求查詢的題目。事實尋求查詢的定义：具有單一、明確答案的查詢。

Translation:

The following are frequent Q&A about Taiwan's patent service platform, which is used by the public for inquiries and reference in the preparation of various applications:

{context_str}

Your task is to generate {K} fact seeking queries that still correspond to the original answer, following the method used in the original question. The definition of fact seeking query: Queries that have a single, unambiguous answer.

Figure 7: Prompt of generating fact seeking query

Dataset	Method	ROUGE-1	ROUGE-2	ROUGE-L	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BERT-P	BERT-R	BERT-F1
	ARR	0.094	0.037	0.055	0.010	0.006	0.005	0.003	0.611	0.445	0.513
PC	SKR	0.054	0.015	0.039	0.011	0.006	0.004	0.003	0.543	0.395	0.455
	SuRe	0.025	0.006	0.015	0.002	0.002	0.001	0.001	0.481	0.315	0.380
	Self-RAG	0.008	0.003	0.006	0.003	0.002	0.002	0.001	0.502	0.38	0.437
	Adaptive-RAG	0.085	0.040	0.062	0.014	0.011	0.010	0.009	0.582	0.411	0.480
	MQG-RFM (Ours)	0.112	0.061	0.084	0.019	0.016	0.015	0.013	0.609	0.430	0.502
	AAR	0.094	0.037	0.055	0.010	0.006	0.005	0.003	0.611	0.445	0.513
NPTR	SKR	0.072	0.023	0.044	0.001	0.002	0.001	0.001	0.632	0.450	0.524
	SuRe	0.025	0.006	0.015	0.002	0.002	0.001	0.001	0.481	0.315	0.380
	Self-RAG	0.008	0.002	0.004	0.003	0.002	0.001	0.001	0.493	0.395	0.437
	Adaptive-RAG	0.100	0.048	0.067	0.015	0.011	0.008	0.006	0.6258	0.44	0.516
	MQG-RFM (Ours)	0.320	0.221	0.235	0.077	0.069	0.064	0.059	0.809	0.587	0.676

Table 5: Comparison of the performance of different RAG methods in generation using Qwen

Table 6: Positive and negative examples generated by the LLM for fine-tuning

Dataset	**classified as positive**	**classified as negative**
MuSiQue	**Query** 如果我不是专利权人，但我被指控侵犯新型专利，我可以如何申请技术报告来协助处理这一争议? If I am not the patent owner but I am accused of infringing a utility patent, how can I request a technical report to assist in the dispute?	**Query** 如果我不是专利权人，但我被指控侵犯新型专利，我可以如何申请技术报告来协助处理这一争议? If I am not the patent owner but I am accused of infringing a utility patent, how can I request a technical report to assist in the dispute?
	Document 1 一、如遇非专利权人有相同或类似商品为商业上之实施者，为早获得技术报告，专利权人可以检附相关证明文件，如专利权人之书面通知、非专利权人之广告目录或其他商业上实施事实之书面资料，申请新型专利技术报告(参照专利法第115条第5项及专利法施行细则第43条)。本局将于申请后6个月内完成新型专利技术报告。二、若新型专利技术报告之申请人为非专利权人，为协助当事人间侵权争议之处理，在非专利权人提具事涉专利侵权争议之相关证明文件，如已遭新型专利权人提出专利侵权之存证信函、涉及专利侵权诉讼案件之起诉书或诉讼传票等文件资料者，本局亦将优先作成新型专利技术报告。三、对于不符合商业上实施规定的技术报告申请案，本局将于受理通知函中叙明“有关商业上实施的主张不符合专利法第115条第5项之规定”。 1. If a non-patent holder has the same or similar products that are commercially implemented, in order to obtain a technical report as soon as possible, the patent holder may attach relevant supporting documents, such as a written notice from the patent holder, an advertising catalog of the non-patent holder, or other written materials of commercial implementation facts, to apply for a new patent technical report (refer to Article 115, Paragraph 5 of the Patent Law and Article 43 of the Patent Law Enforcement Rules). The Office will complete the new patent technical report within 6 months after the application. 2. If the applicant for the new patent technical report is a non-patent holder, in order to assist in the handling of infringement disputes between the parties, the Office will also give priority to preparing a new patent technical report if the non-patent holder provides relevant supporting documents related to the patent infringement dispute, such as a registered letter of patent infringement filed by the new patent holder, an indictment or a litigation summons involving a patent infringement lawsuit, and other documents and materials. 3. For technical report applications that do not meet the requirements for commercial implementation, the Office will state in the acceptance notice that "the claim regarding commercial implementation does not meet the requirements of Article 115, Paragraph 5 of the Patent Law."	**Document 2** 新型专利技术报告申请专利范围中每一请求项逐项比对后，引用文献之记载分别说明如下：一、关于引用文献一览表之记载审查人员制作新型专利技术报告时，选取适用之全部先前技术文献相关资料，总记载于“引用文献一览表”栏位内。二、关于引用文献之记载(一)每一请求项下之引用文献，针对该请求项所适用之文献，仅须记载该文献的序号。每一请求项之引用文献多寡不一，撰写方式可为“引用文献：1”或“引用文献：1及3”。(二)适用情形如下;1.否定新颖性等要件，应记载与请求项之记载内容最接近或最适当之先前技术文献。2.无法否定新颖性等要件，应记载该技术领域中一般技术水准之参考文献（代码6）。 After comparing each claim item in the patent application scope of the new patent technical report, the records of the cited documents are explained as follows: 1. Records on the list of cited documentsthe examiner prepares the new patent technical report, he selects all applicable previous technical documents and records them in the "list of cited documents" column. 2. Records on cited documents(1) The cited documents under each claim item are the documents applicable to the claim item, and only the serial number of the document needs to be recorded. The number of cited documents for each claim item varies, and the writing method can be "cited document: 1" or "cited document: 1 and 3". (2) The applicable situations are as follows; 1. To negate the requirements of novelty, the previous technical documents that are closest or most appropriate to the content recorded in the claim item should be recorded. 2. If the requirements of novelty cannot be negated, the reference documents of the general technical level in the technical field should be recorded (code 6).

Prompt of generating keyword query

下面是有关台湾便利民眾查詢及各項申請準備參考使用的專利服務平台平日答詢之常見問答：

{context_str}

你的任務是仿照原題目的問法，生成 {K} 個依舊能夠對應原答案的關鍵字查詢的題目。關鍵字查詢的定义：僅包含關鍵標識符詞的簡短查詢。

Translation:

The following are frequent Q&A about Taiwan's patent service platform, which is used by the public for inquiries and reference in the preparation of various applications:

{context_str}

Your task is to generate {K} keyword queries that still correspond to the original answer, following the method used in the original question. The definition of keyword query: Short queries containing only key identifier words.

Figure 8: Prompt of generating keyword query

Prompt of generating query with spelling

下面是有关台湾便利民眾查詢及各項申請準備參考使用的專利服務平台平日答詢之常見問答：

{context_str}

你的任務是仿照原題目的問法，生成 {K} 個依舊能夠對應原答案的拼寫錯誤查詢的題目。拼寫錯誤查詢的定义：包含拼寫錯誤、換位和常見拼寫錯誤的查詢。

Translation:

The following are frequent Q&A about Taiwan's patent service platform, which is used by the public for inquiries and reference in the preparation of various applications:

{context_str}

Your task is to generate {K} queries with spelling mistakes that still correspond to the original answer, following the method used in the original question. The definition of query with spelling mistakes: Queries containing misspellings, transpositions, and common spelling errors.

Figure 9: Prompt of generating query with spelling

Prompt of generating web search-like

下面是有关台湾便利民眾查詢及各項申請準備參考使用的專利服務平台平日答詢之常見問答：

{context_str}

你的任務是仿照原題目的問法，生成 {K} 個依舊能夠對應原答案的網頁搜索查詢的題目。網頁搜索查詢的定义：類似於通常輸入搜索引擎的簡短查詢轉換。

Translation:

The following are frequent Q&A about Taiwan's patent service platform, which is used by the public for inquiries and reference in the preparation of various applications:

{context_str}

Your task is to generate {K} web search-like queries that still correspond to the original answer, following the method used in the original question. The definition of web search-like query: Similar to short queries commonly entered into search engines.

Figure 10: Prompt of generating web search-like query

Connected Prompt

System Prompt: You are an expert in the field of intellectual property who is good at answering questions based on given documents. Please answer the questions based on the given documents.

User Prompt:
Question: [the user query]
Context: [the related context]

Figure 11: Connected Prompt

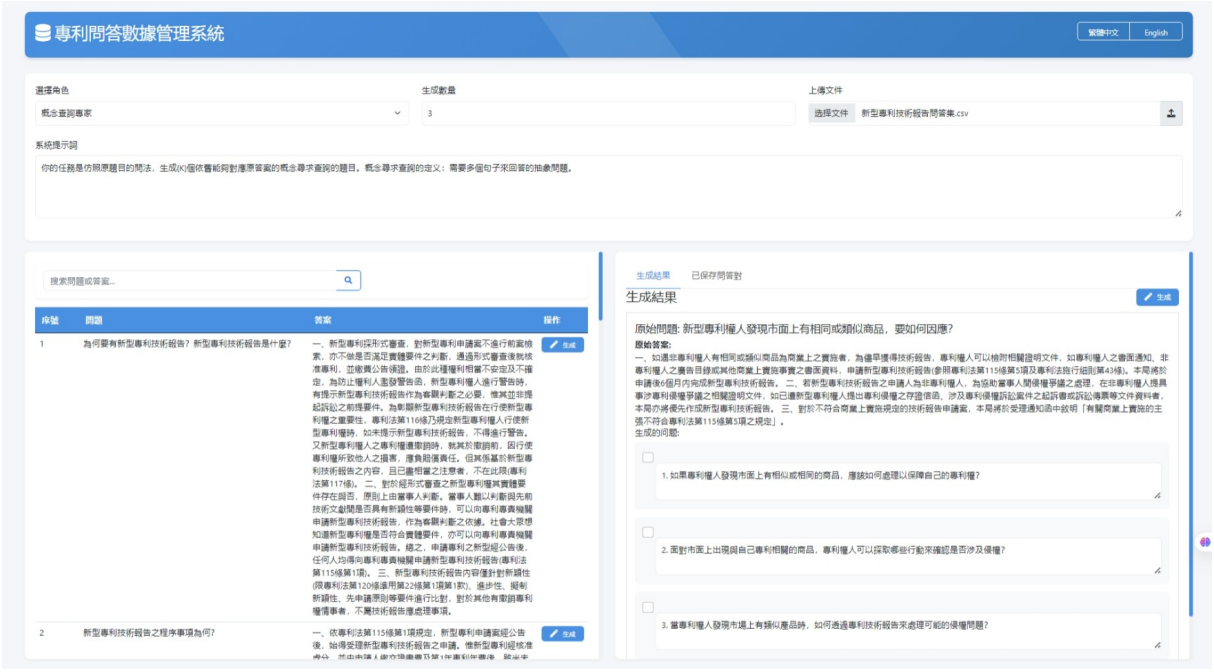

Figure 12: Data management platform for large language model generation queries