2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

Latent Query Alignment for Enhanced Domain-specific Retrieval-Augmented Generation

Yijun Bei[1], Yan Jiang[1], Bin Zhao[1], Lihua Yu[2], Zhaoyu Zhong[3], and Yao Zhu[3]

[1] School of Software Technology, Zhejiang University, 315100 Ningbo, China
{beiyj,yan.jiang,bzhao}@zju.edu.cn
[2] Bank of Hangzhou, 310003 Hangzhou, China
peterylh@163.com
[3] Zhejiang Yufeng Information Technology Co., Ltd, 315201 Ningbo, China
{zhongzy,zhuy}@shatayy.com

Abstract. Large Language Models (LLMs) demonstrate impressive generalization capabilities in various natural language processing tasks but often encounter performance degradation, particularly in domain-specific applications due to hallucinations and semantic mismatches. We propose LaQuA, a novel retrieval framework leveraging latent query alignment to bridge the semantic gap between user queries and specialized domain documents. LaQuA integrates three core innovations: latent query generation using LLMs, contrastive alignment with similarity-constrained synthetic queries, and a semantic bridging inference mechanism employing proxy queries. Comprehensive experiments on public benchmarks and a custom domain-specific dataset show that LaQuA significantly improves retrieval quality compared to standard dense retrievers and pseudo-query approaches. Additionally, evaluation within a Retrieval-Augmented Generation (RAG) pipeline demonstrates consistent enhancements in factual accuracy and content relevance across multiple language models and domains. Our findings suggest that latent query-driven semantic alignment substantially mitigates hallucinations and improves LLM performance in knowledge-intensive, domain-specific tasks.

Keywords: Large Language Models · Dense Retrieval · Latent Query Generation · Contrastive Learning · Retrieval-Augmented Generation.

1 Introduction

Large Language Models (LLMs) have recently attracted significant research attention due to their strong performance across a wide range of natural language processing (NLP) tasks. Models such as ChatGPT [1], Qwen [2], and DeepSeek [26] have exhibited impressive generalization capabilities across multiple benchmarks and application domains. Despite this progress, the performance of LLMs in domain-specific applications remains limited. In particular, when applied to specialized fields such as finance, medicine, law, or manufacturing[27,37,38], LLMs often struggle to meet the precision

and reliability requirements of these domains. One of the most pressing challenges in such scenarios is the phenomenon of hallucination.

Hallucination refers to the tendency of LLMs to produce outputs that are linguistically fluent and contextually plausible, yet factually incorrect. This issue is especially critical in professional domains where factual accuracy is essential. A key reason for hallucination lies in the nature of LLM training data. General-purpose LLMs are predominantly pre-trained on large-scale open-domain corpora, which seldom capture the depth, domain-specific terminology, and structured patterns characteristic of specialized fields. As a result, these models often lack the necessary knowledge to accurately address domain-specific queries [3,17,43].

To mitigate these limitations, two main lines of research have emerged. The first line focuses on embedding domain-specific knowledge directly into the model through approaches such as continuous pre-training and parameter-efficient fine-tuning [13]. These methods aim to align the model parameters with domain-relevant data distributions, thereby enhancing the model's internal representation of specialized knowledge. The second line explores incorporating external knowledge at inference time by augmenting the model's input context. One representative strategy is the Retrieval-Augmented Generation (RAG) framework [23], which improves output quality by retrieving relevant information from external knowledge sources and injecting it into the model's context window [14,30].

Within retrieval-augmented paradigms, the effectiveness of the retrieval component plays a decisive role in the system's overall performance [9,14]. Similar to the limitations observed in general-purpose LLMs, dense retrievers based on Transformer architectures often exhibit degraded performance when applied to specialized domains, particularly if trained only on general-domain corpora [20]. A major challenge stems from the misalignment between user queries and domain documents. Queries are typically short, intent-driven, and focused, whereas documents are often lengthy, verbose, and include both relevant and extraneous information. This semantic and structural discrepancy hinders dense dual-encoder retrievers from learning a unified embedding space that can accurately capture the query-document relationship [8,46].

To address these challenges, we propose **Latent Query Alignment (LaQuA)** in Fig. 1, a novel dense retrieval framework tailored for domain-specific applications. The core idea is to leverage LLMs to generate latent queries, which are synthetic queries that semantically represent the content of a document. These queries serve as a bridge between user intents and domain documents. Within our framework, latent queries play a dual role: they guide representation learning during training and enhance semantic matching during inference. The overall design incorporates several innovations across three key components:

- **Latent Query Generation.** For each document in the domain corpus, we use an LLM to generate a set of latent queries that simulate plausible user intents answerable by the document. To ensure semantic diversity and avoid redundancy, we apply prompt engineering, tune generation parameters (e.g., presence penalties), and filter

generated queries based on lexical and embedding-level similarity. The resulting latent queries provide rich semantic anchors for learning and evaluation.

- **Contrastive Alignment with Synthetic Queries.** We fine-tune a dense retriever using a contrastive learning objective, where training triplets are constructed from documents (anchors), their associated latent queries (positives), and sampled negatives. Instead of naive random sampling, we introduce a similarity-constrained negative sampling strategy that adaptively selects challenging but informative negatives based on their embedding similarity to the anchor document. We further inject noise into embedding space to promote robustness and generalization.
- **Semantic Bridging via Proxy Queries.** During inference, we implement a two-stage retrieval process that leverages proxy queries as semantic intermediaries. First, given a user query, we retrieve the most semantically relevant proxy queries from the index. Each proxy query is linked to a source document in the corpus, allowing us to route the user query through a semantically aligned pathway. This bridging mechanism reduces mismatch between short, intent-driven user queries and complex domain-specific documents. To improve both recall and precision, we further integrate direct document retrieval results and apply a cross-encoder to rerank the combined candidate set.

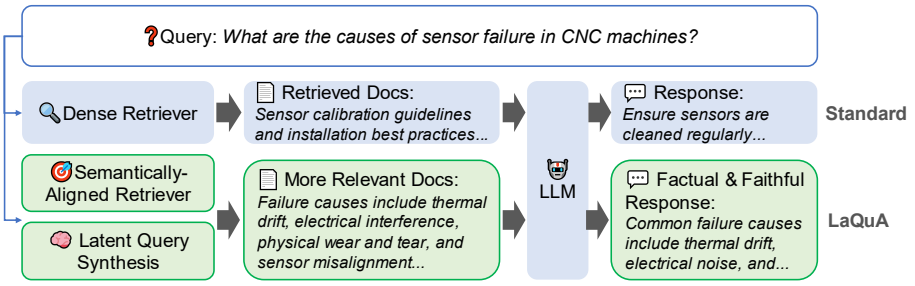

Fig. 1. RAG's effectiveness relies on retrieval quality. We propose Latent Query Alignment (LQA), using proxy queries to bridge the user intent-domain content gap. This enhances retrieval precision for more factual RAG outputs.

To evaluate the generalizability and practical value of LaQuA, we adopt a two-stage experimental strategy designed to balance methodological rigor with real-world relevance. The first stage centers on domain-specific retrieval tasks conducted under standardized conditions. Leveraging public evaluation benchmarks, this phase investigates how effectively LaQuA aligns user queries with relevant content across specialized domains, providing a controlled setting for comparison with existing retrieval approaches.

While such benchmarks offer consistency and comparability, they may fall short in capturing the complexities of real-world deployment, particularly when portions of the evaluation data overlap with sources seen during pretraining. To address this limitation, the second stage introduces a custom evaluation scenario designed to reflect more practical application needs. We construct a proprietary dataset in a non-English domain-specific setting, comprising documents and queries representative of downstream information-seeking tasks. This setup enables us to examine LaQuA's behavior in a low-

resource, cross-lingual environment, where retrieval quality has a direct impact on downstream generation. We integrate LaQuA into a retrieval-augmented generation pipeline and use the retrieved content to support answer generation, simulating the demands of knowledge-intensive real-world applications. Our approach offers a scalable and generalizable solution for improving LLM utility in knowledge-intensive settings.

2 Related Work

2.1 Dense Retrieval

Dense retrieval utilizing pre-trained language models (PLMs) have gained prominence due to their ability to represent both queries and documents as continuous vector embeddings, which capture semantic relationships beyond surface-level keyword matching [12,33,46]. Among the most common architectures for dense retrieval are the cross-encoder [11,32,35] and bi-encoder [4,7,19,40] models. In the cross-encoder architecture, both the query and document are concatenated and processed together by a single PLM, enabling the model to learn fine-grained interactions between the two. This approach is highly effective for tasks requiring precise relevance ranking, but it can be computationally expensive, especially in large-scale retrieval systems [34,45]. In contrast, the bi-encoder architecture [4,16] processes the query and document independently through separate encoders, generating individual embeddings for each. Among the most common architectures for dense retrieval are the cross-encoder and bi-encoder models. In the cross-encoder architecture, both the query and document are concatenated and processed together by a single PLM, enabling the model to learn fine-grained interactions between the two. In contrast, the bi-encoder architecture processes the query and document independently through separate encoders, generating individual embeddings for each [33,46].

2.2 Contrastive Learning

Contrastive learning is a method that learns embedding spaces by pulling positive pairs closer and pushing negative pairs apart. Initially developed for image representation learning, it has recently been applied to natural language processing tasks. One representative application of contrastive learning in sentence embedding is the SimCSE model [10]. This method constructs positive pairs by applying dropout to the same input sentence and treating the resulting different encoded representations as positives, while other sentences in the batch serve as negatives. This strategy enables unsupervised semantic alignment. In its supervised variant, SimCSE further leverages sentence pairs from natural language inference datasets as positive examples, which significantly improves the model's ability to capture sentence similarity. Subsequently, methods such as CoSERT [44] refined the contrastive objective by designing a pairwise ranking loss, aiming to better align with tasks that involve semantic matching or sentence ranking.

Moreover, contrastive learning has been widely applied in tasks such as textual similarity, helping models learn fine-grained semantic distinctions between sentences, enabling more accurate similarity judgments [22,31].

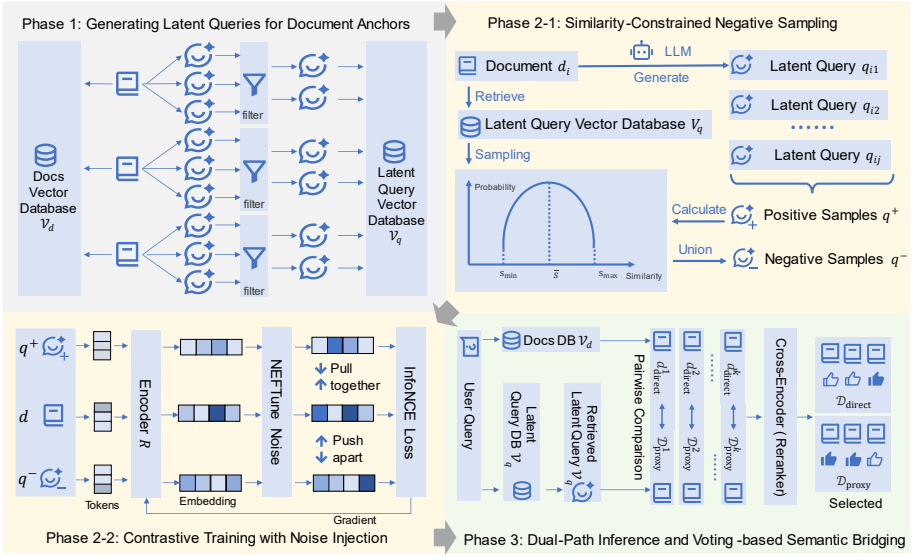

Fig. 2. Overview of the LaQuA framework. Our method consists of three key components. (1) Latent Query Generation: We use an LLM to produce diverse intent-driven queries for each document, which are filtered based on embedding similarity to form a latent query index. (2) Contrastive Alignment with Synthetic Queries: Using similarity-constrained sampling, we construct training triplets and apply InfoNCE loss with NEFTune noise injection to learn robust representations. (3) Semantic Bridging via Proxy Queries: During inference, we retrieve documents via both proxy and direct paths. A cross-encoder reranker selects the more relevant path using a voting-based strategy.

2.3 Pseudo Relevance Feedback

Pseudo Relevance Feedback (PRF) [5,28,36,42] is a widely employed technique in information retrieval designed to enhance search results without requiring explicit user feedback. The fundamental idea behind PRF is that the system assumes the top-ranked documents from an initial search are relevant to the user's query, even in the absence of direct feedback. This assumption allows the system to extract additional terms, phrases, or concepts from these initial documents, which are then used to expand or reformulate the original query. The expanded query is subsequently used to perform another round of retrieval, ideally resulting in more relevant documents than the original query could yield.

The advantage of PRF is that it can enhance search performance without requiring explicit user feedback, thus saving time and effort. However, it relies heavily on the quality of the initial retrieval, and if the top documents are not truly relevant, the feedback can lead to worse results. PRF is commonly used in search engines, recommendation systems, and natural language processing tasks like question answering and semantic search [25,40].

3 Methods

We introduce a novel framework, illustrated in **Fig. 2**, designed to overcome the persistent semantic gap in domain-specific information retrieval. Our core innovation lies in the synergistic use of LLM-generated latent queries. These synthetic queries serve a dual purpose: firstly, they enable an advanced contrastive training methodology for the retrieval model, incorporating similarity-constrained negative sampling and NEFTune for enhanced robustness. Secondly, they function as dynamic semantic bridges (proxy queries) within a novel dual-path inference strategy, adaptively guiding retrieval towards relevant complex documents. The subsequent sections detail the key components of this approach: latent query generation (Section 3.1), contrastive model alignment (Section 3.2), and the semantic bridging inference mechanism (Section 3.3).

3.1 Latent Query Generation

Given a domain-specific document corpus $\mathcal{D} = \{\mathrm{d}_1, \mathrm{d}_2, \ldots, \mathrm{d}_\mathrm{N}\}$, our goal is to generate semantically rich and diverse queries that reflect the latent user intents associated with each document. For each document $d_i \in \mathcal{D}$, we prompt an LLM to generate a set of latent queries Q_i as follows:

$$Q_i = \mathrm{LLM}(d_i, \mathcal{P}), \quad \forall d_i \in \mathcal{D} \tag{1}$$

where $\mathcal{P}$ is a carefully designed prompt that encourages the generation of specific, informative, and diverse queries. To promote broader semantic coverage and discourage redundancy, we apply a positive presence penalty during generation.

To ensure query diversity, we apply a post-processing step that removes redundant queries based on embedding similarity. Let $R(\cdot)$ denote the bi-encoder used to embed queries into dense vectors. We construct a filtered set $\widetilde{Q}_i$ by sequentially adding queries from Q_i whose cosine similarity with all previously selected queries remains below a fixed threshold θ:

$$\widetilde{Q}_i = \{q \in Q_i \mid \forall q' \in \widetilde{Q}_i, \mathrm{sim}\big(R(q), R(q')\big) < \theta\} \tag{2}$$

where sim $(\cdot,\cdot)$ denotes cosine similarity and $\theta \in [0,1)$ is a predefined similarity threshold. We aggregate all filtered latent queries across the corpus to form the full latent query set:

$$\mathcal{Q} = \bigcup_{i=1}^{|\mathcal{D}|} \widetilde{Q}_i \tag{3}$$

Finally, all queries in $\mathcal{Q}$ and documents in $\mathcal{D}$ are encoded using R, and stored in separate vector databases for retrieval:

$$\mathcal{V}_q = R(\mathcal{Q}), \quad \mathcal{V}_d = R(\mathcal{D}) \tag{4}$$

3.2 Contrastive Alignment with Synthetic Queries

After obtaining a set of diverse latent queries for each document (Section 3.1), we aim to construct effective training triples (d_i, q^+, q^-) for contrastive or ranking-based learning. Here, $q^+ \in \widetilde{Q}_i$ denotes a relevant (positive) query for document d_i, while q^- is a negative query sampled from other documents' query sets, i.e., $\mathcal{Q} \setminus \widetilde{Q}_i$.

Naively sampling negatives at random often results in queries that are either trivially dissimilar or semantically ambiguous (false negatives), both of which degrade training quality. To address this, we propose a similarity-constrained negative query sampling strategy that adaptively selects informative negatives based on the semantic similarity between queries and the target document.

Concretely, for each document d_i, we compute the cosine similarities between it and its associated positive queries $\widetilde{Q}_i$, obtaining a distribution of positive similarities. Let s_{min}, s_{max} and $\bar{s}$ denote the minimum, maximum, and mean similarity values in this set. We then filter candidate negatives $q^- \in \mathcal{Q} \setminus \widetilde{Q}_i$, retaining only those whose similarity to d_i falls within the positive similarity interval $[s_{min}, s_{max}]$ ensures that the selected negatives are semantically close enough to be challenging, but still distinct from the document's true queries.

To further differentiate between candidates within this interval, we assign each negative a weight using a Gaussian kernel centered at $\bar{s}$, with a sharpness parameter α:

$$w_{q^-} = \exp\left(-\alpha \cdot \left(\mathrm{sim}\big(R(q^-), R(d_i)\big) - \bar{s}\right)^2\right) \tag{5}$$

Intuitively, this favors negatives whose similarity to the document lies near the "semantic center" of its positive queries. All candidates outside the defined range receive zero weight and are excluded.

The final negative set $\mathcal{N}_i$ is constructed by selecting up to k negatives from the weighted pool. If fewer than k candidates remain after filtering, we include all. Otherwise, we perform weighted sampling based on the normalized scores. Each query $q^- \in \mathcal{N}_i$ is used in constructing a training triple (d_i, q^+, q^-) together with a randomly sampled $q^+ \in \widetilde{Q}_i$.

Let $R(\cdot)$ be the shared bi-encoder used to embed both documents and queries into a common vector space R^d. During training, we apply Noisy Embedding Fine-Tuning (NEFTune) [15] to enhance generalization and training stability. Specifically, we inject uniform noise into the input embeddings before passing them through the encoder.

For any input $x \in \{d_i, q^+, q^-\}$, let $E(x) \in R^d$ denote its token embedding matrix of shape $L \times d$, where L is the sequence length and d is the embedding dimension. We define a noise matrix $\epsilon \in R^{L \times d}$, sampled element-wise from a uniform distribution:

$$\epsilon_{ij} \sim \mathcal{U}\left(-\frac{\beta}{\sqrt{L \cdot d}}, \frac{\beta}{\sqrt{L \cdot d}}\right) \tag{6}$$

where $\beta > 0$ is a hyperparameter controlling the noise scale. The perturbed embedding is given by $\tilde{E}(x) = E(x) + \epsilon$. The encoder then operates on the noisy embeddings by $z_x = R\left(\tilde{E}(x)\right)$.

We adopt an InfoNCE-style contrastive loss to train the encoder. Given a triple (d_i, q^+, q^-), the loss is defined as:

$$\mathcal{L}_i = -\log \frac{\exp\left(\text{sim}\left(z_{d_i}, z_{q^+}\right)/\tau\right)}{\exp\left(\text{sim}\left(z_{d_i}, z_{q^+}\right)/\tau\right) + \sum_{q^- \in \mathcal{N}_i} \exp\left(\text{sim}\left(z_{d_i}, z_{q^-}\right)/\tau\right)} \tag{7}$$

where $\text{sim}(\cdot,\cdot)$denotes cosine similarity and $\tau > 0$ is a temperature parameter. During inference, NEFTune noise is disabled and the encoder uses the clean embeddings:

$$z_x = R\big(E(x)\big), \quad \text{for all } x \in \mathcal{D} \cup \mathcal{Q}. \tag{8}$$

3.3 Semantic Bridging via Proxy Queries

To mitigate the semantic mismatch between short user queries and long, domain-specific documents, we propose a dual-path retrieval strategy that uses pre-generated proxy queries as semantic intermediaries. At inference time, for each user query q_u, we first retrieve a set of top-ranked proxy queries $\mathcal{Q}_{match}$ from the proxy query index $\mathcal{V}_q$, based on their dense similarity to the query embedding $R(q_u)$. These proxy queries are pre-associated with source documents, and their retrieval results are used to form a proxy-based candidate set:

$$\mathcal{D}_{\text{proxy}} = \{d \in \mathcal{D} \mid d \text{ is linked to } q,\ q \in \mathcal{Q}_{\text{match}}\} \tag{9}$$

In parallel, we retrieve the top-k documents directly from the document index $\mathcal{V}_d$, yielding another candidate set $\mathcal{D}_{direct}$.
Rather than merging or reranking the union of these sets, we employ a comparison-based path selection mechanism. Specifically, we compare the top k_{compare} documents from both $\mathcal{D}_{proxy}$ and $\mathcal{D}_{direct}$ using a cross-encoder reranker. For each rank position $i \in \{1, \ldots, k_{\text{compare}}\}$, we compute the cross-encoder relevance score:

$$s_d = \text{CrossEnc}(q_u, d), \quad \forall d \in \mathcal{D}_{proxy}^{[:k_{\text{compare}}]} \cup \mathcal{D}_{direct}^{[:k_{\text{compare}}]} \tag{10}$$

For each pair of documents at rank i, we retain the one with the higher relevance score and record which retrieval path it belongs to. After completing all comparisons, we determine the dominant retrieval path by majority count. If proxy-based candidates are preferred in at least half of the comparisons, we select $\mathcal{D}_{proxy}$ as the final output; otherwise, $\mathcal{D}_{direct}$ is used. All remaining top-ranked documents beyond k_{compare} are then drawn from the chosen path without further reranking.

4 Experiments

We conduct a two-stage experimental study to evaluate the effectiveness of LaQuA. The first stage examines its retrieval performance on public domain-specific benchmarks, focusing on how well the model aligns user queries with relevant documents.

In the second stage, we incorporate LaQuA into a RAG pipeline to assess its impact on answer generation. For this purpose, we construct a proprietary dataset based on course materials provided by a vocational training institution, covering manufacturing-related domains such as mechanical systems, hydraulics, and mold design. This setting allows us to evaluate LaQuA in a more realistic, domain-specific downstream task.

4.1 Document Retrieval Evaluation

Datasets and Evaluation Metrics. We conducted retrieval experiments on two datasets, FiQA-2018 [29] and SciFact [39], from the BEIR (Benchmarking Information Retrieval) benchmark [18], focusing on the financial and scientific domains. BEIR is a heterogeneous benchmark dataset designed to systematically evaluate the generalization ability of information retrieval models in zero-shot settings. It provides a unified framework for evaluating and comparing different retrieval models.

The FiQA-2018 dataset was developed for sentiment analysis and question-answering systems in the financial domain. It contains structured and unstructured English text sourced from various financial data sources. On the other hand, SciFact consists of scientific claims primarily in the biomedical domain, where each claim is paired with evidence-containing abstracts along with labels and rationales. Each claim is linked to relevant scientific document abstracts that have been manually annotated.

Table 1 summarizes key statistics of the datasets, including the number of queries, the number of documents in the corpus, and the average lengths of queries and documents. Since our proposed method operates in a zero-shot setting without requiring training or fine-tuning, we only used the test set for evaluation, without utilizing the training or validation sets.

Table 1. Summary of the datasets used in the experiments, including the number of queries, the size of the corpus, and the average lengths of queries and documents.

Dataset	Queries	Corpus Size	Avg. Query Length	Avg. Document Length
FiQA-2018	648	57,638	10.77	132.32
SciFact	300	5,183	12.37	213.63

We evaluate retrieval performance using several standard metrics. The primary metric is normalized Discounted Cumulative Gain at rank 10 (nDCG@10), which reflects both the relevance and rank position of retrieved documents. To provide a more comprehensive assessment, we also report Mean Average Precision at rank k (MAP@k), Recall@k, and Precision@k. MAP@k measures the average ranking quality across queries, Recall@k quantifies coverage of relevant documents, and Precision@k assesses the relevance density among top-ranked results.

Experimental Setup. We use ChromaDB[1] as the vector database to store and retrieve dense representations of documents, user queries, and proxy queries. For both

[1] https://trychroma.com/

datasets, we concatenate the title and body of each document into a single text sequence prior to encoding.

The proxy query generation and selection are conducted using the public API of DeepSeek LLM. To promote semantic diversity and reduce redundancy, we apply a cosine similarity threshold of $\theta = 0.9$ during post-generation filtering. All proxy queries are pre-encoded using the same retriever $R(\cdot)$ and stored alongside the document embeddings for fast lookup during inference.

For dense retrieval, we adopt bge-base-en-v1.5 as the shared bi-encoder backbone $R(\cdot)$, and employ bge-reranker-v2-m3 [4,24] as the cross-encoder for contextual relevance estimation. During training, we fine-tune the retriever using a contrastive InfoNCE objective. Positive query-document pairs are constructed from proxy queries and their corresponding source documents. Negative queries are sampled using a similarity-constrained Gaussian weighting strategy with a sharpness parameter $\alpha = 20$, and a total of $k = 8$ negatives per positive pair. We apply NEFTune with a noise scale $\beta = 15$ to improve generalization, and normalize all output embeddings to unit length. The temperature parameter for contrastive loss is set to $\tau = 0.02$.

Training is conducted with the AdamW optimizer using a learning rate of 1×10^{-5}, a linear learning rate scheduler with a warm-up ratio of 0.1, and mixed-precision fp16 training. We train for 10 epochs with a batch size of 128. All experiments are run on a single NVIDIA A40 GPU with 48GB VRAM.

At inference time, we adopt the semantic bridging strategy described in Section 3.3. For each query, we independently retrieve top-k candidates from both the proxy-based and direct retrieval paths. We then apply a lightweight decision mechanism by reranking the top $k_{\text{compare}} = 5$ document pairs (one from each path per rank) using the cross-encoder. The path that receives more favorable pairwise scores is selected as the source of the final output. This selection strategy allows dynamic path choice without the overhead of reranking the entire candidate pool.

To monitor training dynamics, we also track the InfoNCE loss over training epochs. Fig. 3 presents the loss curves on FiQA-2018 and SciFact. The models exhibit stable convergence behavior, suggesting effective optimization under our proposed sampling and augmentation strategies.

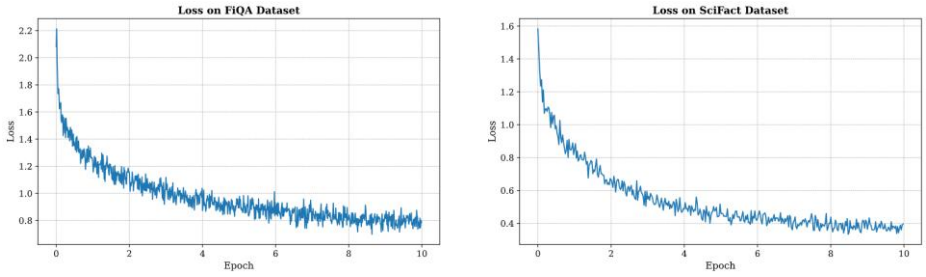

Fig. 3. Training loss curves for LaQuA's dense retriever on FiQA and SciFact.

Main Results. We compare LaQuA with several representative baselines to evaluate its retrieval effectiveness. The primary baseline is DenseBaseline, a standard dense

retriever trained directly on original query-document pairs without any latent query supervision. We also include Offline Pseudo-Relevance Feedback (OPRF) [41], a two-stage retrieval approach based on synthetic queries. We evaluate two variants: OPRF-B, which uses BM25 for retrieving pseudo-queries and a dense encoder for retrieving documents; and OPRF-D, which uses dense encoders for both stages to better align with our method. Additionally, we consider QOQA (Query Optimization using Query expAnsion) [21], a query rewriting method using large language models. Since QOQA only reports nDCG@10, we include that metric for completeness. "--" indicates metrics not reported in the original paper.

Table 2 summarizes the retrieval results on FiQA and SciFact. The full model, LaQuA (full), achieves the highest performance across all evaluation metrics and datasets, reflecting strong alignment between queries and documents in domain-specific contexts.

We also perform ablation studies to quantify the contributions of LaQuA's key components. LaQuA w/o SB removes the Semantic Bridging module described in Section 3.3, reverting to direct dense retrieval at inference time. LaQuA w/o CA removes the Contrastive Alignment strategy outlined in Section 3.2, and uses the pretrained encoder without domain-specific fine-tuning. Both ablations lead to notable drops in performance, confirming the complementary value of semantic bridging during inference and contrastive alignment during training.

Table 2. Retrieval performance (%) on FiQA and SciFact.

Method	FiQA				SciFact			
	nDCG @10	MAP@ 10	P@10	R@100	nDCG @10	MAP@ 10	P@10	R@100
Dense-Baseline	38.4	30.6	10.9	71.9	73.3	68.9	9.7	96.7
OPRF-B	17.8	13.0	6.1	47.8	65.8	62.3	8.5	91.2
OPRF-D	22.9	16.9	7.8	59.7	71.6	66.4	9.9	94.3
QOQA	40.6	--	--	--	75.4	--	--	--
LaQuA (full)	42.1	33.2	11.9	73.6	76.8	72.3	10.0	97.1
LaQuA w/o SB	39.2	31.5	11.1	71.5	74.5	70.1	9.7	93.5
LaQuA w/o CA	37.7	30.2	10.8	68.1	73.7	69.4	9.7	96.3

Case Study: Improved Retrieval Focus via Proxy Queries. To complement the quantitative results, Fig. 4 presents a representative case comparing direct dense retrieval with LaQuA's proxy-query-based approach. The user query asks about the typical lifespan of a CNC machine tool. As shown on the left, the direct retriever retrieves a long passage that embeds key information within a broader, loosely related discussion, requiring additional effort from downstream components to extract the relevant content.

In contrast, LaQuA first retrieves latent proxy queries that are semantically aligned with the user's intent—such as "How to predict and control tool usage cycles in CNC machining?" and then maps them to their associated documents. This intermediate step effectively narrows the retrieval scope to more focused and contextually appropriate segments.

This example illustrates LaQuA's semantic bridging capability: by leveraging proxy queries as intent-reflective intermediaries, the retrieval process becomes more targeted, reducing noise and enhancing precision in domain-specific information access.

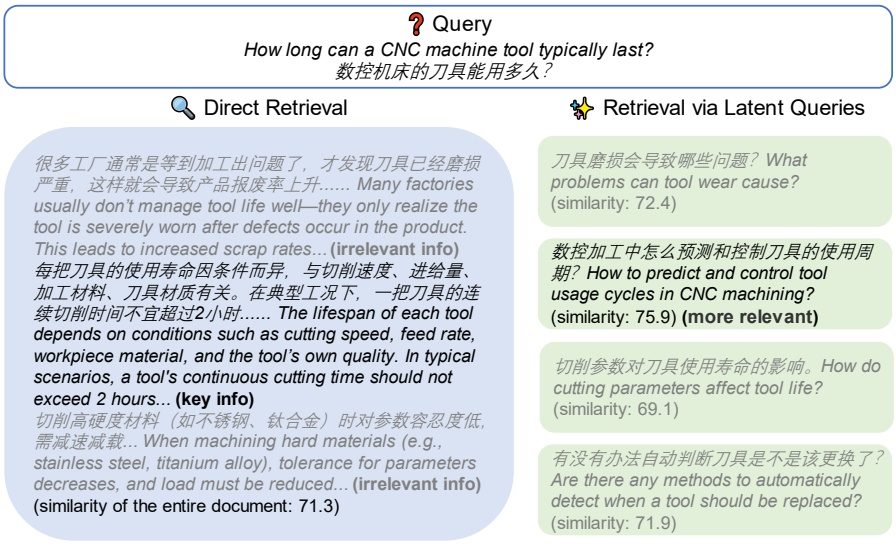

Fig. 4. Example comparison between direct retrieval and LaQuA's proxy-query approach.

4.2 Generation Quality in RAG Pipelines

Evaluation Setup. To assess the real-world utility of LaQuA in downstream generation tasks, we construct a domain-specific evaluation benchmark in Chinese. The dataset is built from instructional resources in document formats such as Word, PDF, and PowerPoint, covering topics in technical domains including mechanics, hydraulics, and mold design. We extract the raw text content, perform data cleaning, and segment it into approximately 300-character passages. The final corpus consists of over 1,200,000 tokens, forming a dense vector knowledge base indexed via a FAISS-based retriever.

We design 25 open-ended, domain-relevant queries that reflect typical information-seeking scenarios encountered in practice. Each query is paired with a human-written reference answer, authored by domain experts. These answers serve as ground truth for evaluating both retrieval and generation quality.

Retrieval Evaluation with LLM-as-a-Judge. To compare retrieval effectiveness, we adopt an LLM-as-a-judge protocol. Given a query, we retrieve top-ranked passages using different retrieval methods and use an LLM to evaluate which set of results is more relevant. Specifically, we compare LaQuA with two baselines: a standard dense

retriever trained on original query-document pairs (DenseBaseline), and OPRF (Dense+Dense), a two-stage pseudo-query method based on dense encoders.
We use the DeepSeek API as the judge model. For each query, the top results from two retrieval methods are presented to the judge in random order. To mitigate positional bias, we conduct two rounds of evaluation by switching the order of the presented results. If the model outputs consistent preferences across both rounds, we record the comparison as a win/loss. If the preferences contradict, the outcome is marked as a tie. Results are aggregated across all 25 queries to determine which retrieval method yields higher relevance according to the LLM. Table 3 reports the win/loss/tie outcomes for each comparison.

Table 3. Pairwise retrieval comparison using LLM-as-a-Judge (DeepSeek) across 25 queries.

Comparison	Wins	Ties	Losses
LaQuA vs. DenseBaseline	18	5	2
LaQuA vs. OPRF (Dense+Dense)	16	6	3

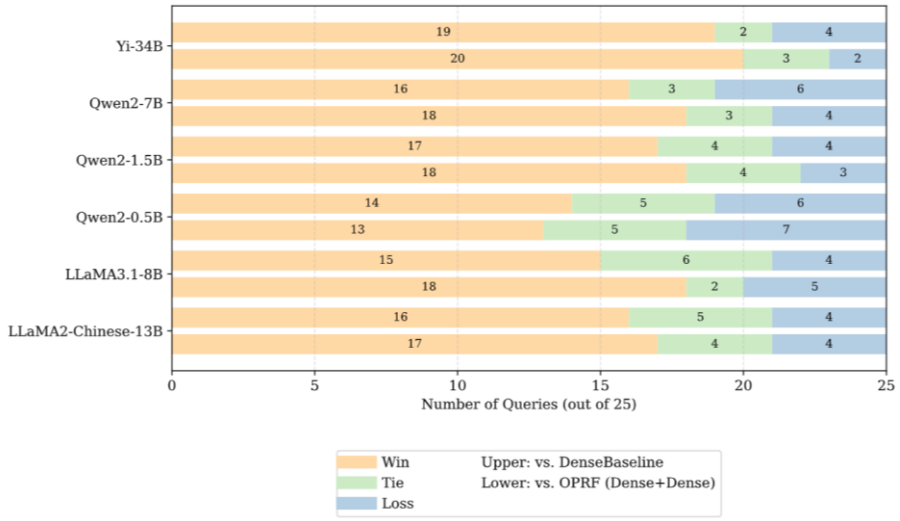

Fig. 5. LLM-as-a-Judge evaluation of answer generation quality in the RAG setting.

Generation Evaluation via RAG. To assess whether improvements in retrieval quality lead to better downstream generation, we evaluate LaQuA in a RAG pipeline. Each query is first processed by a retriever to obtain the top 5 most relevant passages, which are then concatenated with the query and provided as input to a language model for answer generation. We compare LaQuA against two retrieval baselines: DenseBaseline, a standard bi-encoder model trained on original query-document pairs, and OPRF (Dense+Dense), a two-stage pseudo-query retrieval method using dense encoders at both stages.

We use multiple open-source large language models deployed locally via Ollama[2], including LLaMA2-Chinese-13B [3], LLaMA3.1-8B [4], Qwen2 [5]-0.5B, Qwen2-1.5B, Qwen2-7B, and Yi-34B[6]. This diversity allows us to evaluate whether retrieval quality improvements generalize across model sizes and architectures.

To evaluate the quality of generated responses, we again adopt the LLM-as-a-judge framework. The DeepSeek API serves as the evaluation model. For each query, the judge receives the original query, a reference answer (authored by experts), and two generated responses produced under different retrieval strategies. The judge is prompted to select the better response based on factual correctness and content coverage. As in the retrieval comparison, evaluations are repeated with reversed order of presentation, and results are labeled as win, loss, or tie. Fig. 5 presents a comparative analysis of answer quality across several language models, using an LLM-as-a-Judge protocol. The figure reports the number of queries for which LaQuA's responses were evaluated as preferred, comparable, or inferior to those generated using two baseline retrieval strategies.

5 Conclusion

This paper presents LaQuA, a unified framework for latent query-driven dense retrieval in domain-specific contexts. By treating latent queries not merely as data augmentation but as structural bridges between user intent and document semantics, LaQuA systematically enhances both representation learning and inference-time alignment.

Empirical evaluations show LaQuA surpasses dense retrieval and pseudo-query baselines on public benchmarks and real-world RAG (English/Chinese, low-resource). Its semantic bridging offers a scalable alternative to full reranking, beneficial in constrained settings. Importantly, LaQuA's stronger retrieval alignment leads to downstream gains in generation accuracy, highlighting the value of latent query modeling as a principled component for RAG systems in practical LLM pipelines.

Future directions include generalizing proxy selection to multi-hop settings, leveraging instruction-tuned LLMs for query generation, and integrating domain priors or user intent modeling into the latent query space.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)

[2] https://ollama.com/
[3] https://huggingface.co/FlagAlpha/Llama2-Chinese-13b-Chat
[4] https://ai.meta.com/blog/meta-llama-3-1/
[5] https://huggingface.co/Qwen
[6] https://huggingface.co/01-ai/Yi-34B

2. Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al.: Qwen technical report. arXiv preprint arXiv:2309.16609 (2023)
3. Bang, Y., Cahyawijaya, S., Lee, N., Dai, W., Su, D., Wilie, B., Lovenia, H., Ji, Z., Yu, T., Chung, W., et al.: A multitask, multilingual, multimodal evaluation of chat-gpt on reasoning, hallucination, and interactivity. arXiv preprint arXiv:2302.04023 (2023)
4. Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., Liu, Z.: Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. arXiv preprint arXiv:2402.03216 (2024)
5. Clinchant, S., Gaussier, E.: A theoretical analysis of pseudo-relevance feedback models. In: Proceedings of the 2013 Conference on the Theory of Information Retrieval. pp. 6–13 (2013)
6. DeepSeek-AI: Deepseek llm: Scaling open-source language models with longter-mism. arXiv preprint arXiv:2401.02954 (2024), https://github.com/deepseek-ai/DeepSeek-LLM
7. Devlin, J.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
8. Engelmann, B., Breuer, T., Friese, J.I., Schaer, P., Fuhr, N.: Context-driven interactive query simulations based on generative large language models. In: European Conference on Information Retrieval. pp. 173–188. Springer (2024)
9. Fan, W., Ding, Y., Ning, L., Wang, S., Li, H., Yin, D., Chua, T.S., Li, Q.: A survey on rag meeting llms: Towards retrieval-augmented large language models. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 6491–6501 (2024)
10. Gao, T., Yao, X., Chen, D.: Simcse: Simple contrastive learning of sentence embeddings. arXiv preprint arXiv:2104.08821 (2021)
11. Guo, J., Fan, Y., Ai, Q., Croft, W.B.: A deep relevance matching model for ad-hoc retrieval. In: Proceedings of the 25th ACM international on conference on information and knowledge management. pp. 55–64 (2016)
12. Hambarde, K.A., Proenca, H.: Information retrieval: recent advances and beyond. IEEE Access (2023)
13. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for nlp. In: International conference on machine learning. pp. 2790–2799. PMLR (2019)
14. Huang, Y., Huang, J.: A survey on retrieval-augmented text generation for large language models. arXiv preprint arXiv:2404.10981 (2024)
15. Jain, N., Chiang, P.y., Wen, Y., Kirchenbauer, J., Chu, H.M., Somepalli, G., Bartoldson, B.R., Kailkhura, B., Schwarzschild, A., Saha, A., et al.: Neftune: Noisy embeddings improve instruction finetuning. arXiv preprint arXiv:2310.05914 (2023)
16. Jha, R., Wang, B., Günther, M., Mastrapas, G., Sturua, S., Mohr, I., Koukounas, A., Akram, M.K., Wang, N., Xiao, H.: Jina-colbert-v2: A general-purpose multi-lingual late interaction retriever. arXiv preprint arXiv:2408.16672 (2024)
17. Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y.J., Madotto, A., Fung, P.: Survey of hallucination in natural language generation. ACM Computing Surveys 55(12), 1–38 (2023) Title Suppressed Due to Excessive Length 19
18. Kamalloo, E., Thakur, N., Lassance, C., Ma, X., Yang, J.H., Lin, J.: Resources for brewing beir: Reproducible reference models and an official leaderboard (2023)
19. Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.: Dense passage retrieval for open-domain question answering. arXiv preprint arXiv:2004.04906 (2020)
20. Khramtsova, E., Zhuang, S., Baktashmotlagh, M., Wang, X., Zuccon, G.: Selecting which dense retriever to use for zero-shot search. In: Proceedings of the Annual International ACM

SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region. pp. 223–233 (2023)
21. Koo, H., Kim, M., Hwang, S.J.: Optimizing query generation for enhanced document retrieval in rag. arXiv preprint arXiv:2407.12325 (2024)
22. Lee, C., Roy, R., Xu, M., Raiman, J., Shoeybi, M., Catanzaro, B., Ping, W.: Nv-embed: Improved techniques for training llms as generalist embedding models. arXiv preprint arXiv:2405.17428 (2024)
23. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., et al.: Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in Neural Information Processing Systems 33, 9459–9474 (2020)
24. Li, C., Liu, Z., Xiao, S., Shao, Y.: Making large language models a better foundation for dense retrieval (2023)
25. Li, H., Mourad, A., Zhuang, S., Koopman, B., Zuccon, G.: Pseudo relevance feedback with deep language models and dense retrievers: Successes and pitfalls. ACM Transactions on Information Systems 41(3), 1–40 (2023)
26. Liu, A., Feng, B., Xue, B., Wang, B., Wu, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., et al.: Deepseek-v3 technical report. arXiv preprint arXiv:2412.19437 (2024)
27. Liu, F., Liu, Y., Shi, L., Huang, H., Wang, R., Yang, Z., Zhang, L.: Exploring and evaluating hallucinations in llm-powered code generation. arXiv preprint arXiv:2404.00971 (2024)
28. Lv, Y., Zhai, C.: Positional relevance model for pseudo-relevance feedback. In: Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. pp. 579–586 (2010)
29. Maia, M., Handschuh, S., Freitas, A., Davis, B., McDermott, R., Zarrouk, M., Balahur, A.: Www'18 open challenge: financial opinion mining and question answering. In: Companion proceedings of the the web conference 2018. pp. 1941–1942 (2018)
30. Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., Gao, J.: Large language models: A survey. arXiv preprint arXiv:2402.06196 (2024)
31. Mohr, I., Krimmel, M., Sturua, S., Akram, M.K., Koukounas, A., Günther, M., Mastrapas, G., Ravishankar, V., Martínez, J.F., Wang, F., et al.: Multi-task contrastive learning for 8192-token bilingual text embeddings. arXiv preprint arXiv:2402.17016 (2024)
32. Ni, J., Abrego, G.H., Constant, N., Ma, J., Hall, K.B., Cer, D., Yang, Y.: Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. arXiv preprint arXiv:2108.08877 (2021)
33. Patil, R., Boit, S., Gudivada, V., Nandigam, J.: A survey of text representation and embedding techniques in nlp. IEEE Access 11, 36120–36146 (2023)
34. Pouramini, A., Faili, H.: Matching tasks to objectives: Fine-tuning and prompt-tuning strategies for encoder-decoder pre-trained language models. Applied Intelligence 54(20), 9783–9810 (2024)
35. Qiao, Y., Xiong, C., Liu, Z., Liu, Z.: Understanding the behaviors of bert in ranking. arXiv preprint arXiv:1904.07531 (2019)
36. Rasheed, I., Banka, H., Khan, H.M.: Pseudo-relevance feedback based query expansion using boosting algorithm. Artificial Intelligence Review 54(8), 6101–6124 (2021)
37. Rawte, V., Sheth, A., Das, A.: A survey of hallucination in large foundation models. arXiv preprint arXiv:2309.05922 (2023)
38. Shi, J., Guo, Q., Liao, Y., Liang, S.: Legalgpt: Legal chain of thought for the legal large language model multi-agent framework. In: International Conference on Intelligent Computing. pp. 25–37. Springer (2024)
39. Wadden, D., Lin, S., Lo, K., Wang, L.L., van Zuylen, M., Cohan, A., Hajishirzi, H.: Fact or fiction: Verifying scientific claims. arXiv preprint arXiv:2004.14974 (2020)

40. Wang, X., Macdonald, C., Tonellotto, N., Ounis, I.: Colbert-prf: Semantic pseudo-relevance feedback for dense passage and document retrieval. ACM Transactions on the Web 17(1), 1–39 (2023)
41. Wen, X., Chen, X., Chen, X., He, B., Sun, L.: Offline pseudo relevance feedback for efficient and effective single-pass dense retrieval. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2209–2214 (2023)
42. Xu, Y., Jones, G.J., Wang, B.: Query dependent pseudo-relevance feedback based on wikipedia. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. pp. 59–66 (2009)
43. Xu, Z., Jain, S., Kankanhalli, M.: Hallucination is inevitable: An innate limitation of large language models. arXiv preprint arXiv:2401.11817 (2024)
44. Yan, Y., Li, R., Wang, S., Zhang, F., Wu, W., Xu, W.: Consert: A contrastive framework for self-supervised sentence representation transfer. arXiv preprint arXiv:2105.11741 (2021)
45. Yang, J., Liu, Z., Li, C., Sun, G., Xie, X.: Longtriever: a pre-trained long text encoder for dense document retrieval. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 3655–3665 (2023)
46. Zhao, W.X., Liu, J., Ren, R., Wen, J.R.: Dense text retrieval based on pretrained language models: A survey. ACM Transactions on Information Systems 42(4), 1–60 (2024)