Dense Query Representations Alternative to Text Queries for Dense Conversational Search

Anonymous ACL submission

Abstract

Query rewriting in conversational search aims to transform context-dependent user utterances, often containing pronouns or ellipses, into well-formed queries suitable for retrieval. Existing approaches predominantly rely on generative language models trained with human-written queries, optimizing a language modeling objective that is inherently misaligned with the ranking objective of retrievers. As a result, improvements in query rewriting do not necessarily translate into retrieval performance gains, and feedback from retrievers cannot directly influence the rewriter. In this work, we address this objective mismatch by proposing a simple yet effective framework, DenseQR[1], that replaces text queries with dense representations. Instead of generating text, our model directly produces dense embeddings that are consumed by a dense retriever, enabling end-to-end optimization under the retrieval objective. This design eliminates the reliance on costly human-written rewrites and allows the representation model to be trained solely from retrieval supervision. Experiments on widely used conversational search benchmarks demonstrate that our approach consistently outperforms state-of-the-art generative query rewriting methods when paired with dense retrievers, achieving superior retrieval effectiveness while reducing training complexity.

1 Introduction

With the recent advances in large language models (LLMs), retrieval-augmented generation (RAG) has emerged as a compelling paradigm that integrates external knowledge sources into generative models. By grounding generation in retrieved evidence, RAG enhances factual correctness, mitigates hallucination, and improves generalization across knowledge-intensive tasks (Lewis et al., 2020). This framework has demonstrated strong performance in open-domain question answering, fact-driven dialogue, and real-world decision-support systems, positioning retrieval as a core component of next-generation conversational AI.

Despite substantial progress in ad-hoc retrieval for single-turn information retrieval (IR), extending these techniques to multi-turn conversational settings remains a significant challenge. Conversational queries are inherently context-dependent and often exhibit complex linguistic phenomena such as co-reference, ellipsis, pragmatic implicature, and gradual topic drift (Adlakha et al., 2022). These characteristics frequently obscure explicit lexical cues required by retrieval systems, resulting in underspecified queries that degrade retrieval effectiveness and, consequently, downstream generation quality (Gao et al., 2022; Anantha et al., 2021).

To mitigate this issue, prior work in conversational search has predominantly relied on query rewriting, which aims to transform a context-dependent user utterance into a fully specified, stand-alone query. Most existing approaches employ generative language models trained with human-authored rewrite annotations that provide canonical query formulations Wu et al. (2022); Mo et al. (2023); Lai et al. (2024). Although such methods can improve surface-level query clarity, they suffer from fundamental limitations. First, they introduce an inefficient conversion pipeline in which conversational context is encoded into hidden states, decoded into discrete tokens, and then re-embedded into dense vectors for retrieval, despite dense retrievers operating exclusively on continuous representations. This redundant transformation not only incurs additional computational overhead but also risks information loss. Second, the training objective of the rewriting model is inherently misaligned with that of the retriever: rewriting models are optimized using language

[1] https://anonymous.4open.science/r/denseqr-BB01/

modeling objectives, whereas retrievers are trained with ranking-based objectives. Consequently, improvements in rewrite fluency or contextual completeness do not necessarily translate into improved retrieval effectiveness.

This objective misalignment primarily arises from the non-differentiable interface between the rewriting model and the retriever. Because generative rewriting models output discrete token sequences, relevance signals from the retriever cannot be directly backpropagated through the generation process under standard training paradigms. As a result, the rewriting model receives no explicit feedback regarding which lexical or semantic choices improve document ranking, remaining largely agnostic to retrieval errors. Moreover, the reliance on human-authored rewrite annotations imposes substantial annotation costs, limiting scalability. Together, these drawbacks motivate a more principled framework that both removes the unnecessary conversion between dense and discrete representations and directly aligns conversational query modeling with retrieval objectives.

In this work, we hypothesize that dense representations are more suitable than textual rewrites for dense retrieval, as they not only match the retriever's input space but also enable direct optimization with retrieval-oriented objectives. Based on this insight, we propose a simple yet effective framework that replaces textual query rewrites with dense query embeddings. Specifically, we introduce Dense Query Representor (DenseQR), which encodes conversational questions directly into dense vectors that are consumed by a dense retriever, without decoding them into text. DenseQR is trained end-to-end using the retriever's ranking loss, allowing gradients to flow directly from retrieval supervision to the query representation model. This design explicitly aligns the objectives of conversational query modeling and retrieval, while eliminating the redundant conversion between dense vectors and discrete tokens.

Extensive experiments on widely used conversational search benchmarks demonstrate that the proposed approach consistently outperforms generative query rewriting methods in dense retrieval settings. Notably, DenseQR achieves state-of-the-art performance without introducing additional architectural complexity, highlighting the effectiveness of objective alignment and representation-level modeling for efficient conversational dense retrieval.

Our contributions are summarized as follows:

- We propose DenseQR, an encoder-based framework that represents conversational queries as dense vectors and enables direct optimization using retrieval objectives.
- The proposed method eliminates the need for human-authored query rewrites, significantly reducing annotation cost.
- Empirical results demonstrate that aligning query modeling and retrieval objectives leads to substantial improvements over prior generative query rewriting approaches.

2 Related Work

In early stage of conversational search, query rewriting method, enforcing generative language models to mimic human rewrite queries, were proposed (Anantha et al., 2021; Lin et al., 2020). As pioneering studies, they used inputs of concatenated conversational history and a current question and trained generative language models with language modeling loss by using the human written query as a supervision signal.

The generated queries quite plausibly addressed the problems mentioned above that can occur in conversational environments, but they were still lack of information in terms of search purpose. To address this problem, query expansion methods emerged that additionally input useful terms to short generated queries. The basic idea comes from utilizing important terms that are contained in conversational history (Voskarides et al., 2020). They trained a query resolution model individually scoring the importance of input terms and selected highly scored ones as search query terms assuming they are possibly helpful for retrieving relevant passages. On the other hand, some studies tried to expand queries with generated answers because the terms selected from the histories might cause the retrievers focus on the passages relevant to the past questions. The generated answers can be a reasonable alternative to get new terms strongly related to the current question although they did not appear in the past conversations. This approach has been adopted to several subsequent studies and has shown its worth (Lai et al., 2024; Mo et al., 2023; Yoon et al., 2024).

Although the previously described approaches were competitive, the biggest issue was that the

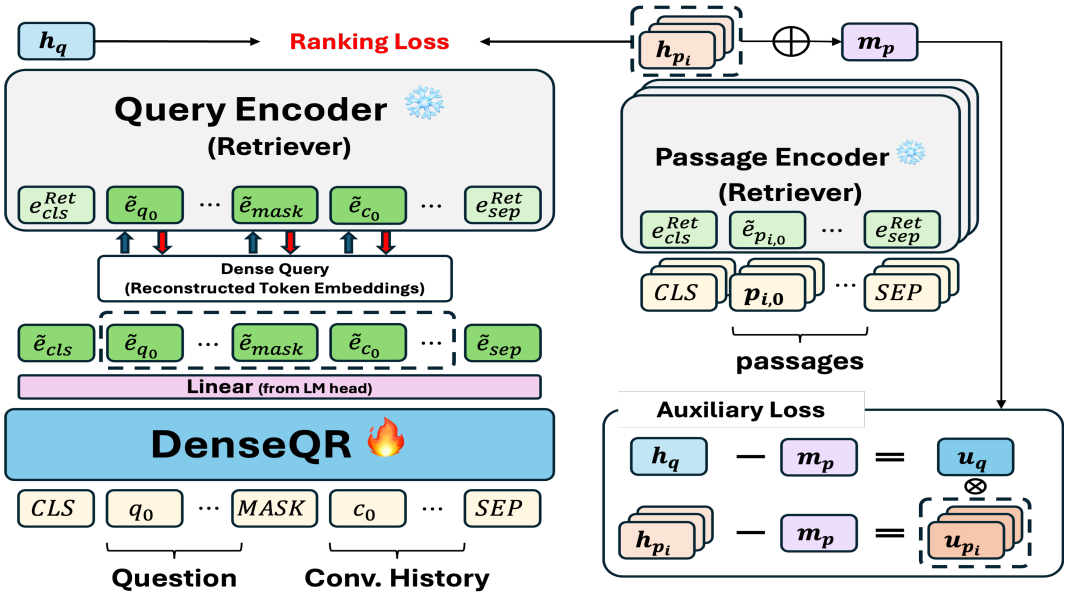

Figure 1: The overview of the proposed framework. The dense embeddings from the DenseQR goes through the retriever except for special tokens. It is important to note that the linear layer is from the lm-head of DenseQR which actually transform the hidden states into token embeddings in the backbone model .

generative models were not trained with retriever's signal which is the actual objective of the field. Since the generative models cannot directly use the retrieval signal as an objective, the preference optimization techniques were adopted to represent retriever's signal indirectly (Jang et al., 2024; Lai et al., 2024; Yoon et al., 2024). Built upon the recent advancement of large language models, they generated pseudo queries, sorted them by retrieval performance, and optimized the rewriter models to generate queries more preferred by the retriever. The optimization method significantly improved conversational search performances, but they utilized quite larger models for rewriter than the previous ones, causing question mark whether increased query latency is allowable or not. In addition, indirect application of retrieval's signal still remains as a critical bottleneck.

3 Methodology

3.1 Motivation

Dense retrievers encode tokenized input queries into high-dimensional vector spaces via token embeddings. Importantly, these models do not strictly require inputs to be expressed as natural language tokens, as retrieval is ultimately performed over dense vectors. This observation suggests that the standard token embedding layer can, in principle, be replaced with alternative dense representations, provided that they lie within a distribution compatible with the retriever's encoder.

In addition, dense retrievers are commonly trained using contrastive objectives that maximize the similarity between query and document representations, typically derived from the final hidden state of a designated pooling token (e.g., the [CLS] token). During this training process, higher layers of the encoder are primarily adapted to optimize retrieval-specific similarity structures, while lower layers tend to retain much of their pre-trained behavior. As a consequence, lower layers continue to capture general lexical and syntactic properties of the input sequence, whereas upper layers specialize in producing retrieval-oriented representations. This hierarchical specialization provides feasibility that dense representations can serve as inputs for dense retrievers as long as they capture lexical meanings.

In pre-trained language models (PLMs), the final hidden states of tokens can be viewed as reconstructed embeddings of the original input tokens. This property arises from parameter sharing between the input embedding layer and the output projection layer (i.e., the LM head), which encourages hidden states to preserve semantic information aligned with the vocabulary embedding space. As a result, these hidden representations encode rich contextual semantics while maintaining compatibility with downstream embedding-based operations.

Motivated by these observations, we propose an encoder-based query representation framework that maps conversational queries directly into dense vectors and enables end-to-end optimization using retrieval objectives. An overview of the proposed framework is shown in Figure 1. We deliberately exclude decoder-based generative models from our design, as their autoregressive generation incurs additional latency and produces discrete token sequences that are inherently misaligned with the continuous embedding distribution expected by dense retrievers. By operating entirely in the dense vector space, the framework facilitates direct gradient propagation from the retriever's ranking loss to the query representation model.

3.2 Dense Query Representations

DenseQR represents a conversational query directly as a sequence of dense vectors. Let $c_i = \{(q_k, a_k)\}_{k=0}^{i-1}$ denote the conversational history up to the i-th turn, where q_k and a_k represent the question and answer at turn k, respectively. The input to the representor is constructed by concatenating the current user question with its preceding conversational history.

To reflect the conversational structure where answers follow questions, we append a fixed number of `[MASK]` tokens immediately after the current question. These mask tokens serve two purposes: (1) they provide representational capacity for a potential answer, and (2) they facilitate information aggregation from the conversational history. The resulting input sequence, augmented with special tokens, is defined as

$$x_q = \texttt{[CLS]}\ q_i\ \texttt{[MASK]}^m\ c_i\ \texttt{[SEP]}, \qquad (1)$$

where m denotes the number of appended mask tokens. The tokenized input sequence is mapped to its corresponding token embeddings using the embedding layer of the representor, yielding

$$E_q^\theta = \texttt{Embedding}^\theta(x_q) \in \mathbb{R}^{|q| \times D}, \qquad (2)$$

where $|q|$ denotes the length of the tokenized input and D is the embedding dimensionality. θ denotes the representor. The embedding sequence is then processed by the DenseQR encoder, producing contextualized hidden representations

$$H_q^\theta = \texttt{Encoder}^\theta(E_q^\theta), \qquad (3)$$

where $H_q^\theta \in \mathbb{R}^{|q| \times D}$. These hidden states capture the semantic content of the conversational query while incorporating contextual dependencies across the sequence.

To project the hidden representations back into the token embedding space expected by the dense retriever, we apply a linear transformation followed by layer normalization and a non-linear activation function:

$$\tilde{E}_q^\theta = \sigma\left(\texttt{LayerNorm}\left(H_q^\theta \cdot W_D^\top\right)\right), \qquad (4)$$

where $W_D \in \mathbb{R}^{D \times D}$ is a trainable projection matrix and $\sigma(\cdot)$ denotes the non-linear activation function. The parameters of W_D are initialized using the linear layer of the LM head in DenseQR, as this layer is explicitly trained to map final hidden states back into the token embedding space.

We call the reconstructed embeddings $\tilde{E}_{q_{1:-1}}^\theta$ as **dense query** except for special tokens, an input for the dense retriever replacing the text query. Notably, we preserve the original embedding of the `[CLS]` token rather than replacing it with a reconstructed embedding. Since dense retrievers are typically trained to use the `[CLS]` representation as the primary vector for query–document similarity, dynamically modifying this token across inputs is expected to destabilize training. By retaining the original `[CLS]` embedding, we ensure compatibility with the retriever's learned similarity structure while allowing DenseQR to flexibly represent the remaining tokens.

3.3 Training Objective

The final input for the retriever are constructed by wrapping the dense query with the special tokens' embeddings of the dense retriever, ϕ:

$$E_q^\phi = [e_{cls}^\phi; \tilde{E}_{q_{1:-1}}^\theta; e_{sep}^\phi]. \qquad (5)$$

The retriever encodes the inputs and results in a dense representation corresponding to the last hidden state of `[CLS]` token. Similarly, a passage representation is obtained by the retriever with textual inputs:

$$h_q = \texttt{Encoder}^\phi(E_q^\phi)_{cls}, \qquad (6)$$

$$h_p = \phi(x_p)_{cls}. \qquad (7)$$

Finally, a similarity between the query and the passage is calculated by the inner product, denoted as f, of the two dense representations and is used for calculating contrastive loss. The rewriter is then optimized by the contrastive loss which is

directly backpropagated from the retriever through the reconstructed token embeddings:

$$\mathcal{L}_{\text{aux}} = -\log \frac{\exp^{f(h_q, h_{p^+})}}{\exp^{f(h_q, h_{p^+})} + \sum_{p_i \in \mathcal{N}} \exp^{f(h_q, h_{p_i})}}. \tag{8}$$

where p^+ and $\mathcal{N}$ indicate a positive passage and a set of negative passages, respectively. In addition to the typical contrastive loss based on query-passage representation pairs, various contrastive losses are adopted to elaborately rewrite queries. The specifics of the additional losses are described in the following paragraphs.

Auxiliary contrastive loss Although DenseQR encodes both the current question and its conversational history, incorporating long histories may introduce noise into the query representation, particularly as the conversation progresses. In such cases, the model may overemphasize shared or background information present across passages rather than focusing on turn-specific intent. To mitigate this issue, we introduce an auxiliary contrastive loss designed to suppress shared semantic components and emphasize discriminative query features.

Inspired by quantization- and clustering-based representation learning techniques Zeng et al., 2024; Babenko and Lempitsky, 2014; Chen et al., 2010, we compute a common passage representation by averaging the representations of both positive and negative passages within a training instance:

$$m_p = \frac{1}{|p^+ \cup \mathcal{N}|} \sum_{p_i \in p^+ \cup \mathcal{N}} h_{p_i}, \tag{9}$$

where $m_p \in \mathbb{R}^D$ captures shared semantic content across candidate passages. We then subtract this common component from both query and passage representations:

$$u_q = h_q - m_p, \tag{10}$$

$$u_{p_i} = h_{p_i} - m_p. \tag{11}$$

The resulting representations are expected to emphasize discriminative, turn-specific information while suppressing common background semantics. We apply a second contrastive loss using these adjusted representations:

$$\mathcal{L}_{\text{aux}} = -\log \frac{\exp^{f(u_q, u_{p^+})}}{\exp^{f(u_q, u_{p^+})} + \sum_{p_i \in \mathcal{N}} \exp^{f(u_q, u_{p_i})}}. \tag{12}$$

This auxiliary objective encourages DenseQR to produce query representations that are robust to conversational noise and better aligned with discriminative retrieval signals.

The final training objective for DenseQR is defined as the sum of the base and auxiliary contrastive losses:

$$\mathcal{L} = \mathcal{L}_{\text{base}} + \mathcal{L}_{\text{aux}}. \tag{13}$$

4 Experiments

Dataset For the fair comparison, we conducted experiments on QReCC (Anantha et al., 2021) and TopiOCQA (Adlakha et al., 2022) datasets following the prior studies. Questions in both datasets are paired with passages which are necessary for answers at each turn. QReCC dataset consists of 14K open-domain English conversations with 80K question-answer pairs and about 54M passage collections. Note that human rewrite queries are exclusively available for the QReCC dataset. TopiOCQA dataset includes 3.9K conversations with 51k question-answer pairs and about 25.7M passages. Compared to QReCC dataset, conversations of TopiOCQA are longer and introduce more complexities in query reformulation.

Evaluation Metrics We evaluated the retrieval performance of our model with four evaluation metrics as widely used in earlier studies: mean reciprocal rank (MRR), NDCG@3 Recall@10, Recall@100. The higher value indicates the better performance for all the metrics.

Implementation Details The retriever was initialized from ANCE (Xiong et al., 2021) of which backbone model is RoBERTa (Liu et al., 2019). Consequently, the rewriter also used RoBERTa initialization to meet the vocabulary space of the retriever. We implemented both models in PyTorch (Paszke et al., 2019) and Huggingface `Transformer` library (Wolf et al., 2020). More details about the training parameters are described in Appendix A.

4.1 Main Results

The specifics of adopted baselines are detailed in Appendix B. The retrieval performance of our proposed model, DenseQR, exhibits substantial improvements over a wide range of competitive query reformulation baselines, as summarized in Table 1. Across both QReCC and TopiOCQA, DenseQR consistently sets new state-of-the-art results on

Type	Method	QReCC				TopiOCQA			
		MRR	NDCG@3	R@10	R@100	MRR	NDCG@3	R@10	R@100
Dense (ANCE)	*Text-based Query*								
	Raw	7.3	6.0	11.5	17.2	3.8	3.8	7.4	13.4
	Human rewrite	38.5	36.6	60.1	79.3	-	-	-	-
	Finetuning	45.4	44.2	64.8	79.7	29.1	28.4	49.2	70.7
	GPT2QR	33.9	30.9	53.1	72.9	12.6	12.0	22.0	33.1
	QuReTec	35.0	32.6	55.0	72.9	11.2	10.2	20.2	34.4
	T5QR	34.5	31.8	53.1	72.8	23.0	22.2	37.6	54.4
	ConQRR	41.8	-	65.1	84.7	-	-	-	-
	ConvGQR	42.0	39.1	63.5	81.8	25.6	24.3	41.8	58.8
	EDIRCS	42.1	-	65.5	**85.3**	-	-	-	-
	IterCQR	42.9	40.2	65.5	84.1	26.3	25.1	42.6	62.0
	LLM-Aided	43.5	41.3	65.6	82.3	-	-	-	-
	LLM4CS	36.8	34.0	56.1	74.2	30.3	29.0	49.9	67.7
	RETPO	44.0	41.1	66.7	84.6	30.0	28.9	49.6	68.7
	AdaCQR	45.1	42.4	66.3	83.4	32.8	31.5	**54.6**	73.0
	Embedding-based Query								
	DenseQR (Ours)	**47.0**	**45.8**	**68.4**	82.4	**33.0**	**32.8**	54.5	**74.4**

Table 1: Retrieval results on dense (ANCE) retriever as our model is not available to sparse retriever. The higher value indicates the better performance in all metrics. The best results are **bolded**, and the second-best results are underlined.

most evaluation metrics. In particular, DenseQR surpasses the previous bestperforming methods by +3.4%p on QReCC and +1.3%p on TopiOCQA in terms of NDCG@3, highlighting its superior ability to refine conversational queries into forms that maximize retrieval quality.

Prior approaches to conversational query rewriting largely fall into three categories. (1) Iterative-update methods, such as IterCQR, refine the query step-by-step by repeatedly interacting with the retriever, gradually improving retrieval scores through multiple reformulation rounds. (2) Retriever-guided optimization methods, including RETPO and AdaCQR, indirectly optimize the rewriter by sampling multiple candidate rewrites and training the model to generate reformulations that the retriever prefers. (3) LLM-based prompting methods, such as LLM-Aided and LLM4CS, leverage large language models to generate high-quality reformulations through instruction tuning or incontext prompting, often yielding strong performance with minimal finetuning.

Despite the sophistication of these recent techniques, DenseQR demonstrates that directly optimizing the dense search query in the embedding space—jointly with the underlying retriever—enables more effective retrieval-oriented reformulation. Unlike conventional rewriters that generate textual rewrites, DenseQR produces latent reformulations that need not correspond to human-readable text. This allows the model to exploit the geometric properties of the retriever' s representation space, producing reformulations that are more directly aligned with the retriever' s scoring function. As a result, DenseQR achieves the best performance in nearly all metrics. These results demonstrate that optimizing the reformulation process inside the retriever' s embedding space offers a more effective and scalable solution for conversational search than relying solely on textual rewriting techniques.

Backbone	Retriever	Datasets	
		QReCC	TopiOCQA
RoBERTa	ANCE	45.8	32.8
ANCE	ANCE	23.7	25.3
BERT	SimLM	39.8	23.8
SimLM	SimLM	36.6	13.1

Table 2: Retrieval performances according to initialized backbone.

4.2 Further Experiments

The Role of DenseQR Table 2 highlights an important observation regarding the role of DenseQR in terms of its representation ability. To demonstrate that DenseQR is not just extending layers of the retrievers, we experimented with stacking the same retriever directly beneath the dense retriever. However, this approach results in a sub-

Backbone	Retriever	Datasets	
		QReCC	TopiOCQA
RoBERTa	ANCE	45.8	32.8
BERT	SimLM	39.8	23.8
Qwen3	Qwen3-Embedding-0.6B	43.8	38.8

Table 3: Retrieval performances depending on the dense retrievers. The backbone model os DenseQR is also aligned with the retriever's backbone model.

Ablation	QReCC		TopiOCQA	
	N@3	R@10	N@3	R@10
DenseQR	**45.8**	**68.4**	**32.8**	**54.5**
(1) w/o mask ext.	45.1*	67.7*	31.8	52.7*
(2) w/o aux. loss	44.5*	66.9*	30.4*	51.6*

Table 4: Performances for the ablation study. N@3 and R@10 indicate NDCG@3 and Recall@10, respectively. The results marked with * are statistical significance (p<0.05) with repect to DenseQR.

stantial performance drop compared to DenseQR. For instance, the ANCE-stacked variant achieves only 23.7 NDCG@3 on QReCC and 25.3 on TopiOCQA, far below the corresponding DenseQR results.

This behavior underscores a key insight: The lower layers of dense retrievers remain stable during retriever training and become highly specialized for processing token-level representations. Simply stacking identical encoders disrupts this specialization and leads to degraded retrieval performance. DenseQR avoids this issue by reconstructing embeddings rather than forwarding the final hidden states of a pretrained encoder. This design preserves compatibility with the retriever' s representational structure while still enabling powerful reformulation through latent embedding optimization. The large performance gap between DenseQR and stacked variants confirms that reconstructed embeddings, rather than raw encoder hidden states, are essential for effective layer extension in dense retrieval.

Compatibility of the Framework Based on Table 3, we observe that the proposed framework, representing conversational queries directly as dense vectors, exhibits strong compatibility across a diverse set of dense retrievers and backbone architectures. Specifically, consistent retrieval effectiveness is achieved when pairing the framework with heterogeneous retrievers such as ANCE, SimLM, and Qwen3-Embedding, each built upon different backbone models. This indicates that the proposed approach does not rely on retriever-specific inductive biases or architectural assumptions, but instead produces dense representations that are broadly aligned with the embedding spaces learned by various retrievers. Moreover, aligning the backbone of DenseQR with that of the target retriever further ensures representational compatibility, enabling effective retrieval without retriever-specific adaptation. These results demonstrate that the proposed dense-vector-based conversational query formulation is a flexible and general mechanism that can be seamlessly integrated with existing dense retrieval systems.

4.3 Analysis

Ablation Study To evaluate the contribution of the inserted mask tokens and the added quantized loss, we conducted an ablation study by removing each component individually. The results, presented in Table 4, show that both components positively contribute to the final performance of DenseQR.

To better understand what information the mask tokens capture, we further analyzed the queries that exhibited performance gains when mask tokens were included. On one hand, when the query appears at the beginning of a conversation and no prior context exists, the mask tokens tend to represent information similar to the corresponding answer. On the other hand, when the conversation contains substantial context, the mask token encode tokens that appear to be semantically important within the dialogue-information that the retriever is likely to rely on when identifying relevant passages. This indicates that mask tokens function as answer placeholders in short-context scenarios and as context-sensitive summary cues in longer conversation. The examples of the cases are reported in Appendix C.

Regarding the auxiliary loss, we did not observe a clear pattern tied to its inclusion or exclusion in terms of query features or context lengths. Nevertheless, the improved queries gain, on average, three higher ranking positions, while the degraded queries lose only one position on average. This asymmetry indicates that the quantized loss still provides a net positive effect on retrieval quality and contributes meaningfully to the overall performance of DenseQR.

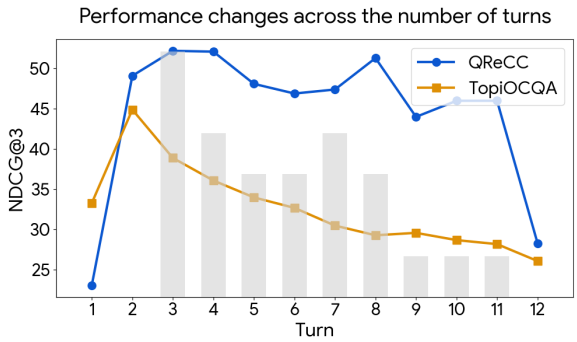

Figure 2: Illustration of the performance change across the number of turns. The gray bar indicates the number of samples that topic shift occurs in TopiOCQA dataset.

Model	Latency (ms/sample)		
	Rewriting	Encoding	Total
T5QR	1,040	4	1,044
DenseQR	350 (×2.97)	5	355 (×2.94)

Table 5: Query rewriting latency between T5QR and DenseQR. The T5QR is set to generate 32 tokens and DenseQR outputs representations of all input sequence including conversational histories.

Effect of Topic shift The observed performance trends across the two benchmark datasets, QReCC and TopiOCQA, strongly suggest that the presence of topic shifts is a critical factor influencing the robustness of conversational retrieval systems, particularly those relying on a full conversational history.

The TopiOCQA dataset, which is characterized by shifts in the underlying topic of conversation, exhibits a pattern where performance degradation aligns closely with the frequency of these shifts. As visually presented in Figure 2, the most substantial decline in NDCG@3 performance is clearly observed at Turn 3. This specific turn corresponds precisely to the peak in the "Topic shift" bar distribution, indicating the point in the conversation flow where topics are most frequently altered. This correlation leads to the interpretation that the context from prior topics begins to act as noise, resulting in the observed drop from the high performance at Turn 2 to the significantly lower performance at Turn 3, and a subsequent gradual decline thereafter.

In contrast, the QReCC dataset, which primarily maintains a single topic throughout the conversation, demonstrates a relatively stable and high retrieval performance over a substantial portion of the conversational history (Turns 2 through 8). This resilience suggests that when the conversational history remains focused on a consistent topic, the accumulating context is cumulative and beneficial, providing increasingly richer information for the retriever rather than introducing noise. Therefore, the overall retrieval performance for QReCC is deemed well-maintained across the core, topic-consistent turns.

Comparison on Query Latency As DenseQR adopts an encoder-based model instead of a generative query rewriting model, it inherently removes the autoregressive decoding process, which has a positive impact on query latency—an important factor in practical retrieval systems. To quantify this, we measure the latency from feeding a query into the rewriter (or representor) to obtaining the rewritten (or represented) query, as reported in Table 5. T5QR is adopted for comparison model because of its simple reproduction, while most generative rewriting model employ T5 as their backbone.

Despite encoding the entire input sequence including the conversational history, DenseQR achieves substantially lower rewriting latency than T5QR. Specifically, DenseQR completes rewriting in 0.35 seconds, which is approximately 2.97× faster than T5QR. Although the dense query is longer than the textual query, the difference in encoding time is minimal and does not dominate the overall latency.

5 Conclusion

In this paper, we introduced DenseQR, a novel framework for conversational query reformulation that constructs query representations directly in the dense embedding space rather than generating textual queries. By eliminating the need for intermediate text generation, DenseQR fundamentally resolves the long-standing objective misalignment between query rewriters and dense retrievers, enabling end-to-end optimization under the retrieval objective. Our extensive experiments demonstrate that DenseQR not only achieves state-of-the-art retrieval performance across multiple conversational search benchmarks but also provides substantial query latency reductions, making it both more effective and more efficient than prior query rewriting methods.

Limitations

While the proposed DenseQR model has demonstrated superior efficacy in resolving contextual ambiguity and generating robust dense embeddings

for dense retrieval model, its most significant limitation is its current inapplicability to sparse retrievers, such as BM25. The retrieval performance often varies considerably across different retriever types and datasets, underscoring the necessity of a retrieval system that can be leveraged effectively regardless of the underlying index mechanism. The primary objective for future research is, therefore, to generalize DenseQR to be compatible with both sparse and dense retrieval frameworks, enabling its deployment irrespective of dataset characteristics.

A promising avenue to achieve this lies in adapting a SPLADE-like mechanism within the existing DenseQR structure. Specifically, we may leverage the CLS token's final hidden state, which is currently unused as input to the dense retriever. This CLS state can be explicitly trained to identify and encode the most salient lexical tokens from the current question and the conversational history. By training the CLS embedding to focus on memorizing and outputting these important query terms (similar to generating an expanded query or lexical weight vector), this vector could then serve as input for a sparse, term-weighting retrieval system. We expect that the unified framework would enable DenseQR to be universally utilized in conversational settings, bridging the performance gap between semantic and lexical matching models.

References

Vaibhav Adlakha, Shehzaad Dhuliawala, Kaheer Suleman, Harm de Vries, and Siva Reddy. 2022. TopiOCQA: Open-domain Conversational Question Answering with Topic Switching. *Transactions of the Association for Computational Linguistics*, 10:468–483.

Raviteja Anantha, Svitlana Vakulenko, Zhucheng Tu, Shayne Longpre, Stephen Pulman, and Srinivas Chappidi. 2021. Open-domain question answering goes conversational via question rewriting. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 520–534, Online. Association for Computational Linguistics.

Artem Babenko and Victor Lempitsky. 2014. Additive quantization for extreme vector compression. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 931–938.

Yongjian Chen, Tao Guan, and Cheng Wang. 2010. Approximate nearest neighbor search by residual vector quantization. *Sensors (Basel, Switzerland)*, 10:11259 – 11273.

Jianfeng Gao, Chenyan Xiong, Paul Bennett, and Nick Craswell. 2022. Neural approaches to conversational information retrieval. *Preprint*, arXiv:2201.05176.

Yunah Jang, Kang-il Lee, Hyunkyung Bae, Hwanhee Lee, and Kyomin Jung. 2024. IterCQR: Iterative conversational query reformulation with retrieval guidance. 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 8121–8138, Mexico City, Mexico. Association for Computational Linguistics.

Yilong Lai, Jialong Wu, Congzhi Zhang, Haowen Sun, and Deyu Zhou. 2024. Adacqr: Enhancing query reformulation for conversational search via sparse and dense retrieval alignment. *arXiv preprint arXiv:2407.01965*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Sheng-Chieh Lin, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy Lin. 2020. Conversational question reformulation via sequence-to-sequence architectures and pretrained language models. *Preprint*, arXiv:2004.01909.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Kelong Mao, Zhicheng Dou, Bang Liu, Hongjin Qian, Fengran Mo, Xiangli Wu, Xiaohua Cheng, and Zhao Cao. 2023a. Search-oriented conversational query editing. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4160–4172, Toronto, Canada. Association for Computational Linguistics.

Kelong Mao, Zhicheng Dou, Fengran Mo, Jiewen Hou, Haonan Chen, and Hongjin Qian. 2023b. Large language models know your contextual search intent: A prompting framework for conversational search. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1211–1225, Singapore. Association for Computational Linguistics.

Fengran Mo, Kelong Mao, Yutao Zhu, Yihong Wu, Kaiyu Huang, and Jian-Yun Nie. 2023. Convgqr: Generative query reformulation for conversational search. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4998–5012.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. *PyTorch: an imperative style, high-performance deep learning library*. Curran Associates Inc., Red Hook, NY, USA.

Nikos Voskarides, Dan Li, Pengjie Ren, Evangelos Kanoulas, and Maarten de Rijke. 2020. Query resolution for conversational search with limited supervision. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 921–930.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zeqiu Wu, Yi Luan, Hannah Rashkin, David Reitter, Hannaneh Hajishirzi, Mari Ostendorf, and Gaurav Singh Tomar. 2022. Conqrr: Conversational query rewriting for retrieval with reinforcement learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10000–10014.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*.

Fanghua Ye, Meng Fang, Shenghui Li, and Emine Yilmaz. 2023. Enhancing conversational search: Large language model-aided informative query rewriting. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5985–6006, Singapore. Association for Computational Linguistics.

Chanwoong Yoon, Gangwoo Kim, Byeongguk Jeon, Sungdong Kim, Yohan Jo, and Jaewoo Kang. 2024. Ask optimal questions: Aligning large language models with retriever's preference in conversational search. *arXiv preprint arXiv:2402.11827*.

Hansi Zeng, Chen Luo, Bowen Jin, Sheikh Muhammad Sarwar, Tianxin Wei, and Hamed Zamani. 2024. Scalable and effective generative information retrieval. In *Proceedings of the ACM Web Conference 2024*, WWW '24, page 1441–1452, New York, NY, USA. Association for Computing Machinery.

A Implementation Details

DenseQR was trained for 3 epochs with learning rate of 2e-5 on 2x RTX A6000 48GB GPUs. For contrastive learning, we set batch size per GPU to 8 for query and 128 for passages. With in-batch negative sampling, each query was mapped to 1 positive and 127 negative samples. The same hyperparameters were used for both QReCC and TopiOCQA datasets.

B Details of Baselines

We compared the proposed model with following baselines: (1) **GPT2QR** (Anantha et al., 2021) trained GPT-2 based Transformer++ model with QReCC train dataset that generates rewritten queries. (2) **QuReTec** (Voskarides et al., 2020) expanded a query of current turn with relevant terms from corresponding conversation history. (3) **T5QR** (Lin et al., 2020) trained seq-to-seq T5 model to generate human rewrite queries using a conversational history and a current question as an input. (4) **ConQRR** (Wu et al., 2022) introduced reinforcement learning, enabling the T5 model to be trained using a retriever-based rewards. (5) **ConvGQR** (Mo et al., 2023) combined query rewriting and query expansion through knowledge infusion mechanism. (6) **EDIRCS** (Mao et al., 2023a) selected tokens needed to be newly generated and edited queries by replacing the selected tokens with generated tokens that leverage retrieval performances. (7) **IterCQR** (Jang et al., 2024) iteratively trained a query reformulation model with refined retriever-based rewards. (8) **LLM-Aided** (Ye et al., 2023) and (9) **LLM4CS** (Mao et al., 2023b) prompted to ChatGPT API to rewriter queries based on the conversational histories. (10) **RETPO** (Yoon et al., 2024) and (11) **AdaCQR** (Lai et al., 2024) used a high-quality queries generated by LLM as supervision labels and also learned retriever's preference of generated candidate queries.

C Case Study

The Table 6 and Table 7 show two cases of mask token's behavior with respect to no-context and long-context. Specifically, we passed the dense representations of the mask tokens through the LM head of DenseQR and examined the indices corresponding to the highest logits. For the case that contains no context, the mask tokens appear to have answer information. This behavior arises because our input

Type	No Context
Question	when will the new Dunkirk film be released on dvd?
Context	-
masks	Dunk, Dunkirk, film, making, is, begin, 2018, 2019
Answer	18 December 2017

Table 6: The case with no context.

Type	Long Context
Question	Did they also have a fan club?
Context	Q: What does the song Hotel California by the Eagles mean? A: Eagles described it as their "interpretation of the high life in Los Angeles" Q: What is Hotel California? A: Title track from the Eagles' album Q: Who is the singer of this song? A: Don Henley Q: When did this song release? A: February 1977 Q: Where was it recorded? A: Record plant, Los Angeles and criteria studios, Miami Q: When was the first one established? A: 1968 Q: Does it also have branches in other places? A: yes, Los Angeles, branch and Sausalito, California, location
masks	Los, Record, agles fan club is online
Answer	Yes, Jim Keltner Fan Club

Table 7: The case with long contexts.

format repeatedly includes question-answer pairs from conversational history; thus, during training, the mask tokens implicitly learn the positional role of answers and act as stand-ins for them.