Real-time Ad retrieval via LLM-generative Commercial Intention for Sponsored Search Advertising

Anonymous ACL submission

Abstract

The integration of Large Language Models (LLMs) with retrieval systems has shown promising potential in retrieving documents (docs) or advertisements (ads) for a given query. Existing LLM-based retrieval methods generate numeric or content-based DocIDs to retrieve corresponding docs/ads. However, due to the unique mapping between numeric IDs and docs, as well as the time-consuming nature of content extraction, these DocIDs suffer from semantic inefficiency and cannot scale to large-scale corpora. In this paper, we propose a framework of **R**eal-time **A**d **RE**trieval (RARE), which uses generated text named commercial intentions (CIs) as a bridge to directly retrieve ads for queries in real-time. CIs are generated by a customized LLM that incorporates commercial knowledge through injection, making it domain-adaptive commercial intents. CIs and ads have a one-to-many relationship, leading to a lightweight set of CIs. Therefore, the proposed RARE is not only semantically efficient but also capable of scaling to large-scale corpus. We have implemented RARE in a real-world online system that handles daily search volumes in the tens of millions. The implementation has yielded significant benefits: a 5.04% increase in consumption, a 6.37% rise in Gross Merchandise Volume (GMV), a 1.28% enhancement in click-through rate (CTR), a 5.29% increase in shallow conversions.

1 Introduction

Ad retrieval is essential for efficiently identifying high-quality subsets from extensive ad libraries, which contain millions or even billions of candidates. Traditional ad retrieval models typically use a two-stage process (Wang et al., 2024a). Initially, the query retrieves relevant keywords, which are then used to fetch corresponding ads. These methods are primarily categorized into word-based and semantic-based. The former, e.g., TF-IDF (Ramos et al., 2003), directly obtains keywords by parsing user queries. The latter, e.g., DSSM (Huang et al., 2013), obtains embeddings for queries and keywords in a shared semantic space using a dual encoder, allowing for retrieval based on semantic similarities. However, the existing two-stage retrieval methods amplify the difference between user queries and advertiser-purchased keywords, resulting in numerous missed retrieval issues. The end-to-end single-stage approach (Zhu et al., 2018, Gao et al., 2020) alleviates the missed recall issue by directly retrieving ad candidates. However, these methods still face limitations in exploring the deeper commercial intention of users and ads due to their insufficient reasoning capabilities and lack of domain knowledge.

In recent years, LLMs (Touvron et al., 2023; Yang et al., 2023) have garnered widespread attention and made remarkable achievements in the field of search and recommendation (Pradeep et al., 2023; Tang et al., 2024). The combination of LLMs with retrieval systems enables a deeper understanding of user expressions and doc information. Most LLM-based retrieval approaches first construct an index of docs, which is achieved by training the model to learn the connections between documents and their identifiers (DocIDs). Subsequently, during the retrieval phase, the model processes a query as input and generates the corresponding DocIDs (Li et al., 2024a). However, existing methods rely on DocIDs (numeric IDs and document content) for indexing, creating an inefficient correspondence between documents and indexes. This makes DocIDs inefficient and difficult to scale for large-scale corpora. For example, DSI (Tay et al., 2022) employs numeric IDs to represent documents and establish connections between user queries and numeric IDs. LTRGR (Li et al., 2024b) extracts document content, i.e., article title and body, to represent the document and implement the retrieval from a user query to a document.

Using semantically inefficient DocIDs (Zeng et al., 2023) brings about the following drawbacks. Firstly, the inference efficiency is low due to the inefficient correspondence between DocIDs and candidates (Wang et al., 2024b), making it difficult to achieve real-time generation on large-scale datasets. Secondly, representing docs/ads with DocIDs (numeric IDs and extracted contents from the documents) does not fully leverage the capabilities of LLM in commercial intention mining, thus hindering the effective exploration of the advertiser's intent. Thirdly, it exhibits poor generalization. When new candidates emerge, it is typically required to retrain the model or update the FM-index (Ferragina and Manzini, 2000) to learn their DocIDs. It becomes challenging to swiftly update or remove candidates as well. Ad retrieval demands real-time fetching of large sets of ads aligned with the user's commercial intent, making semantically inefficient DocIDs unsuitable. Therefore, utilizing the powerful semantic capabilities of LLMs to design more effective semantic tokens for indexing and developing a more comprehensive end-to-end architecture has become a crucial challenge.

To address the challenge, we developed a Real-time LLM-Generative Ad Retrieval framework named RARE that leverages semantically enriched generated text as a bridge to directly retrieve ads for queries in real-time. Our method enables real-time generalization to all candidates, enhances inference efficiency and fully leverages the commercial intention mining capability of the LLMs. Specifically, RARE initially utilizes a knowledge-injected LLM to generate commercial intentions (CIs) for the ads in the corpus, selects a limited but comprehensive CIs, and then constructs an index that maps CIs to their corresponding ads (one-to-many). Then, the well-trained LLM generates CIs in real-time for queries and retrieves corresponding ads based on the generated CIs. CIs are defined as aggregations of keywords, generated by knowledge-injected LLM based on relevant materials of ads including titles of ads and descriptions of ads, etc. Since LLM is injected with world knowledge and domain-specific knowledge in advertising, the generated CIs are also knowledge-driven, enabling a comprehensive exploration of the commercial intent behind ads and queries. While keywords are manually selected by advertisers, CIs surpass the limitations of manually selected keywords, allowing for more precise targeting of commercial traffic.

To ensure the diversity and specialization of CIs generated by the LLM, we inject various domain-specific knowledge, i.e., ad-related information, into the base LLM during fine-tuning. Upon receiving a query, the customized LLM generates CIs in real-time and retrieves corresponding ads using the pre-built index. The CIs generated in real-time are confined within a defined set, and the one-to-many correspondence between CIs and ads makes the decoding process highly efficient. For new ad candidates, RARE generates CIs directly based on their title and other materials, without the need to retrain the model or modify the CIs set. In summary, RARE is capable of generating commercial intentions in real-time, efficiently retrieving advertisements, and generalizing to new ads swiftly.

The main contributions of our work are as follows: (1) We propose a novel end-to-end generative retrieval framework named RARE to achieve real-time retrieval, which is the first known work on LLM-generative architecture that displays real-time retrieving on millions of databases. (2) We propose a method for knowledge injection and format fine-tuning to enable the base LLM to uncover the deep commercial intentions of advertisers and users, which are expressed as CIs. (3) We have deployed an online system based on LLMs for real-time inference and ad retrieval, which serves tens of millions of users in real-world scenarios everyday. (4) We conduct online A/B testing and offline experiments to verify the effectiveness of RARE. A/B testing has yielded a 5.04% increase in consumption, a 6.37% increase in Gross Merchandise Value (GMV), a 1.28% increase in Click-Through Rate (CTR), a 5.29% increase in conversions, and a remarkable 24.77% increase in deep conversions. Simultaneously, in terms of offline evaluation metrics, RARE has significantly improved PVR, MAP, and HR@500 compared to the traditional two-stage online retrieval methods.

2 Related Works

Technological advancements have driven the evolution of retrieval methods. Beginning with word-based inverted index retrieval, progressing to deep learning-based semantic retrieval, and to retrieval methods leveraging large language models (LLMs) (Zhao et al., 2023). Numerous studies integrate large language models (LLMs) into retrieval system, which can be broadly categorized into encoder-based retrievers and generative-based

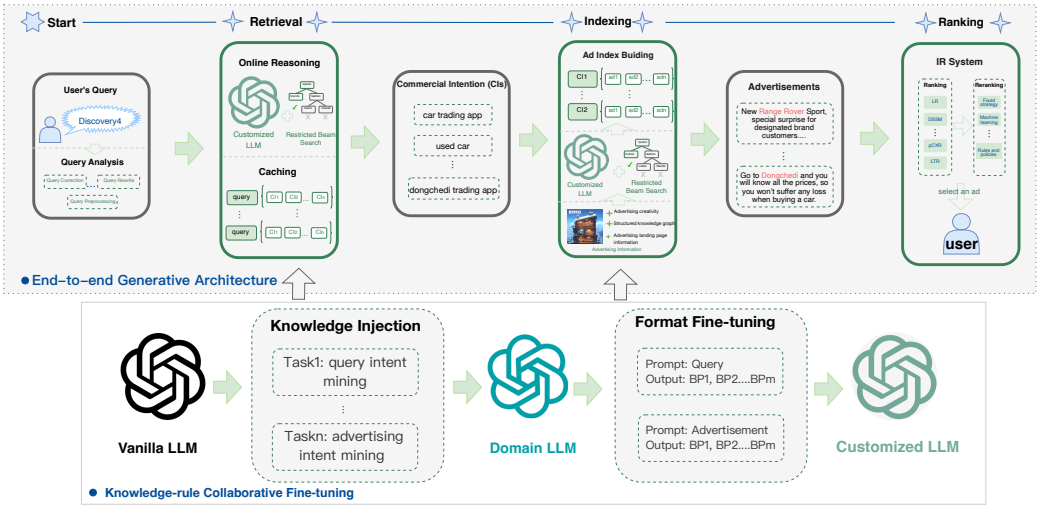

Figure 1: The Real-time LLM-Generative Ad Retrieval framework named RARE inputs a query into the LLM to generate commercial intentions (CIs), which are then used to inquire the inverted index and retrieve ads. We obtain customized LLM by injecting knowledge and learning rules into the base LLM to generate the CIs used in RARE.

retrievers.

Encoder-based LLMs Retriever Encoder-based retrievers leverage the semantic capabilities of LLMs to obtain text embedding (Hou et al., 2024). For instance, cpt-text (Neelakantan et al., 2022) uses contrastive learning to train GPT-3 (Brown, 2020) from scratch, generating high-quality embedding of text. GTR (Ni et al., 2021) utilizes the T5 (Raffel et al., 2020) model, fine-tuning it to derive text vector representations. NoteLLM (Zhang et al., 2024) achieves text-image embeddings by incorporating additional training tasks and modifying the LLM model structure.

Generative-based LLMs Retriever Generative based retrievers utilize the generative capabilities of LLMs to construct end-to-end retrieval models. Some approaches, such as DSI (Tay et al., 2022), NCI (Wang et al., 2022), Tiger (Rajput et al., 2024), use document IDs as the generation target to implement query retrieval for docs/ads. These methods leverage LLMs to learn the correspondence between docs/ads and their IDs, directly generating the ID of the relevant docs/ads for query retrieval. Other approaches, such as SEAL (Bevilacqua et al., 2022)and LTRGR (Li et al., 2024b), use document content as an intermediary to achieve document retrieval. They employ FM-Index to generate fragments that appear in the document, facilitating query-to-document retrieval. MINDER (Li et al., 2023) employs pseudo-queries and document content for retrieval, but this significantly increases indexing volume, making it unsuitable for scenarios with large candidate sets.

Semantic Tokens LLM-generative retrieval typically employs DocIDs to perform query-to-document retrieval tasks. Existing DocIDs mainly include numeric IDs and document content. For instance, the numeric IDs in Tiger is represented as a tuple of discrete semantic tokens. In LTRGR, document content consists of predefined sequences that appear within the document. However, the semantic tokens used in these approaches are ID-like features, which suffer from low decoding efficiency because each DocID corresponds to a single candidate. For new documents or advertisements, it is necessary to retrain the model or rebuild the FM-Index to learn their DocID representation, making it challenging to fast update or delete ads.

3 Method

In this paper, we introduce a novel end-to-end generative retrieval architecture designed for online retrieval, named **R**eal-time **A**d **re**trieval (RARE). RARE employs commercial intentions (CIs) as the generation target to directly retrieve candidate advertisements based on user queries. To mine more accurately and diversely commercial intentions of user and ads, we propose a knowledge-rule collaborative fine-tuning approach. To facilitate real-time retrieval from millions of candidates, we introduce an efficient inference method that includes caching the head query and utilizing restricted beam search.

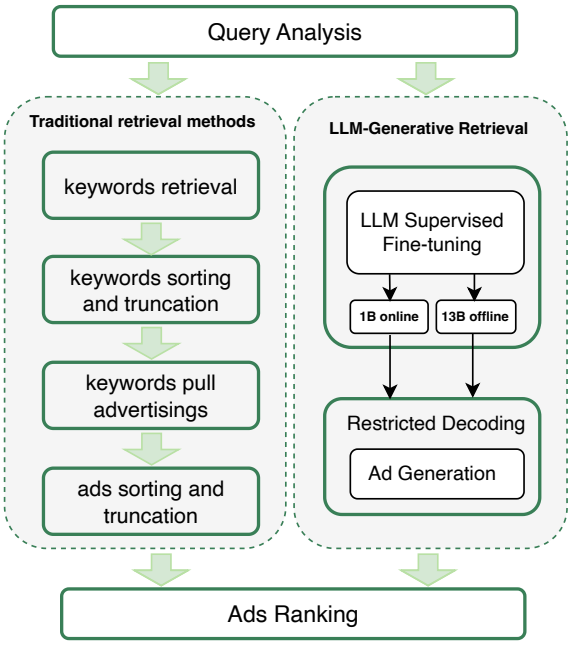

Figure 2: Comparison between LOGR and traditional retrieval methods. RARE directly generates candidate ads based on user queries by using LLMs, effectively shortening the link structure.

The overall architecture of RARE is illustrated in Figure 1.

3.1 An End-to-end Generative Architecture

The proposed end-to-end generative architecture differs from the traditional query-keyword-ads framework by enabling the direct retrieval of advertisements. This approach effectively shortens the link structure, allowing advertisements to overcome the limitations of keyword bidding and helping advertisers acquire more accurate traffic, as illustrated in Figure 2. LLM-generative retrieval employs commercial intentions (CIs)—text with specific linguistic meaning—as the generation target to facilitate the recall process from query to advertisement. Section 3.3 presents the details of CIs.

The RARE architecture primarily consists of two components: retrieval and indexing. When a user submits a query, the retrieval component performs intent mining on the query to generate CIs. Subsequently, the indexing component uses these CIs to retrieve the final advertisements from the CI-Ads index, thereby achieving direct retrieval of advertisements based on the query.

Indexing We utilize offline 13B-LLMs to generate the CIs for advertisements. Initially, we conduct commercial intention inference on the entire advertisement library to construct a commercial intention set. For subsequent new advertisements, we perform restricted inference based on the current commercial intention set and then update the index of CIs-Ads. More detailed implementation specifics are discussed in 3.3.

Retrieval The generation of CIs for queries is based on a combination of offline and online parallel processing. Initially, we check whether the current query matches an entry in the cache. If a match is found, we directly retrieve the corresponding commercial intention to fetch advertisements. Otherwise, we use the trained LLM for real-time inference. We introduce the detailed implementation in 3.4.

The end-to-end architecture that uses commercial intentions (CIs) as bridge has the following advantages.

(1) Efficient inference. RARE allows one commercial intention to correspond to thousands of similar ads, enabling online retrieval across tens of millions of ads. This overcomes the limitation of previous methods that one semantic token is matched to a single candidate.

(2) Diverse commercial intentions mining. RARE uses knowledge infused LLM to compress and summarize ads into commercial intention space, clustering similar ads and enhancing diversity by reducing homogeneous retrieval.

(3) High-quality candidate ads. RARE enables ads to surpass purchased word limitations, exploring deeper user intent and providing higher-quality candidates.

(4) Strong generalization ability. For new ads, there is no need to retrain the model or update the FM-index. We simply perform restricted decoding on the existing CIs set, which is memory-efficient, fast to build, and dynamically update.

3.2 Knowledge-rule Collaborative Fine-tuning

To achieve real-time reasoning, the model must directly output the CIs based on the prompt, without including intermediate reasoning information. Figure 4 illustrates the impact of output length on reasoning time. Simultaneously, to ensure that the generated CIs incorporates ad-related knowledge and is output in the desired format, we apply knowledge injection and format fine-tuning to the LLM.

Stage 1: knowledge injection This stage primarily involves the injection of advertising knowledge. We train the model on various advertising tasks,

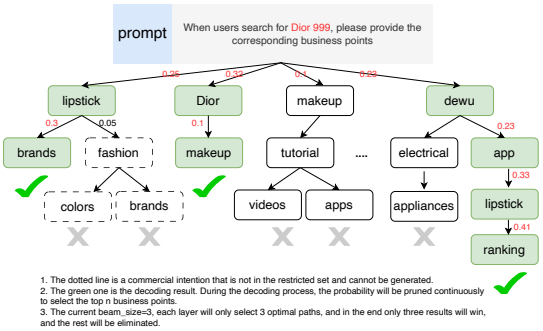

Figure 3: Restricted Beam Search Decoding Process.

such as query intent mining and advertising intent mining. The data format at this stage is natural language, sourced from open-source LLMs, and includes rich reasoning processes and guidance information. This approach effectively enhances the model's performance on advertising tasks.

Stage 2: format fine-tuning This stage focuses on learning the generation format and enhancing the diversity of generated commercial intentions (CIs). The input consists of a prompt containing query/ad information, and the output is a set of CIs. The CIs for queries/ads in the fine-tuning stage are derived from keywords that have been clicked online, ensuring high relevance. In this stage, only thousands of data are used to enable the model to quickly follow instructions and minimize the output of irrelevant information.

3.3 Indexing

We use the customized LLM to build the index of ads and CIs. We first feed advertising information, such as titles and landing pages, into the customized LLM for constrained generation, obtaining the CIs set and then construct the CIs-Ads index.

Commercial Intentions (CIs) We utilize a customized LLM to generate CIs for ad library. Given a prompt x containing advertising information (such as ad title, landing page), it generates b commercial intentions with the highest probability, as shown in formula 1,

$$CIs = arg \max_{y^{<1>}...y^{<b>}} \sum_{t=1}^{T} logP(y_t^{<1>}, y_t^{<2>}... y_t^{<b>}|x, y_{t-1}^{<1>}, y_{t-1}^{<2>}...y_{t-1}^{<b>}), \quad (1)$$

where $y_t^{<i>}$ is the output of the top-i commercial intention at time t, b is the beam size, and T is the maximum length of the CIs.

Ad Indexing Building We initially employ a offline 13B model to infer commercial intentions from all advertisement information, establishing a correspondence between CIs and ads. Subsequently, we construct an index of CIs-Ads. The advertisement information is derived from ad titles, landing pages, and delivery materials of all online ads in the ad library. We utilize a restricted beam search to ensure that the generated CIs are within the predefined set of two million CIs. For newly added advertisements, we only need to perform restricted inference on the existing set of CIs and update the CIs-Ads index. It enables the rapid deployment of new advertisements.

3.4 Efficient inference

Efficient inference is essential for real-time retrieval from millions of candidate sets, as sponsored search advertising has strict requirements on retrieval time. In this section, we mainly introduce efficient decoding methods: restricted decoding and caching technology.

Restricted Beam Search In this work, we employ a restricted beam search algorithm for generating commercial intentions(CIs), ensuring that the model's outputs are confined to a predefined CIs. We have developed a CUDA-based implementation of the restricted beam search and integrated it with the LLM inference process to enable parallel generation of beam-size CIs, thereby enhancing decoding efficiency. Furthermore, we introduced a truncation function within the restricted beam search framework, which allows for the discarding of individual tokens with lower scores to improve the accuracy of the model's output. The specific restriction process is illustrated in Figure 3.

Caching Technology The search system exhibits a pronounced long-tail effect, where 5% of the queries account for 60% of the total query requests. To enhance inference efficiency, we perform offline inference and storage for these high-frequency head queries. When a user submits a query, the system first checks the offline cache. If a match is found, the result is returned immediately. If no match is found, the inference service processes the request.

Offline processing is less time-sensitive, allowing us to utilize a large model, specifically a 13B parameter model, to handle these queries. We also employ a higher temperature setting and a larger beam size to increase the diversity of CIs. The top 12% of head queries are cached, accounting for 68% of online requests, effectively reducing time consumption and improving inference efficiency. By caching millions of head queries offline, we can reduce online machine consumption by 70%, which not only decreases the time required for inference but also enhances the quality of CIs for head queries.

Online inference, however, has stringent time constraints, typically requiring completion within milliseconds. Figure 4 illustrates the impact of different CIs lengths on inference time using an NVIDIA A100, so a smaller 1B model and smaller CIs maxlength are used for online inference. We developes an online inference GPU cluster that evenly distributes cache-miss queries across all GPUs for inference, thereby achieving effective load balancing. For queries without commercial intent, the model will refuse to generate a response to ensure the relevance of the CIs generated online.

4 Experiments

In this section, we mainly introduce our experimental settings, the effect of RARE on offline metrics and online systems, and ablation studies.

4.1 Experimental Settings

Training Dataset To enable the LLM learns the relationships between queries, ad titles, and keywords in the advertising domain, we collected online logs and designed several tasks, including query intent mining, ad intent mining, and ad generation, to facilitate knowledge injection. These data inputs are derived from online desensitized log information, while the outputs are generated from open-source LLMs. The outputs include detailed reasoning processes, which help improve the model's reasoning capabilities. Format fine-tuning primarily involves the commercial intentions of queries and advertisements. These data are sourced from real online interactions and are combined according to fixed rules.

Evaluation Dataset To better evaluate the model's effectiveness, we collected the head queries and advertisement pairs clicked online over the course of one day. After cleaning the data, we

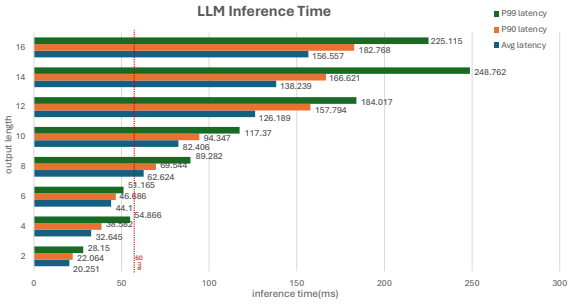

Figure 4: The impact of different output lengths on inference time.

obtained 5,000 queries and 150,000 ads to serve as the ground truth and each query has a maximum of 1,000 ad candidates. We then evaluated the existing two-stage (query-keywords-ads) online models and LOGR against this dataset.

Baselines We compare RARE with online retrieval methods, including word-based BM25 and semantic-based BERT. BM25 segment the query to be calculated into w_1, w_2, ..., w_n, and then calculate the relevance score of each w_i and the keyword. Finally, these scores are accumulated to finally get the text similarity calculation result. BERT uses a 4 layers of transformer network to obtain the embedding of query and keywords, and then use HNSW(Malkov and Yashunin, 2018) to retrieve candidate keywords for the query.

Evaluation Metrics We use PVR, Hit Ratio(HR@K) (Alsini et al., 2020) and Mean Average Precision(MAP) (Cormack and Lynam, 2006) to evaluate the generation effect of RARE.

PVR in ad recall means coverage, which is the proportion of requests with ad recall. As shown in Formula 2, adpv is the number of requests with ad recall, and pv is the number of requests.

$$PVR = adpv/pv \tag{2}$$

Hit Ratio(HR@K) is shown in Formula 3, where GT represents all candidate ads, and Number of Hits is the total count of recalled ads that belong to the candidate set.

$$HR@K = \frac{Number of Hits@K}{|GT|} \tag{3}$$

Mean Average Precision(MAP) is the average of AP of all candidates are generated. The average accuracy of q is shown in formula 4, where Ω_q represents the ground-truth results, p_{qj} represents the position of ad_j in the generated list, and $p_{qj} < p_{qi}$

Retriever	HR@50	HR@100	HR@500	MAP	PVR
BM25(online baseline)	0.0870	0.1336	0.3807	0.1232	76.01%
BERT(online baseline)	**0.0995**	0.1518	0.4311	0.1719	78.65%
RARE	0.0985	**0.1541**	**0.5134**	**0.1845**	**95.05%**

Table 1: Comparison of the offline performance between RARE and online baselines.

Scenarios	Consumption	GMV	CTR	Shallow Conversions	Deep Conversions
Scenario 1	+5.04%	+6.37%	+1.28%	+5.29%	+24.77%
Scenario 2	+7.18%	+5.03%	-	+6.85%	+5.93%
Scenario 3	+4.50%	+5.02%	-0.74%	+17.07%	+7.86%

Table 2: RARE applied to the retrieval system, the results of online A/B testing.

means that ad_j ranks before ad_i in the generated list.

$$AP_q = \frac{1}{\Omega_q} \sum_{i \in \Omega_q} \frac{\sum_{j \in \Omega_q} h(p_{qj} < p_{qi}) + 1}{p_{qi}} \quad (4)$$

MAP represents the average AP of all query q, as shown in formula 5.

$$MAP = \frac{\sum_{q \in Q} AP_q}{Q} \quad (5)$$

Implementation Details We utilize open-source LLM[1] as the backbone, with parameters including 1B-Dense-SFT and 13B-Dense-SFT. For the offline cache, we employ a 13B model with a beam size of 256, a temperature of 0.8, and a maximum output length of 6. For online inference, we use a 1B model with a beam size of 50, a temperature of 0.7, and a maximum output length of 4 to ensure that inference latency remains within 60 milliseconds. The CIs-Ads index is updated daily, while the commercial intention set is updated monthly.

4.2 Experimental Results

Offline Evaluation Results We selected two traditional online recall methods as baselines: the word-based BM25 (Robertson et al., 2009) and the semantic-based BERT (Kenton and Toutanova, 2019). We used HR@K and MAP to evaluate the recall performance of RARE, BM25, and BERT on offline data. As shown in Table 1, RARE not only significantly improved the recall rate but also enhanced the HR and MAP metrics, outperforming both the BM25 and BERT strategies.

[1]To adhere to anonymity guidelines, the model name is concealed and will be stated in the full paper.

Online A/B Testing We applied RARE to ad recall in various online search scenarios[2]. During a one-month A/B testing experiment with a 20% user sample, we observed significant benefits across multiple scenarios, including increased system revenue, enhanced user experience, and boosted advertiser conversions. Take scenario 1 as an example, we achieved a 5.04% increase in consumption, a 6.37% increase in GMV, a 1.28% increase in CTR and a 5.29% increase in shallow conversions. Additionally, CTR and conversion rates were significantly improved, demonstrating that RARE can effectively understand user intent and deliver high-quality ads. The results of the online A/B testing experiment are presented in Table 2.

Using scenario 1 as an example, we analyzed the impact of RARE within a specific industry. As illustrated in Figure 5, RARE significantly increases consumption, GMV, and conversion numbers across the top 10 industries.

4.3 Ablation study

In this section, we primarily analyze the Knowledge-rule collaborative fine-tuning method. Table 4 presents the results of RARE on offline evaluation data, demonstrating that the two-stage fine-tuning approach significantly outperforms the one-stage fine-tuning. Table 3 shows the inference results of different training methods. From Table 3, we can observe the following: (1)Zero-shot LLM lacks a reasoning process for prompts, relying mainly on the surface-level understanding of queries, which results in numerous poor cases. (2)Knowledge injection stage teaches LLM how to

[2]To comply with anonymity rules, we conceal the information of scenarios. Scenarios 1-3 represent a large-scale formal search system, with an average daily search volume in the tens of millions. Detailed scenarios will be presented in the full paper.

prompt: When users search for **shantytown housing**, please provide the corresponding commercial intentions.
zero-shot output: (A total of 26 commercial intentions, but some of them are poorly relevant) commercial intentions: shantytown, demolition, housing supply, price, user feedback, case sharing..., etc.
satge1: knowledge injection (A total of 9 commercial intentions, with good correlation and insufficient diversity.) Step 1: Mining user query intent Shantytown reconstruction is a government-led urban renewal project involving demolition.... Step 2: Generate commercial intentions Shantytown reconstruction policy, demolition compensation standards..... Step3: Ensure commercial intentions are relevant to the query commercial intentions such as shantytown reconstruction policy, Demolition compensation standards... Final result: shantytown reconstruction policy, demolition compensation standards.....
satge1 + stage2: knowledge injection && format fine-tuning (A total of 12 commercial intentions, with good correlation and diversity, but a small number) Anjuke rents a house, second-hand housing, house hunting...
satge1 + stage2 + restricted decoding (A total of with 142 commercial intentions, with with good correlation, good diversity and large number) Anjuke house hunting, demolition compensation, new house decoration, renovation of old houses, public housing application.....

Table 3: commercial intention generation effects based on different fine-tuning methods.

metrics	single stage	two stages
HR@500	0.171	0.513
MAP	0.1540	0.1845
Recall	59.51%	95.05%
Avg CIs	22.78	74.49
Accuracy	90.4%	96.5%

Table 4: Ablation studies on knowledge injection (stage1) and format fine-tuning (stage2).

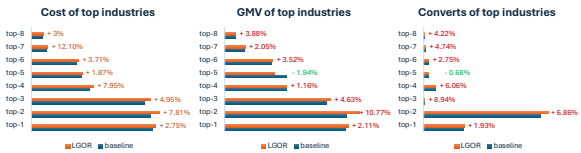

Figure 5: RARE's performance in the top ten industries.

reason, enabling it to analyze the query in the first step, provide commercial intentions in the second step, and judge relevance in the third step. However, this approach lacks diversity. (3)Format fine-tuning stage instructs the model to follow rules while generating more diverse results. Since most of the training data in this stage comes from online data, the commercial intentions are relatively rich. However, due to the free generation process, the number of commercial intentions is limited. (4)Incorporating restricted decoding can increase the number of commercial intentions while ensuring relevance and diversity.

5 Conclusion

In this paper, we propose a real-time LLM-generative ad retrieval framework called RARE. This framework utilizes commercial intentions (CIs) as semantic tokens for real-time retrieval and employs restricted decoding, allowing the model to generate CIs from a fixed set in parallel, thereby significantly enhancing decoding efficiency. CIs are knowledge-driven and leverage the reasoning capabilities of LLMs to effectively discern the intentions behind users and advertisements. Our architecture not only demonstrates strong generalization capabilities but also achieves high inference efficiency, enabling real-time generation and retrieval from a library containing tens of millions of ads. To mine deeper intentions of ads and users, we introduce an effective fine-tuning method named knowledge-rule collaborative fine-tuning. Evaluations on offline data and online A/B testing indicate that our architecture achieves state-of-the-art (SOTA) advertising retrieval performance, while substantially improving search system revenue, user experience and advertiser conversion.

Limitations

We briefly outline some of the existing limitations of our work. First, the end-to-end generation architecture proposed in this paper primarily facilitates the generation process from query/ad to commercial intention, while the correlation between query and ad is managed by downstream processes. In future work, we aim to integrate correlation judgment into LLMs, enabling the model to assess the relevance between prompts and commercial intentions during the generation process. This integration of generation and judgment is expected to further enhance the generation effect.

Currently, we have implemented the knowledge injection method by designing domain-specific tasks during the fine-tuning stage. To avoid losing the world knowledge embedded in the LLMs during the pretraining stage, the amount of data used has been relatively small. Moving forward, we plan to apply knowledge injection during the pretraining stage as well. This will involve collecting large-scale advertising domain knowledge and other general task data to train the model more comprehensively during the pretraining phase.

References

Areej Alsini, Du Q Huynh, and Amitava Datta. 2020. Hit ratio: An evaluation metric for hashtag recommendation. *arXiv preprint arXiv:2010.01258*.

Michele Bevilacqua, Giuseppe Ottaviano, Patrick Lewis, Scott Yih, Sebastian Riedel, and Fabio Petroni. 2022. Autoregressive search engines: Generating substrings as document identifiers. *Advances in Neural Information Processing Systems*, 35:31668–31683.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Gordon V Cormack and Thomas R Lynam. 2006. Statistical precision of information retrieval evaluation. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 533–540.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2020. Autoregressive entity retrieval. *arXiv preprint arXiv:2010.00904*.

Paolo Ferragina and Giovanni Manzini. 2000. Opportunistic data structures with applications. In *Proceedings 41st annual symposium on foundations of computer science*, pages 390–398. IEEE.

Weihao Gao, Xiangjun Fan, Jiankai Sun, Kai Jia, Wenzhi Xiao, Chong Wang, and Xiaobing Liu. 2020. Deep retrieval: An end-to-end learnable structure model for large-scale recommendations. *arXiv preprint arXiv:2007.07203*.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2333–2338.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. *Bert: Pre-training of deep bidirectional transformers for language understanding*, volume 1.

Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou. 2024a. From matching to generation: A survey on generative information retrieval. *arXiv preprint arXiv:2404.14851*.

Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. 2023. Multiview identifiers enhanced generative retrieval. *arXiv preprint arXiv:2305.16675*.

Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. 2024b. Learning to rank in generative retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8716–8723.

Yu A Malkov and Dmitry A Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence*, 42(4):824–836.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, et al. 2022. Text and code embeddings by contrastive pretraining. *arXiv preprint arXiv:2201.10005*.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Ábrego, Ji Ma, Vincent Y Zhao, Yi Luan, Keith B Hall, Ming-Wei Chang, et al. 2021. Large dual encoders are generalizable retrievers. *arXiv preprint arXiv:2112.07899*.

Ronak Pradeep, Kai Hui, Jai Gupta, Adam D Lelkes, Honglei Zhuang, Jimmy Lin, Donald Metzler, and Vinh Q Tran. 2023. How does generative retrieval scale to millions of passages? *arXiv preprint arXiv:2305.11841*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, et al. 2024. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36.

Juan Ramos et al. 2003. *Using tf-idf to determine word relevance in document queries*, volume 242.

Stephen Robertson, Hugo Zaragoza, et al. 2009. *The probabilistic relevance framework: BM25 and beyond*, volume 3. Now Publishers, Inc.

Yubao Tang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, and Xueqi Cheng. 2024. List-wise generative retrieval models via a sequential learning process. *ACM Transactions on Information Systems*, 42(5):1–31.

Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022. Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems*, 35:21831–21843.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Yang Wang, Zheyi Sha, Kunhai Lin, Chaobing Feng, Kunhong Zhu, Lipeng Wang, Xuewu Jiao, Fei Huang, Chao Ye, Dengwu He, et al. 2024a. One-step reach: Llm-based keyword generation for sponsored search advertising. In *Companion Proceedings of the ACM on Web Conference 2024*, pages 1604–1608.

Yidan Wang, Zhaochun Ren, Weiwei Sun, Jiyuan Yang, Zhixiang Liang, Xin Chen, Ruobing Xie, Su Yan, Xu Zhang, Pengjie Ren, et al. 2024b. Enhanced generative recommendation via content and collaboration integration. *arXiv preprint arXiv:2403.18480*.

Yujing Wang, Yingyan Hou, Haonan Wang, Ziming Miao, Shibin Wu, Qi Chen, Yuqing Xia, Chengmin Chi, Guoshuai Zhao, Zheng Liu, et al. 2022. A neural corpus indexer for document retrieval. *Advances in Neural Information Processing Systems*, 35:25600–25614.

Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Hansi Zeng, Chen Luo, Bowen Jin, Sheikh Muhammad Sarwar, Tianxin Wei, and Hamed Zamani. 2023. Scalable and effective generative information retrieval. corr abs/2311.09134 (2023).

Chao Zhang, Shiwei Wu, Haoxin Zhang, Tong Xu, Yan Gao, Yao Hu, and Enhong Chen. 2024. Notellm: A retrievable large language model for note recommendation. In *Companion Proceedings of the ACM on Web Conference 2024*, pages 170–179.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Han Zhu, Xiang Li, Pengye Zhang, Guozheng Li, Jie He, Han Li, and Kun Gai. 2018. Learning tree-based deep model for recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1079–1088.

A Fine-tuning Prompt Templates

In this section, we mainly introduce the fine-tuning prompt templates. Knowledge-rule collaborative fine-tuning mainly includes two stages, namely knowledge injection and format fine-tuning. The fine-tuning data of knowledge injection mainly includes query intent mining, advertising intent mining and advertising words buying. Table 5 shows the fine-tuning data of these two stages in detail. Injecting a large amount of data (on the order of hundreds of thousands or millions) during the fine-tuning phase can cause LLMs to lose their general knowledge and reasoning capabilities. Therefore, in this phase of fine-tuning, we selected only 2,000 instances for each task.

B GPU Cluster For Online Inference

We use L40 cards to implement online inference. At peak, the GPU cluster can handle hundreds of thousands of requests simultaneously. By designing efficient inference and scheduling mechanisms, the maximum GPU utilization can reach 90% and effectively achieve load balancing.

C Related Work of Beam Search

As a decoding strategy for heuristic search, beam search has been widely used in many works. For example, DSI uses beam search to generate a sorted list of candidate documents, and Tiger uses beam search to generate multiple candidate product IDs at once. As early as a few years ago, the combination of seq2seq and restricted Beam Search has achieved a win-win effect and efficiency in entity linking and document retrieval. For example, GENRE (De Cao et al., 2020) applied restricted Beam Search to document retrieval tasks and achieved SOTA.

task name	prompt	data size
query intent mining (knowledge injection)	As a search advertising system, given the user query = xxx, identify the commercial intentions based on the user's search characteristics. Provide ten commercial intentions for each category from different intents, and finally, compile all the results into a single line separated by commas, step by step.	2000
advertising intent mining (knowledge injection)	As an advertiser, given the ad title = xxx, identify the commercial intentions for ad based on search characteristics. Provide ten commercial intentions for each category from different intents, and finally, compile all the results into a single line separated by commas, step by step.	2000
ad generation (knowledge injection)	As a search advertising system, your task is to recommend suitable ads for user queries. To meet the user's potential intent and ensure the relevance between the query and the ad title, you should provide a diverse range of ads. Given the query = xxx, please generate ten appropriate ad titles for each intent based on search characteristics, step by step.	2000
query understanding (knowledge injection)	As a search advertising system, your task is to rewrite and expand the user's query to ensure it meets the diverse needs of users. The rewritten terms should be as varied as possible. For instance, if the query is "SF Express," the expanded terms could include "cross-border logistics", "express shipping", "express delivery", "express service" and "logistics warehousing". Following this example, when the query is = xxx, provide the expanded terms that meet these requirements, step by step.	2000
query generates commercial intentions (format fine-tuning)	As a search advertising system, given a user query of xxx, please provide a broader range of commercial intentions that still align with the user's potential intentions.	2000
ad generates commercial intentions (format fine-tuning)	As a search advertiser, given the ad title = xxx, please identify the commercial intentions of the ad based on the characteristics of the search and various user intentions.	2000

Table 5: The details of fine-tune prompt template.