Event-Centric Query Expansion in Web Search

Yanan Zhang[1*] **Weijie Cui**[2*] **Yangfan Zhang**[1] **Xiaoling Bai**[1†]
Zhe Zhang[2] **Jin Ma**[2] **Xiang Chen**[1] **Tianhua Zhou**[1]
[1]Tencent Inc. [2]University of Science and Technology of China
{yananzhang, devinbai}@tencent.com, can@mail.ustc.edu.cn

Abstract

In search engines, query expansion (**QE**) is a crucial technique to improve search experience. Previous studies often rely on long-term search log mining, which leads to slow updates and is sub-optimal for time-sensitive news searches. In this work, we present **E**vent-Centric **Q**uery **E**xpansion (EQE), a novel QE system that addresses these issues by mining the best expansion from a significant amount of potential events rapidly and accurately. This system consists of four stages, i.e., *event collection*, *event reformulation*, *semantic retrieval* and *online ranking*. Specifically, we first collect and filter news headlines from websites. Then we propose a generation model that incorporates contrastive learning and prompt-tuning techniques to reformulate these headlines to concise candidates. Additionally, we fine-tune a dual-tower semantic model to function as an encoder for event retrieval and explore a two-stage contrastive training approach to enhance the accuracy of event retrieval. Finally, we rank the retrieved events and select the optimal one as QE, which is then used to improve the retrieval of event-related documents. Through offline analysis and online A/B testing, we observe that the EQE system significantly improves many metrics compared to the baseline. The system has been deployed in Tencent QQ Browser Search and served hundreds of millions of users. The dataset and baseline codes are available at https://open-event-hub.github.io/eqe.

1 Introduction

People are always eager to obtain details and updates on current hot events through search engines. To efficiently return dozens of relevant documents from billions of candidates, most search engines use a "retrieval-rank-rerank-mixed rank"architecture, as illustrated in Figure 1.

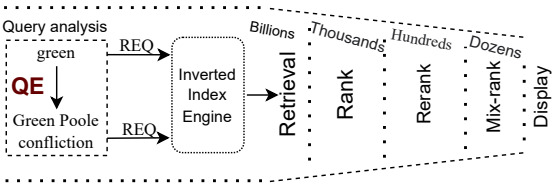

Figure 1: Overview of the query expansion process and search system in Tencent QQ Browser Search.

Queries, particularly those relying on keywords, present a tough challenge for query intent understanding due to their brevity, absence of world knowledge, and lack of grammatical structure (Broder et al., 2007). When a significant event takes place, the search intent of users subsequently can rapidly and drastically shift. For example, during the Green-Poole conflict, a user searching for "green" may be seeking information about the color, while many others desire news about NBA player Draymond Green. While methods based on search log mining (Jansen et al., 2007; Zamora et al., 2014; Caruccio et al., 2015) are still commonly used for query intent understanding, they are limited by their reliance on the accumulation of posterior data and struggle with timely and accurately processing the intent for recent events, making it difficult to retrieve and rank event-related documents. Recent approaches (Zhang et al., 2020a; Nogueira et al., 2019; Sun et al., 2022) suggest using additional context from query-associate documents or entities to improve the performance of query understanding. However, they still face challenges in real-time search scenarios.

To tackle this challenge, we present EQE, a real-time query expansion system specifically designed to efficiently capture query intent for ongoing events. As depicted in Figure 1, EQE extends the original query with the most fulfilling event, selected from a large pool of candidate events. By performing the same retrieval step with both the original and expanded queries, more results related to the event will be returned. This bypass architec-

[*] Equal Contributions
[†] Corresponding author

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 5: Industry Track, pages 464–475

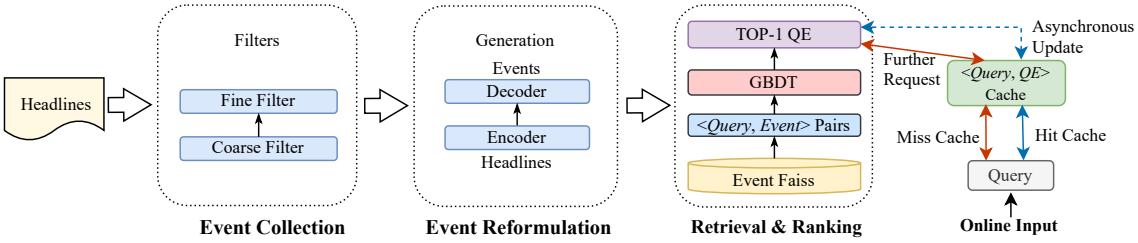

Figure 2: Architecture of the proposed EQE system.

ture effectively ensures system flexibility. When there are sufficient machine resources, the number of bypasses can be increased, and multiple query expansions can be used to improve the performance of document retrieval.

Our EQE system employs a four-stage structure, as illustrated in Figure 2, consisting of event collection, event reformulation, semantic retrieval, and online ranking. Events are collected from news headlines as they are typically more concise and event-centric, compared to body texts which are lengthier and contain extraneous information. To guarantee the accuracy of the collected events, we employ a combination of rule-based coarse filtering and language model-based fine filtering (§ 2.1). The collected headlines, as described in Appendix A, may contain noise, irregular grammar, and lack of world knowledge, making them unsuitable for query expansions. To solve these problems in such scenarios, we reformulate them using a generation model, which is more effective than extractive models (§ 2.2). Our method employs keyword-based prompt learning to make generated content more controllable and applies contrastive learning on the encoder to counteract embedding degradation (Gao et al., 2019). After this step, we obtain a high-quality candidate set of event-centric QE. For a given query, to further narrow down the QE candidate set, we utilize a supervised SimCSE model (Gao et al., 2021) to retrieve relevant QEs. SimCSE effectively improves the accuracy of retrieval by addressing the issue of representation space degradation. Moreover, inspired by (Gillick et al., 2019), we employ a two-stage training approach that incorporates informative hard negative samples for each query, resulting in a further improvement in representation quality (§ 2.3). Finally, we design an online ranking module to select the best QE. Features of query-side, event-side, and interactive are considered comprehensively (§ 2.4).

As far as we know, EQE is the first query expansion solution developed explicitly for real-time event intent. The efficiency of EQE is verified through offline analysis and online A/B testing. The main contributions of this work are summarized as follows:

- We propose a real-time and efficient query expansion system for timely search scenarios. The system comprises four stages: event collection, event reformulation, semantic retrieval, and online ranking.
- In the event reformulation stage, we introduce an effective generation model that leverages prompt learning and contrastive learning techniques to produce a high-quality candidate set of QE.
- In the semantic retrieval stage, we employ a two-stage contrastive learning approach to improve the accuracy of semantic retrieval.
- Offline analysis and online A/B testing on Tencent QQ Browser Search demonstrate the effectiveness of our proposed EQE framework.

2 Method

In this section, we describe our proposed framework of EQE shown in Figure 2. We first introduce the scheme for event collection in industrial scenarios. We then elaborate on event reformulation and semantic retrieval, describing how we use contrastive learning and prompt learning to improve model performance. Finally, we discuss online ranking, revealing how to select the optimal expansion.

2.1 Event Collection

The essential phase in the "Event Collection" process is to identify events from the vast amount of newly uploaded content on the Internet. We filter events from the headlines using a two-step method that includes a rule-based coarse filter and a semantically-driven fine filter.

Coarse Filter. After using basic feature filters such as publication time, page type, site type, etc., we gather approximately 50 million news article headlines over the duration of recent six months. As described in Appendix E, the headlines generated by these heuristic rules include irregular syntax, missing event components, numerous events, etc., posing a barrier to subsequent event reformulation. So further we use the LTP toolkit (Che et al., 2021) to extract event triggers from headlines and drop headlines missing event elements or with multiple events (the number of triggers more than 2).

Fine Filter. The rule-based coarse filter is based on pre-defined patterns and has limited recognition abilities. To address these issues and further improve the accuracy of event collection, we train an event detection model based on RoBERTa (Liu et al., 2019). We employ six experts to annotate around 200,000 samples, which are utilized to train the model. Subsequently, we use this model to infer event probabilities for the coarsely filtered headlines and filter them using a predefined threshold, achieving a 95% accuracy rate in detecting event-related headlines.

2.2 Event Reformulation

This step aims to make events qualified for query expansion by reformulating them using a generation model, addressing issues such as noise, irregular grammar, and low-frequency words. As illustrated in Figure 3, we introduce two significant improvements to the encoder-decoder architecture-based model. Firstly, we enhance the controllability of the generation process using prompt learning. Secondly, we optimize the representation quality of headlines using contrastive learning. By simultaneously optimizing these two tasks, we can effectively refine events for query expansion.

Prompt Guidance. To ensure important information is not overlooked, we leverage prompt learning technology when training a generation model. Unlike prior work, we propose adaptive keyword templates to provide guidance during sentence generation. Firstly, we use the KeyBERT model (Grootendorst, 2020) to extract the most essential nouns from the sentence. We then insert the extracted keyword into a fixed template to form a keyword template, denoted as T. Finally, we concatenate the headline H, the keyword template T, and the target qualified event E with special

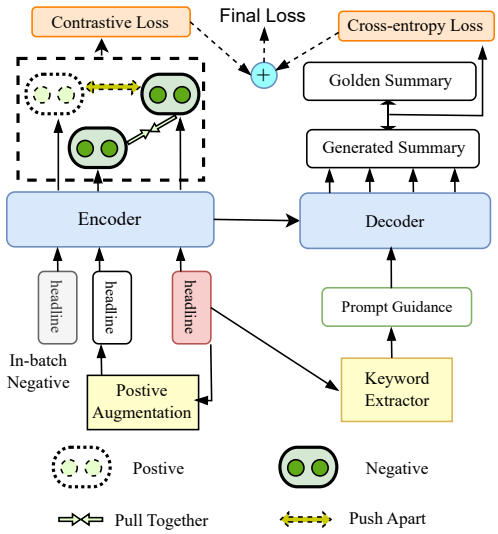

Figure 3: Structure of event reformulation model.

tokens as "[CLS]H [SEP][CLS]TE[SEP]"[1]. In this setup, the front segment "[CLS]H[SEP]" and the latter segment "[CLS]TE[SEP]" serve as the inputs for the encoder and decoder, respectively. It is worth noting that E is omitted during the inference phase.

Contrastive Learning. Previous studies show that natural language generation tasks suffer from representation space degradation problems, which can be alleviated by contrastive learning (Gao et al., 2019). In our model, the embedding corresponding to the [CLS] token of the encoder is regarded as the headline representation and contrastive learning is performed based on it. Specifically, for each headline, we perform a position swap of its two terms to obtain a positive example headline and then use contrastive learning to pull the representations of positive headline pairs closer and push away the representations of negative headline pairs (i.e., in-batch negative samples).

The following is a description of the contrastive learning loss calculation process within a batch.

(a) In a batch of size 2N, the training data consists of 2N pairs denoted by $\{(H_1, E_1), (H_1^+, E_1), \cdots, (H_N, E_N), (H_N^+, E_N)\}$, where $< H_i, H_i^+ >$ denote a pair of similar headlines, both of which can be paired with the event phrase E_i. Contrastive learning aims to pull semantically close neighbors (i.e. (H_i, H_i^+)) together and pushing apart non-neighbors (i.e. $(H_i, H_j^+), i, j \in \{0, 1, ..., N\}$

[1]For the BART model, the start token ID for the decoder is [CLS], while for the mT5 model, it is [PAD].

and $j \neq i$).

(b) The above 2N samples are passed through the encoder to obtain 2N embeddings that are denoted as $(e_1, e_2, .., e_N; e_1^+, e_2^+, ..., e_N^+)$.

(c) The 2N embeddings are used to compute the contrastive learning loss, which is included as part of the loss for this mini-batch. Let τ denote the temperature hyper-parameter and $\text{sim}(e_1, e_2)$ denote the cosine similarity. Then the contrastive learning loss, denoted as L_{cl} is:

$$L_{cl} = -\sum_{i=1}^{N} \log \frac{e^{\text{sim}(e_i, e_i^+)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(e_i, e_j^+)/\tau}}$$

The training data for the Event Reformulation stage consists of pairs <$headline$, $event$> that are sampled from user-click logs. Specifically, we employ two methods to construct the target event: regarding the user query as the target event and extracting the event from the headline using the LTP toolkit. To ensure that the constructed events are qualified as the target for the generation model, we further filter out pairs that do not meet our standards for loyalty, integrity, and cleanness through expert annotation.

2.3 Semantic Retrieval

In this step, we use a dual-tower semantic model based on contrastive learning to retrieve highly satisfying events for a given query. The work of Xiong et al. (2021) points out that the dominance of uninformative negative samples leads to a bottleneck in the recall system, therefore, we employ a novel two-stage training paradigm.

For a given query, the positive samples are events obtained from the reformulated events of its clicked headlines. Then we use features, such as Jaccard Distance (Jaccard, 1901), BERTScore (Zhang et al., 2020b), etc., to only keep relevant pairs as training samples. In the first stage, we use these positive samples with shuffled negative ones to finetune a naive dual-tower retrieval model and obtain the encoder. The weights of the model are initialized using the parameters of RoBERTa. After finetuning, we build an event vector library with more than 4 million entries using this encoder. For a given query vector, based on Faiss (Johnson et al., 2019), the top-K events are recalled according to the cosine similarity. Events located at the upper and lower thresholds of the threshold are regarded as hard neg samples, where the bounds are pre-defined by distribution statistics. In the second stage, we replace the randomly shuffled negative events with hard negative events and retrain the model initialized with the encoder obtained from the first stage. This results in the final retrieval model.

2.4 Online Ranking

Actually, selecting the optimal expansion requires considering multiple factors, such as relevance, event popularity, and timeliness. Therefore, we use the classic light-weight sorting model GBDT (Friedman, 2001), which is compatible with the interpretation of online features. We incorporate three types of features to build the model: query-side, event-side, and interaction. Query-side features encompass query domain classification, entity recognition, word segmentation, and word weighting, among others, generated by existing online operators. Event-side features involve event found time and event popularity (the size of the cluster to which an event belongs). The interaction features include semantic similarity, BM25 (Robertson and Zaragoza, 2009), and entity matching between the query and event.

We describe the method of collecting training samples. For each query, we input the events obtained from the previous stage into the online search engine to obtain the search results pages. Then, we select the page that best satisfies the event intent of the query through expert annotation, and its corresponding event is labeled as a positive sample for the query, while the other events are labeled as negative samples. We obtain 50,000 samples, which are used to train the GBDT model for inferring the best query expansion.

3 System Architecture

In this section, we describe our baseline and EQE architecture in detail.

3.1 Baseline Approach

We first take a glance at our QE baseline, which is a query graph analysis framework. We devise a Query-Document click graph $\mathcal{G}$ based on the click propagation algorithm (Jiang et al., 2016). In order to prioritize time-sensitive queries, we limit our analysis to click-pairs from news websites that occurred within a 3-day window. To mitigate the risk of irrelevant results, we integrate BM25 score

to the adjacency matrix of the graph, denoted as C where each entry $C_{i,j}$ is the weight of the edge between query q_i and document d_j, specifically formulated as:

$$C_{i,j} = \begin{cases} \alpha \cdot \text{BM25}(q_i, d_j) + 1, & \text{with edge} \\ 0, & \text{no edge} \end{cases} \quad (1)$$

where α (set to 0.2) is a smoothing coefficient.

Representations of queries and documents are iteratively updated according to Eq. (2) and Eq. (3), respectively.

$$Q_i^{(n)} = \frac{1}{\left\| \sum_j^{|Doc|} C_{i,j} \cdot D_j^{(n-1)} \right\|_2} \sum_{j=1}^{|Doc|} C_{i,j} \cdot D_j^{(n-1)} \quad (2)$$

$$D_j^{(n)} = \frac{1}{\left\| \sum_{i=1}^{|Query|} C_{i,j} \cdot Q_i^{(n)} \right\|_2} \sum_{i=1}^{|Query|} C_{i,j} \cdot Q_i^{(n)} \quad (3)$$

where $Q_i^{(n)}$ and $D_i^{(n)}$ are the representations of q_i and d_i at the n-th iteration respectively. After the n-th iteration, we perform clustering on $Q_i^{(n)}$ to obtain the query clusters. For each cluster, we select the most frequent query as the expansion of other queries.

3.2 EQE System

Figure 2 illustrates the EQE system, which can be divided into two parts: offline and online. The offline system sequentially processes streaming data from the internet, performing event collection, event reformulation, and Faiss indexing for fast response. These steps can be processed in parallel. In the online part, when a query arrives, a GBDT ranking model selects the top-1 candidate as the query expansion based on rich features.

Furthermore, a rapidly updated caching system stores pairs of <$query$, top-1 $expansion$> to intercept requests to meet the time-consuming demands. Upon receiving a new query request, the system first seeks a pre-prepared QE in the cache. If not found, the system initiates further retrieval and ranking modules to obtain the QE. This QE is then returned to the main search system, while the <$query$, $expansion$> pair is written into the cache for any subsequent identical query requests. On the other hand, if a match is found, the cached result is immediately returned to the main search system. Concurrently, the caching system undergoes an asynchronous update in preparation for future requests. The implementation of an asynchronous execution pipeline does not boost the response delay of the mainstream search process. Therefore, the response time of the popular query is mainly consumed by querying the caching system. Only the stages of retrieval and ranking executed for infrequent queries lead to an increase in the response time of the search engine.

Finally, EQE covers nearly 50% of online traffic, while the other half, such as those requiring explicit knowledge, has already been addressed by other intent understanding modules, is therefore not considered within this scope. Online data indicates that the query expansion system elevates search latency by only 10 ms, evincing the efficacy of the proposed module.

Methods	Recall@100	Recall@150	Recall@200
Baseline	0.41	0.47	0.58
EQE	0.58	0.65	0.74
Improve	+ 41.46%	+ 38.29%	+ 27.58%

Table 1: Offline performance comparison.

4 Experiments

We conduct a series of comprehensive evaluations, both in offline and online environments, incorporating quantitative and qualitative aspects, to prove the advantages of EQE.

4.1 End-to-end evaluation

Firstly, we present the results of the implementation of the EQE system online, taking into account both offline and online metrics.

Offline Evaluation. We measure the offline performance of EQE using $Recall@K$ metric. As illustrated in Eq. (4), given a query Q, the clicked documents by users are denoted as $T = \{t_1, \ldots, t_N\}$, which are regarded as as the target. The top-K documents set recalled by the QE module is denoted as $I = \{i_1, \ldots, i_k\}$. $Recall@K$ is defined as:

$$Recall@K = \frac{\sum_{i=1}^{K} i_i \in T}{N} \quad (4)$$

We first collect user click-log over a certain period of time, where documents are retrieved by original queries without the influence of QE. Specifically, in our scenario, we collect news, videos, and user-generated content (UGC). Meanwhile, we record documents retrieved by both expansions produced by EQE and the baseline approach. After 7 days of accumulation, a total of 850,000 valid online requests are collected. As shown in Table 1, after evaluating $Recall@K$ at different thresholds, it can be seen that EQE significantly surpasses the online baseline.

	ΔGSB	CTR	PCTR	UCTR
EQE	+12.5%	+6.64%	+6.23%	+5.03%

Table 2: Online A/B test of EQE implemented.

Online Evaluation. We construct a 30-day A/B experiment with 1% of online traffic to gather feedback from millions of users and study the online performance of the EQE compared to the strong baseline described in Section 3.1. QEs derived from both frameworks are utilized in downstream tasks (document retrieval and sorting). For online evaluation, we are mainly concerned about the business metrics that impact user experience, such as ΔGSB (Zou et al., 2021), CTR (Rangadurai et al., 2022), PCTR and UCTR (Qin et al., 2022). As shown in Table 2, EQE outperforms the baseline and gains improvements of 6.44%, 6.23% and 5.03% on CTR, PCTR and UCTR, respectively, indicating its SOTA performance.

4.2 Performance of Event Collection

We choose the intersection of "Hot Search List" from various platforms as our evaluation set. This decision serves two purposes: firstly, it can eliminate unfair comparisons due to platform-specific content biases; secondly, these events are highly representative in the search domain, as users consistently demonstrate in them and desire to retrieve relevant information swiftly. We employ two annotation experts to assist in the evaluation process, which involved: 1) Collecting these events from different platforms (such as Baidu and Weibo) to find the earliest time they appeared respectively. Admittedly, since we cannot accurately determine the initial creation time of these events on other platforms, we resort to the first appearance time on the "Hot Search List" as an approximation; 2) Identifying the time when the first publish emerged on the Internet; 3) Recording events discovery coverage rate at several time points.

Figure 4 illustrates the average coverage of Baidu, Weibo, and EQE at different time points after "Hot Search List" events occurred. The coverage of each system is recorded at time points of 1, 2, 5, 10, 15, and 20 minutes. As shown, at the 5-minute time point, EQE discovers more than 10% of the events in advance compared to the other systems.

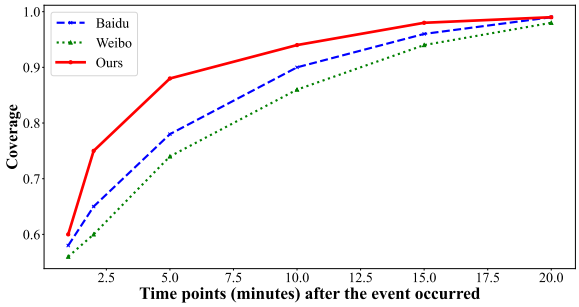

Figure 4: The x-axis represents the time points (in minutes) after the earliest occurrence of events on the internet, while the y-axis represents the coverage rate of events discovery for each system.

Models	Rouge-L	BLEU	BERTScore
BART (vanilla)	0.8391	0.7692	0.9266
BART + CL	0.8406	0.7724	0.9278
BART + PG	0.8458	0.7777	0.9294
BART + CL + PG	**0.8480**	**0.7822**	**0.9312**
mT5 (vanilla)	0.8453	0.7781	0.9297
mT5 + CL	0.8489	0.7833	0.9315
mT5 + PG	0.8511	0.7857	0.9322
mT5 + CL + PG	**0.8533**	**0.7897**	**0.9336**

Table 3: Automated evaluation of ablation experiments. CL and PG are abbreviations for contrastive learning and prompt guidance, respectively.

4.3 Performance of Event Reformulation

We evaluate the performance of the proposed generation model by computing automated metrics. We disclose an annotated test dataset, which is called Title2EventPhrase. Production procedures and analysis of Title2EventPhrase are introduced in Appendix B and C. We conduct ablation experiments to measure the effect of the prompt and contrastive learning modules. Furthermore, to verify the universality of these two modules, we utilize BART (Lewis et al., 2020) and mT5 (Xue et al., 2021) as the backbone networks, respectively. [2] Experiments are measured with the ROUGE (Lin, 2004), BLEU (Papineni et al., 2002) and BERTScore metrics. As shown in Table 3, our proposed components significantly improved the generation performance.

4.4 Performance of Semantic Retrieval

In this section, we evaluate the effectiveness of our proposed semantic retrieval model against the

[2]We use a popular version of the Chinese BART model available at https://huggingface.co/fnlp/bart-large-chinese, and the base version of mT5 available at https://huggingface.co/google/mt5-base. It is worth noting that the number of transformer layers in both models is consistent.

Models	Recall@10	MRR@10	AUC
RoBERTa (vanilla)	0.74	0.43	0.80
RoBERTa + CL	0.75	0.45	0.82
RoBERTa + CL + 2T	**0.80**	**0.51**	**0.87**

Table 4: Evaluation of semantic models. CL and 2T are abbreviations for contrastive learning and two-stage training with hard neg samples, respectively.

baseline by employing three definitive performance metrics: standard Recall@K (Jegou et al., 2010), MRR@K (Craswell, 2009) and AUC (Fawcett, 2006).

We construct a testing dataset with a similar method to obtain <$query$, $event$> pairs as the training dataset mentioned in Section 2.3, with relevance labels annotated by experts. We sample them at different time periods to ensure training and testing datasets have the same distribution but are non-overlapping. Table 4 shows the advantages of our training scheme over other baseline models. In addition, we also visualize the results of a two-dimensional t-SNE (Van der Maaten and Hinton, 2008) graph on the embedding of 100,000 queries, further demonstrating the effectiveness of our proposed method in addressing the problem of representation degradation. For more details, please refer to Appendix D and Figure 7.

5 Related Work

Query understanding (**QU**) is a fundamental task of information retrieval (**IR**), which aims to help reformulate query, predict query intent, and ultimately improve the document relevance modeling (Zhang et al., 2020a). As an essential method for QU, query expansion (**QE**) involves the addition of relevant terms or specific information to a query to clarify intention and enhance retrieval performance (Rosin et al., 2021). In recent years, most QE methods have been based on word embedding techniques (Srinivasan et al., 2022; Padaki et al., 2020; Azad and Deepak, 2019; Kuzi et al., 2016; Zamani and Croft, 2016), which select semantically related terms as expansions. Usually, word embeddings are learned in two ways, one is based on the semantic vector of terms and the other is based on retrieval relevance (Diaz et al., 2016; Zamani and Croft, 2017). Meanwhile, external data sources, such as Wikipedia and WordNet, have also been utilized for QE (Azad and Deepak, 2019).

However, these conventional QE methods mainly rely on mining search logs or pre-built expansion libraries, which leads to slow update rates in time-sensitive scenarios. On the other hand, new occurring events in real time can meet the timeliness requirements well, and mining QE from them is a promising research direction. Recently Deng et al. (2022) construct a large-scale dataset aiming at extracting event arguments, like *subject*, *predicate* and *object*, from news headlines. However, structured outputs from extractive models (Lu et al., 2022; Gao et al., 2022) might not be fully utilized by the retrieval and ranking modules. We thus turn to generative models for event reformation. Normalized events serve as crucial signals for time-sensitive query expansion, which makes the largest contribution to our work.

6 Conclusion

This paper presents our solution for large-scale event-centric query expansion, called EQE, which is able to efficiently capture query intent for ongoing events and help our search engine to retrieve more event-related results. Advanced techniques are involved in each stage of EQE to improve performance. Offline experiments and online A/B tests verify the effectiveness of EQE. We have deployed the system in Tencent QQ Browser Search to serve hundreds of millions of users. Meanwhile, we share the design and deployment scheme.

References

Hiteshwar Kumar Azad and Akshay Deepak. 2019. A new approach for query expansion using wikipedia and wordnet. *Information sciences*, 492:147–163.

Andrei Z Broder, Marcus Fontoura, Evgeniy Gabrilovich, Amruta Joshi, Vanja Josifovski, and Tong Zhang. 2007. Robust classification of rare queries using web knowledge. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 231–238.

Loredana Caruccio, Vincenzo Deufemia, and Giuseppe Polese. 2015. Understanding user intent on the web through interaction mining. *Journal of Visual Languages & Computing*, 31:230–236.

Wanxiang Che, Yunlong Feng, Libo Qin, and Ting Liu. 2021. N-LTP: An open-source neural language technology platform for Chinese. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 42–49, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nick Craswell. 2009. *Mean Reciprocal Rank*, pages 1703–1703. Springer US, Boston, MA.

Haolin Deng, Yanan Zhang, Yangfan Zhang, Wangyang Ying, Changlong Yu, Jun Gao, Wei Wang, Xiaoling Bai, Nan Yang, Jin Ma, Xiang Chen, and Tianhua Zhou. 2022. Title2Event: Benchmarking open event extraction with a large-scale Chinese title dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6511–6524, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Fernando Diaz, Bhaskar Mitra, and Nick Craswell. 2016. Query expansion with locally-trained word embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 367–377, Berlin, Germany. Association for Computational Linguistics.

Tom Fawcett. 2006. An introduction to roc analysis. *Pattern Recognition Letters*, 27(8):861–874. ROC Analysis in Pattern Recognition.

Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. In *Annals of statistics*, pages 1189–1232. JSTOR.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. Representation degeneration problem in training natural language generation models. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Jun Gao, Changlong Yu, Wei Wang, Huan Zhao, and Ruifeng Xu. 2022. Mask-then-fill: A flexible and effective data augmentation framework for event extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4537–4544, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Daniel Gillick, Sayali Kulkarni, Larry Lansing, Alessandro Presta, Jason Baldridge, Eugene Ie, and Diego Garcia-Olano. 2019. Learning dense representations for entity retrieval. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 528–537, Hong Kong, China. Association for Computational Linguistics.

Maarten Grootendorst. 2020. Keybert: Minimal keyword extraction with bert.

Paul Jaccard. 1901. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulletin del la Société Vaudoise des Sciences Naturelles*, 37:547–579.

Bernard J. Jansen, Danielle L. Booth, and Amanda Spink. 2007. Determining the user intent of web search engine queries. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 1149–1150. ACM.

Herve Jegou, Matthijs Douze, and Cordelia Schmid. 2010. Product quantization for nearest neighbor search. *IEEE transactions on pattern analysis and machine intelligence*, 33(1):117–128.

Shan Jiang, Yuening Hu, Changsung Kang, Tim Daly Jr., Dawei Yin, Yi Chang, and ChengXiang Zhai. 2016. Learning query and document relevance from a web-scale click graph. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2016, Pisa, Italy, July 17-21, 2016*, pages 185–194. ACM.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Saar Kuzi, Anna Shtok, and Oren Kurland. 2016. Query expansion using word embeddings. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*, pages 1929–1932. ACM.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Michael Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *Cornell University - arXiv*.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland. Association for Computational Linguistics.

Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019. From doc2query to doctttttquery. *Online preprint*, 6.

Ramith Padaki, Zhuyun Dai, and Jamie Callan. 2020. Rethinking query expansion for bert reranking. In *Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II 42*, pages 297–304. Springer.

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

Yuqi Qin, Pengfei Wang, Biyu Ma, and Zhe Zhang. 2022. A multi-interest evolution story: Applying psychology in query-based recommendation for inferring customer intention. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 1655–1665.

Kaushik Rangadurai, Yiqun Liu, Siddarth Malreddy, Xiaoyi Liu, Piyush Maheshwari, Vishwanath Sangale, and Fedor Borisyuk. 2022. Nxtpost: User to post recommendations in facebook groups. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3792–3800.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, 3:333–389.

Guy D Rosin, Ido Guy, and Kira Radinsky. 2021. Event-driven query expansion. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 391–399.

Krishna Srinivasan, Karthik Raman, Anupam Samanta, Lingrui Liao, Luca Bertelli, and Mike Bendersky. 2022. Quill: Query intent with large language models using retrieval augmentation and multi-stage distillation. *ArXiv preprint*, abs/2210.15718.

Zhongkai Sun, Sixing Lu, Chengyuan Ma, Xiaohu Liu, and Edward Guo. 2022. Query expansion and entity weighting for query reformulation retrieval in voice assistant systems. In *WSDM 2022*.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

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 *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Hamed Zamani and W Bruce Croft. 2016. Embedding-based query language models. In *Proceedings of the 2016 ACM international conference on the theory of information retrieval*, pages 147–156.

Hamed Zamani and W. Bruce Croft. 2017. Relevance-based word embedding. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 505–514. ACM.

Juan Zamora, Marcelo Mendoza, and Héctor Allende. 2014. Query intent detection based on query log mining. *J. Web Eng.*, 13(1&2):24–52.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, and Xueqi Cheng. 2020a. Query understanding via intent description generation. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 1823–1832. ACM.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Lixin Zou, Shengqiang Zhang, Hengyi Cai, Dehong Ma, Suqi Cheng, Shuaiqiang Wang, Daiting Shi, Zhicong Cheng, and Dawei Yin. 2021. Pre-trained language model based ranking in baidu search. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 4014–4022.

A Characteristics of News Headlines

News headlines, designed to catch the reader's attention and highlight the editor's perspective, may contain redundant information beyond the event itself. In addition, they often use irregular grammar for the sake of memorability, and some words require extensive knowledge to comprehend. We provided several examples in Figure 5, which clearly demonstrate the necessity of using generation models to reformulate them before using them as query expansion.

B Title2EventPhrase Construction

Data Collection. We collect a broad range of Chinese web pages from January to March 2022, using web crawler logs from Tencent QQ Browser Search, and choose a reliable business tool to identify web pages that mention events (primarily from news websites). Following this, we extract the titles of the chosen web pages and automatically label them with our predefined topics. Any titles that contain toxic content are removed. To ensure a diverse range of events, we conduct data sampling every ten days during the crawling period. This reduces the frequency of events that belong to the most commonly occurring topics, resulting in a more balanced distribution of topics. In total, we collected over 100,000 instances.

Annotation Standard. We summarize some essential parts of our annotation criteria. Our goal is to obtain clear and concise event descriptions from the titles and to extract the most chief (core) events from titles that contain multiple events. To achieve this, we have outlined some important specifications below:

1) Our definition of an event is a real-world behavior or change in state. It is worth noting that statements like policy notices or subjective opinions are not considered events. Furthermore, if a title is unclear or contains multiple unrelated events (e.g. news roundup), it should be flagged as "invalid" by annotators.

2) We have specified some rules to clarify the labeling of event phrases: **a)** definite (non-interrogative), fluent (good readability) description of the event in the title; **b)** consistent with the fact described in the title; **c)** if there is a progressive relationship between multiple events in the title, they need to be included to ensure the integrity of the information; **d)** quantifiers, gerunds, and complements need to be removed if they do not affect the understanding of the event, otherwise should be retained.

Crowdsourced Annotation. We cooperate with crowdsourcing companies to hire human annotators. After multi-rounds of training in three weeks, 27 annotators are selected. We pay them ¥0.3 per instance. Meanwhile, four experts participated in two rounds of annotation checking for quality control. For each instance, a human annotator is asked to write the core event phase independently. To reduce the annotation difficulty, we provide a reference output along with the raw title. In the beginning, the reference output is mined from the query in click-log. After 10,000 labeled instances are collected, we train a better BART model for the rest of the annotation process. Also, we allow the annotators to refer to search engines to acquire domain knowledge. The crowd-sourced annotation is conducted in batches with two rounds of quality checking before being integrated into the final version of our dataset.

Two rounds Checking. Each time the crowd-sourced annotation of a batch is completed, it is sent to four experts to check whether they meet the requirements of our annotation standard. Instances that do not pass the quality check will be sent back for revision, attached with the specific reasons for rejection. This process repeats until the acceptance rate reaches 90%. Then, the current batch is sent to the authors for dual-check. The authors will randomly check 30% of the instances and send unqualified instances back to the experts along with the reasons for rejection. Slight adjustments on annotation standards also take place in this phase. This process repeats until the acceptance rate reaches 95%.

C Title2EventPhrase Analysis

Overview. Table 5 shows the overview of the Title2EventPhrase dataset, including data size, topic numbers, and the average length of titles and events.

Attribute	Value
Data Size	41351
Number of Topics	25
Avg. Len. of Title	25.85
Avg. Len. of Event	16.68

Table 5: The overall statistics of Title2EventPhrase.

Challenges	Examples	Description
Noise	**冰雪之上的新活力!长春冰雪新天地跨年演唱会拉开帷幕** (New vitality above ice and snow! Changchun Ice and Snow Xintiandi New Year's Eve Concert kicks off.)	The first sentence provides a commentary on the event, aimed at capturing the reader's interest, but it does not directly convey the essential details of the event.
Irregular Grammar	**对标宏光 MINI EV!售价2.99万起，奇瑞QQ冰淇淋正式上市** (Benchmarking Against Wuling Hongguang Mini EV! Priced From ￥29,900, Chery QQ Icecream Officially Launched.)	The subject of the predicate**"对标 (Benchmarking Against)"**, is omitted since it appears at the next sentence.
Lack of word knowledge	**保尔特17号洞冲刺打鸟赶完赛** (Poulter Rushes to Score Birdie, Finishing the Match on 17.)	**"打鸟(Score Birdie)"** is a term in golf but can be literally interpreted as "hit bird" in Chinese. Domain knowledge is needed to properly identify the predicate.

Figure 5: Challenges and examples of news headlines.

Filter Type	Examples	Description
Non-event Headline	**iphone13为啥好，请看介绍** (Why is iPhone 13 good? Please see the introduction.)	This sentence does not contain event-related information.
Missing Event Components	**最新消息，已全部删除** (Latest news, all deleted.)	This sentence lacks the subject of the event.
Numerous Events	**早报|起亚高管怒斥比亚迪；居民存款破纪录；驾校教练撞脸杨洋走红...** (Early News: Kia Exec Heavily Criticizes BYD; Record-breaking Bank Savings from Residents; Driving School Coach Goes Viral After Resembling Yang Yang...)	This sentence contains three different events, all of which pose a great challenge to downstream tasks.
Irregular Syntax	**黎明觉醒突然** (Dawn awakens suddenly)	This sentence is incomplete, lacking a verb and an object.
Interrogative Sentence	**孩子近视危害大，如何才能有效预防？** (Near-sightedness in children is a major hazard. How can it be effectively prevented?)	The sentence is a question and does not contain any objective.

Figure 6: Filter rules in coarse filter stage.

Topic Distribution. Table 6 lists 25 topics with their respective numbers and proportions. The distribution of topics in the dataset is obviously long-tailed. The largest number of topics is Society, whose proportion exceeds 31%, while 11 topics account for less than 1%.

Challenge Distribution. In this section, We analyze the scale of observed challenges described in Figure 5. We randomly select 1,000 headlines and manually annotate which challenge type it belongs to. The annotation result shows that 27% of sampled headlines have redundant expressions, 12% of them suffer from irregular grammar issues, and 11% of them require domain knowledge for sentence understanding.

D Event Representation Analysis

As mentioned in Section 2.3, our baseline model suffers from the issue of representation space degradation, leading to poor generation and retrieval performance.

In Figure 7, we present 2-dimensional t-SNE visualizations of the representation space obtained from queries without and with contrastive learning. As demonstrated in the left part of the figure, without contrastive learning, the model encodes queries into a smaller space with more collapses. As a comparison, the addition of contrastive learning expands the embedding space with better alignment and uniformity.

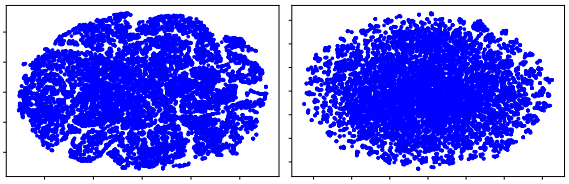

Figure 7: The t-SNE visualization of event representations from encoders without and with contrastive learning.

Topic	Count	Proportion
社会 (Society)	12985	31.40%
财经 (Finance)	5877	14.21%
体育 (Sports)	4504	10.89%
时事 (Current Events)	4078	9.86%
科技 (Technology)	2698	6.52%
娱乐 (Entertainment)	1903	4.60%
教育 (Education)	1415	3.42%
天气 (Weather)	1307	3.16%
汽车 (Cars)	1255	3.03%
军事 (Military)	738	1.78%
房产 (Real Estate)	614	1.48%
旅游 (Travel)	597	1.44%
三农 (Agriculture)	546	1.32%
文化 (Culture)	435	1.05%
游戏 (Games)	365	0.88%
综艺 (Variety Shows)	363	0.88%
电影 (Movies)	324	0.78%
健康 (Health)	314	0.76%
电视剧 (TV Series)	210	0.51%
历史 (History)	202	0.49%
科学 (Science)	150	0.36%
音乐 (Music)	150	0.36%
生活 (Life)	116	0.28%
美食 (Food)	103	0.25%
情感 (Sentiment)	102	0.25%
Total	41351	100%

Table 6: The topics in Title2EventPhrase with their numbers and proportions of instances.

E Event Filter Rules

In this stage, we introduce the filtering criteria for headlines in the coarse filter phase. Five types of headlines will be excluded, namely: non-event headlines, missing important components related to the event, containing multiple events, irregular syntax, and interrogative sentences. We provide examples of each of these five types in Figure 6.