Event-enhanced Retrieval in Real-time Search

Yanan Zhang, Xiaoling Bai, Tianhua Zhou
Tencent Search, Platform and Content Group
{yananzhang, devinbai, kivizhou}@tencent.com

Abstract

The embedding-based retrieval (EBR) approach is widely used in mainstream search engine retrieval systems and is crucial in recent retrieval-augmented methods for eliminating LLM illusions. However, existing EBR models often face the "semantic drift" problem and insufficient focus on key information, leading to a low adoption rate of retrieval results in subsequent steps. This issue is especially noticeable in real-time search scenarios, where the various expressions of popular events on the Internet make real-time retrieval heavily reliant on crucial event information. To tackle this problem, this paper proposes a novel approach called EER, which enhances real-time retrieval performance by improving the dual-encoder model of traditional EBR. We incorporate contrastive learning to accompany pairwise learning for encoder optimization. Furthermore, to strengthen the focus on critical event information in events, we include a decoder module after the document encoder, introduce a generative event triplet extraction scheme based on prompt-tuning, and correlate the events with query encoder optimization through comparative learning. This decoder module can be removed during inference. Extensive experiments demonstrate that EER can significantly improve the real-time search retrieval performance. We believe that this approach will provide new perspectives in the field of information retrieval. The codes and dataset are available at https://github.com/open-event-hub/Event-enhanced_Retrieval.

Keywords: Real-time search, Event-enhanced, Embedding-based retrieval

1. Introduction

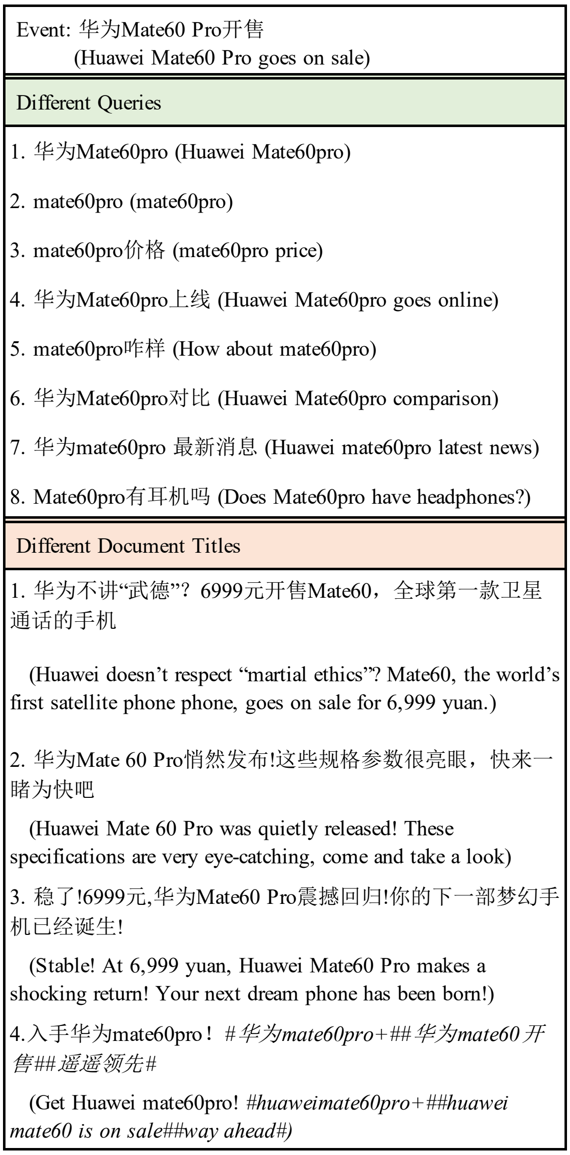
The embedding-based retrieval (EBR) approach has gained attention since its introduction and is widely used in the recall systems of mainstream search engines. It also plays a crucial role in recent methods aimed at mitigating the hallucinations of large language models through retrieval-augmented techniques, with many LLM application frameworks such as Langchain[1] providing such tutorials. Compared to traditional term-level retrieval algorithms like BM25 (Robertson and Zaragoza, 2009), EBR can effectively capture semantic similarity beyond word frequency or term matching, thus better handling synonyms, near-synonyms, and context-related semantic relationships. However, efficiently retrieving the most relevant documents from billions of documents remains a daunting challenge.

One of the main challenges faced by existing EBR models is the "semantic drift" problem, i.e., the semantics of the model encoding deviates from the given context, lacking attention to key information. This problem becomes particularly pronounced in real-time search (Bradley, 2009) scenarios, where users tend to input shorter queries, typically keywords or phrases about an event, to quickly obtain information about the event. On the other hand, there are multiple ways of expressing the same event on the internet, considering different media sources and even self-media. Moreover, documents are generally longer than queries, and even if only considering the title, they still contain a lot of less important information, as shown in Figure 1. The highly asymmetric information between queries and titles makes real-time retrieval of event documents more difficult. Existing research has not paid special attention to the differences and difficulties of real-time search compared to other searches. On the one hand, attempts have been made to improve the embedded representation performance by introducing more massive data and models with larger parameters (Ni et al., 2022b; Su et al., 2022; Wang et al., 2022; Li et al., 2023; Xiao et al., 2023), to achieve a "miracle" effect, which in fact does lead to an improvement, but the pursuit of lower cost and smaller model parameters deserves to be considered all the time. On the other hand, a large number of data augmented approaches (Wei and Zou, 2019; Liu et al., 2021; Wu et al., 2022; Chuang et al., 2022; Tang et al., 2022) are used, such as token duplication, substitution, etc., but these schemes pay little attention to events (the central secret of real-time search) and are not sufficient to cope with the complexity of the scenario.

To address this pressing problem, we propose an event-enhanced retrieval (EER) method, which builds on the traditional EBR dual encoder model that utilizes $< query, title >$ pairs. We introduce various hard negative mining techniques and apply supervised contrastive learning (Gao et al., 2019) to improve the performance of the encoders. To further widen the gap between positive and negative example samples, we also incorporate pairwise learning, enabling the encoder to better focus on

[1] https://github.com/langchain-ai/langchain

LREC-COLING 2024, pages 6594–6606
20-25 May, 2024.



Event: 华为Mate60 Pro开售
(Huawei Mate60 Pro goes on sale)

Different Queries

1. 华为Mate60pro (Huawei Mate60pro)

2. mate60pro (mate60pro)

3. mate60pro价格 (mate60pro price)

4. 华为Mate60pro上线 (Huawei Mate60pro goes online)

5. mate60pro咋样 (How about mate60pro)

6. 华为Mate60pro对比 (Huawei Mate60pro comparison)

7. 华为mate60pro 最新消息 (Huawei mate60pro latest news)

8. Mate60pro有耳机吗 (Does Mate60pro have headphones?)

Different Document Titles

1. 华为不讲“武德”？6999元开售Mate60，全球第一款卫星通话的手机

(Huawei doesn’t respect “martial ethics”? Mate60, the world’s first satellite phone phone, goes on sale for 6,999 yuan.)

2. 华为Mate 60 Pro悄然发布!这些规格参数很亮眼，快来一睹为快吧

(Huawei Mate 60 Pro was quietly released! These specifications are very eye-catching, come and take a look)

3. 稳了!6999元,华为Mate60 Pro震撼回归!你的下一部梦幻手机已经诞生!

(Stable! At 6,999 yuan, Huawei Mate60 Pro makes a shocking return! Your next dream phone has been born!)

4.入手华为mate60pro！ *#华为mate60pro+##华为mate60 开售##遥遥领先#*

(Get Huawei mate60pro! *#huaweimate60pro+##huawei mate60 is on sale##way ahead#)*

Figure 1: An event corresponds to various queries and documents. Most queries are always concise, focusing on the key information of the event, and often contain abbreviations, omissions, grammatical irregularities, etc. For example, in the second query “mate60pro”, “Huawei” is omitted, and “Mate” is entered as “mate”. The document title is lengthy, contains redundant information, and the expression style is diversified. In the third title, the action “稳了 (Steady)” lacks a subject and is an unconventional syntax. The fourth title contains a lot of tags with “#”. It is therefore difficult to relate queries to documents. The data here is from the real world.

relative order and enhance robustness. To address the $< query, title >$ pairs information asymmetry and the abundance of noisy information in titles, we creatively introduce a decoder structure outside the title-side encoder. The decoder aims to receive the encoded title information and extract event triplets through a generative task, facilitating the title-side encoder to focus more on the event information. We also employ keyword-based prompt learning to make the generated content more controllable. The events generated by the decoder also deserve attention from the query, as they represent crucial event information from the title. Therefore, the $< query, generated{-}event >$ pairs interact through supervised contrastive learning to boost the performance of the query encoder. It is worth noting that this decoder is only used in the training step to enhance the understanding of the event, while it can be removed in the inference phase, and EER will revert to the traditional dual-tower model with no impact on latency.

The main contributions of this work are summarized as follows:

- To our knowledge, EER is the first approach that tackles the “semantic drift” issue in real-time search scenarios, aiming to enhance the retrieval of event documents.

- Building upon the traditional dual-tower model, we introduce a generation task specialized for events in titles. By employing loss functions that emphasize the attention of both encoders to the events, we achieve state-of-the-art performance for the encoders.

- Numerous experiments and analyses have indicated the strong merit of EER.

2. Methods

2.1. Overview

In this section, we describe the proposed EER model in detail. As shown in Figure 2(a), the most basic structure of the model is a dual tower that encodes the query and the document title, respectively. Further, we focus on the extraction and usage of document event information, i.e., we add a prompt learning-based decoder module after the encoder of the document and influence the representation performance of both encode modules through the loss feedback, to improve the retrieval ability of the event documents under real-time search. Finally, we mention the difference between the inference and train pipeline. As shown in Figure 2(b), the newly added decode module can be taken off in the inference stage, and the model is restored to the traditional dual towers.

2.2. Encoder

2.2.1. Hard Negative Sampling

Hard negative sampling (Robinson et al., 2021) has become an important method over the years.

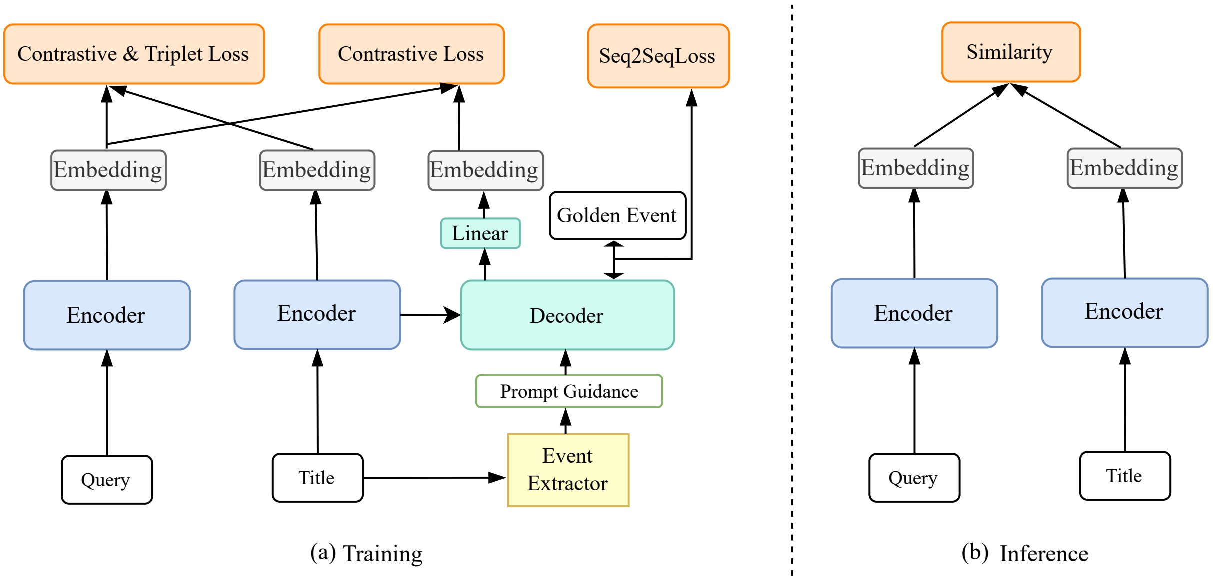


Figure 2: Architecture of the proposed EER model.

Compared to random negative sampling, hard negative sampling is able to pull off the gap between positive and negative samples in a more targeted way. We do hard negative sampling using a knowledge augmentation-based approach and a semantic mining-based approach.

Knowledge Augmentation: To improve the robustness of EER sentence representation, we perform data augmentation of queries and headings following EDA (Wei and Zou, 2019) before feeding pairs of sentences into the encoder. We mainly used three data augmentation strategies, including entity replacement (encyclopedia), random token deletion and replication, and token reordering. We speculate that (1) Using entity replacement is an effective strategy to create semantically similar phrases with different tags, which helps the model capture keyword similarity rather than syntactic similarity. (2) The random deletion strategy can mitigate the impact of frequent words or phrases. (3) The shuffling strategy can reduce the sensitivity of the sentence encoder to position changes.

Semantic Mining: We fine-tune an encoder model based on existing training data to encode all collected titles and store them into a Faiss (Johnson et al., 2019) vector index library. Then, for each query, we retrieve its k neighboring titles using semantic similarity. For these k titles, we randomly keep up to m $(m \ll k)$ of them whose relevance (cosine similarity) is between a predefined upper and lower bound. Through this operation, we get tough but low-relevance negative titles.

2.2.2. Constrastive Learning

To help our model learn sentence representations better and alleviate the problem of vector space collapse, we utilize contrastive learning (Gao et al., 2019) techniques to pull the vector distance closer between a query and its specified positive titles while pushing the distance further between the query and negative titles. The negative titles including randomly sampled and the hard negatives mentioned earlier, mean that the hard negatives of one query will also be shared with other queries, which may further augment the scale of negatives. The comparative learning loss of query and title can be formulated as:

$$\mathcal{L}_{cl_{qt}} = -\log \frac{e^{cos(\mathbf{h}_i, \mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} \left(e^{cos(\mathbf{h}_i, \mathbf{h}_j^+)/\tau} + e^{cos(\mathbf{h}_i, \mathbf{h}_j^-)/\tau} \right)} \tag{1}$$

where $\mathbf{h}_i$, $\mathbf{h}_i^+$ and $\mathbf{h}_i^-$ are the representation of the i-th query, its positive sample and its negative sample respectively, $cos(\cdot)$ is the cosine similarity, τ is a temperature hyper-parameter and N is batch size.

2.2.3. Pairwise Learning

Consider a query and its positive and negative titles are constructed according to the pairwise formula and the ordinal relationship between positive and negative examples is also important, we utilize triplet loss (Wang et al., 2014) to strengthen this kind of relevance ranking. Inspired by Sentence-BERT (Reimers and Gurevych, 2019), the loss can be illustrated as:

$$\mathcal{L}_{pair_{qt}} = \max(0, \epsilon + cos(\mathbf{h}_i, \mathbf{h}_i^-) - cos(\mathbf{h}_i, \mathbf{h}_i^+)) \quad (2)$$

where ϵ is the margin to ensure the similarity of $(\mathbf{h}_i, \mathbf{h}_i^+)$ is at least closer than $(\mathbf{h}_i, \mathbf{h}_i^-)$, which is set as 0.1 to avoid over-fitting.

2.3. Generative Decoder

In this section, we introduce methods to help our model enhance event information awareness by introducing decoder modules and subtasks at the title.

2.3.1. Event Extraction

The event information is one of the core components of a headline. Extracting which phrase directly affects what information we encode in the sentence representation. Given the characteristics and difficulties of Chinese news headlines, we mainly use two existing methods to extract event information: semantic role labeling and dependency syntactic parsing of a sentence. By utilizing LTP toolkit (Che et al., 2021), we can easily obtain the semantic role labeling of a sentence, and extract the subject-predicate-object structure, subject-predicate structure, and predicate-object structure in order. If the semantic role tags are empty, we use dependency syntactic parsing methods to extract event triples centered around the predicate, including subject-predicate-object, post-verb object with attributive, and subject-predicate-verb-object with preposition. Additionally, we use the Title2Event (Deng et al., 2022b) dataset to fine-tune a supervised Seq2SeqMRC model, a pipeline model that replaces the argument extractor with a sequence-to-sequence MRC model using mT5-base (Xue et al., 2021). In practice, we find that the Seq2SeqMRC model performs better than LTP, and we choose it as the event extraction tool.

2.3.2. Generation Learning with Prompt Guidance

We added a decoder module to the original two-tower encoder model to undertake the event information generation subtask. As shown in Figure 2(a), the decoder is another 12-layer RoBERTa (Liu et al., 2019), whose parameters we initialise in the same way as the encoder, inspired by BERT2BERT (Rothe et al., 2020), so that the parameters of the decoder are isomorphic to those of the encoder, reducing maintenance costs. During the training stage, similar to any standard sequence-to-sequence transformer architecture, the original titles are fed into the encoder, and events, marked as E and extracted from the titles, serve as the ground truth and are fed into the decoder.

Further, to ensure that important information is not overlooked, we leverage prompt learning techniques in the generation task. Unlike previous work, we use adaptive keyword templates to guide event generation. One of the keyword templates T is similar to "In `[X]`, the object is `[MASK]`, the trigger is `[MASK]`, and the topic is `[MASK]`", where "`[X]`" is the text of the title. In this form, we mimic the masked language model pre-training task so that the model can perceive the object, subject, and trigger of the event. The final input to the decoder is labeled "`[CLS]`T E`[SEP]`".

The event generation loss is formulated as:

$$\mathcal{L}_{gen} = -\sum_{i=1}^{N}\sum_{t=1}^{T} y_{i,t} \log \hat{y}_{i,t} \quad (3)$$

where the variable N represents the number of samples in the dataset, and T denotes the length of the target sequences. The elements $y_{i,t}$ and $\hat{y}_{i,t}$ are the true and predicted probabilities of the target token at position t in the i-th sample, respectively.

2.3.3. Relevance Learning Between Query and Event

Events generated by the decoder module can be regarded as a "condensed version" of the title because it represents the critical information of the title, has less information noise, and is shorter than the title. Therefore, compared with titles, events can alleviate information asymmetry and are worthy of being used to interact with queries to optimize model performance. Specifically, when a query is related to a title, we consider the query to be related to the event corresponding to that title. We add a sub-task to characterize this similarity, where positive samples are events generated from positive titles, and negative samples are other events in the batch, also learned through contrastive learning loss, which can be formulated as:

$$\mathcal{L}_{cl_{qe}} = -\log \frac{e^{cos(\mathbf{h}_i, \mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N}\left(e^{cos(\mathbf{h}_i, \mathbf{h}_j^+)/\tau} + e^{cos(\mathbf{h}_i, \mathbf{h}_j^-)/\tau}\right)} \quad (4)$$

where $\mathbf{h}_i$, $\mathbf{h}_i^+$ and $\mathbf{h}_i^-$ are the representation of the i-th query, its positive sample and its negative sample respectively. The difference from 2.2.2 is that the samples here are replaced by events.

2.4. Total Loss

In summary, our training task consists of three parts: query-title relevance learning, event generation learning, and query-event relevance learning.

Among them, query-title relevance learning consists of both contrastive learning and pair learning. The loss function can be completely formulated as:

$$\mathcal{L}_{total} = \mathcal{L}_{cl_{qt}} + \mathcal{L}_{pair_{qt}} + \mathcal{L}_{gen} + \mathcal{L}_{cl_{qe}} \quad (5)$$

2.5. Inference Pipeline

Indeed, the purpose of adding a decoder module to the title to implement event extraction and interact with the two encoders is to assist in optimizing the performance of the two encoders. On the other hand, a complex model structure brings inference latency and reduced ease of use, which also needs to be considered. Therefore during the inference stage, we only need vector representations of queries and titles, and the decoder module can be removed, as shown in Figure 2(b). EER reverts to the traditional dual-tower model, which means there is no significant change in time consumption and ease of use.

3. Experiments

3.1. Experimental Settings

3.1.1. Datasets

Considering that there is no retrieval dataset tailored specifically for real-time search, we produce one ourselves and make it publicly available. The dataset is gathered from the massive user logs of Tencent QQ Browser Search, thus the vast majority of data is in Chinese and the authenticity of the data is guaranteed. The query patterns in this data are varied. At the same time, the titles come from various types of documents such as text (including User Generated Content), video (including mini video), etc., covering 23 news categories such as current affairs, sports, finance stock, technology, society, and entertainment. As shown in Figure 1 earlier, the characteristics of query and title are very distinct from the existing benchmark, and the experimental results in Section 3.2.1 later demonstrate the uniqueness of this data.

Specifically, relevant documents are labeled as 1 and irrelevant ones as 0 in the data. For the training data, we sample $< query, title >$ pairs from the user logs of real-time search requests (using an existing intent recognition tool) over the past six months. We start with an initial filtering of the data using some of the features that the logs already contain, such as quality score, authority, and harmfulness and make an effort to remove personal information about private individuals. For each pair, multiple rules are constructed to automatically annotate the dataset using user behaviors such as clicks, document browsing duration, page flipping, and query reformulation, combined with relevance features such as BM25 and term matching rate. This method enables mining a large amount of data from user logs.

Dataset	Queries	Titles	Q-T pairs
Training	2,964,077	5,323,681	10,319,501
Testing	1,096	4,733	10,2279

Table 1: The statistics of the dataset.

To prevent “data leakage”, we use the same sampling method as the training data for the test data, but we choose a different timeframe - specifically, the month following the training data. This helps to ensure that the test data is independent of the training data. Given the enormous size of the log data, the probability of sampling data from the same user is low. Furthermore, we concentrate on event-related queries and documents, which are updated rapidly due to the constant flow of new information. This means that users’ interests can shift quickly, and we need to ensure that our data reflects these changes accurately. Considering the limitations of rule-based annotation, we chose to ensure the data quality through expert annotation on a crowdsourcing platform. We write an 8-page annotation document that includes numerous examples to illustrate the standard. Additionally, we develop an annotation tool with a search function to assist experts in making judgments. Each data is double annotated, and this data will be rechecked when the double annotations are inconsistent to ensure that the final accuracy of the test data reaches 95%. Detailed data statistics are shown in Table 1.

3.1.2. Evaluation Metrics

We adopt Recall@k (R@k) (Jegou et al., 2010), Mean Reverse Ranking (MRR) (Craswell, 2009), and AUC (Fawcett, 2006) to compare models. R@k measures the ratio of queries in which the correct template is within the top-k, while MRR computes the mean reverse of the correct template. Both metrics tend to focus on positional relationships. AUC is used to observe the ability to discriminate between the full range of positive and negative samples. In particular, we track R@10 and MRR@10 as a general indicator of ranking performance.

3.1.3. Baselines and Parameter

We selected several representative baseline methods:

BM25 (Robertson and Zaragoza, 2009) A classic algorithm used to evaluate the relevance between a query and a document. It considers factors such as term frequency, document length, and inverse document frequency to determine the relevance of a document to a given query.

Models	R@10	MRR@10	AUC
BM25	0.579	0.556	0.773
Sentence-BERT	0.693	0.650	0.827
BGE	0.771	0.694	0.915
EER	**0.829**	**0.757**	**0.931**

Table 2: Evaluation of EER and baselines.

Sentence-BERT (Reimers and Gurevych, 2019) Another classic method that uses BERT-based models to generate high-quality sentence embeddings to capture the semantics of sentences. Specifically, we choose roberta-base for initialization.

BGE (Xiao et al., 2023) The state-of-the-art method on MTEB[2], trained on 300 million text pairs of data, also performed well in the retrieval task. To align the vector dimensions, we chose BGE-base for comparison.

We use RoBERTa-base (Liu et al., 2019) models as the backbone for EER training. The encoder and decoder modules of EBB both come with 12 Transformer layers (total 24 layers) and, 768 hidden size. We explore hyperparameters such as batch size, learning rate, and prompt templates. Finally, We train EER with adam as the optimizer, the batch size as 256, and the learning rate as $5e^{-5}$. We search for prompt templates, as is discussed in Section 3.2.3. For Sentence-BERT and BGE models, we use similar parameters to fine-tune the same data[3].

3.2. Results and Study

3.2.1. Overall Results

Table 2 shows the comparison results on our disclosed dataset. Our proposed EER achieves better performance than baseline methods. Additionally, we can make the following three observations, which help to understand real-time retrieval and the advantages of EER.

First, as shown in Figure 1, due to the diversity of expressions of the same popular event on the Internet and the simplicity of the query, the solution based on literal matching is not efficient. Compared with BM25, the performance of semantic-based models is significantly ahead. This simultaneously indicates that the dataset is characterized.

Second, EER goes beyond the two semantic-based baselines to demonstrate the excellent performance of the addition of event extraction - the decoder module on the document side.

Models	R@10	MRR@10	AUC
base	0.687	0.597	0.829
base+CL	0.734	0.628	0.850
base+CL+GD	0.769	0.673	0.884
base+CL+GD+GP	0.786	0.679	0.910
base+CL+GD+QER	0.802	0.704	0.915
EER	0.829	0.757	0.931

Table 3: Evaluation of EER components. CL, GD, GP, and QER are abbreviations for contrastive learning, generative decoder, generative prompt, and relevance learning between query and title, respectively. Base deserves to be Roberta-base.

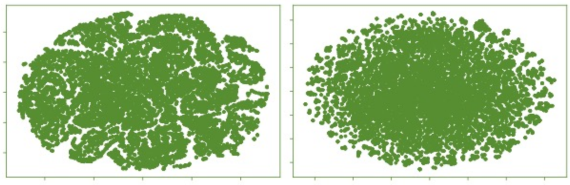


Figure 3: The t-SNE visualization of representations from encoders without and with contrastive learning. As demonstrated in the left part, without contrastive learning, the model encodes queries into a smaller space with more collapses. And on the right, the addition of contrastive learning expands the embedding space with better alignment and uniformity.

Third, the performance of BGE surpassed Roberta, which was trained on 300 million data, demonstrating the importance of larger and better quality data. So we believe that disclosing this real-time search data from a real search engine is meaningful for information retrieval research.

3.2.2. Component Effectiveness Study

In this section, we discuss further the effectiveness of each component in our model. We make comparisons with the baseline by adding only one component at each time. The results of the experiment are illustrated in Table 3.

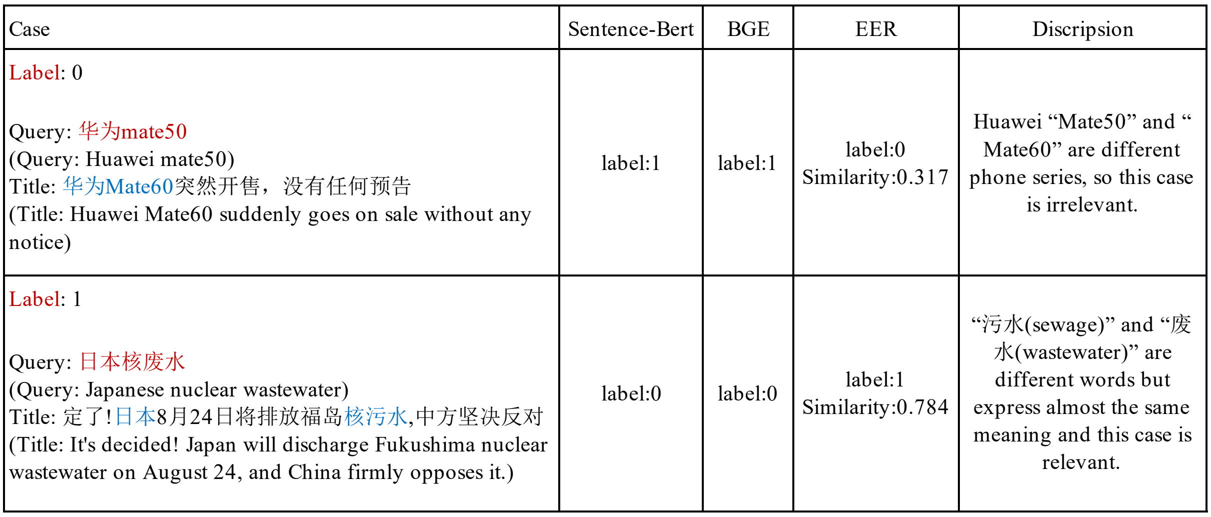
Contrastive Learning Supervised contrastive learning technique is adopted to alleviate representation space degradation problems (Gao et al., 2019). To validate its effect, we illustrated the performance in Line 2 (RoBERTa+CL) of Table 3, where all three metrics, recall, MRR, and AUC, grow significantly compared to the base model. Additionally, to demonstrate that our proposed method can address the issue of representation degradation, we visualize the results of a two-dimensional t-SNE (Van der Maaten and Hinton, 2008) graph on the embedding of 100,000 queries, which is depicted in Figure 3 and provides further evidence to

[2] https://huggingface.co/spaces/mteb/leaderboard

[3] We use a popular version of the Chinese Sentence-BERT model available at `https://huggingface.co/DMetaSoul/sbert-chinese-general-v2/tree/main`, and the Chinese version of BGE-base available at `https://huggingface.co/BAAI/bge-base-zh`.

Templates	R@10	MRR@10	AUC
In [X], the subject is [MASK]	0.817	0.739	0.922
In [X], the subject is [MASK], the object is [MASK], the action is [MASK]	**0.829**	**0.757**	**0.931**
[X] v_1...[MASK]...v_n	0.798	0.726	0.904
[X] v_1...[MASK][MASK][MASK]...v_n	0.819	0.736	0.919

Table 4: Performance of different prompt templates.



Case	Sentence-Bert	BGE	EER	Discripsion
Label: 0 Query: 华为mate50 (Query: Huawei mate50) Title: 华为Mate60突然开售，没有任何预告 (Title: Huawei Mate60 suddenly goes on sale without any notice)	label:1	label:1	label:0 Similarity:0.317	Huawei “Mate50” and “ Mate60” are different phone series, so this case is irrelevant.
Label: 1 Query: 日本核废水 (Query: Japanese nuclear wastewater) Title: 定了!日本8月24日将排放福岛核污水,中方坚决反对 (Title: It's decided! Japan will discharge Fukushima nuclear wastewater on August 24, and China firmly opposes it.)	label:0	label:0	label:1 Similarity:0.784	“污水(sewage)” and “废水(wastewater)” are different words but express almost the same meaning and this case is relevant.

Figure 4: Typical case demonstration of EER and baseline. Relevant query-title pairs are marked as 1 and irrelevant ones as 0.

support our conclusion.

Event Generative To make the model implicitly focus on event information, a decoder module is added for event generative learning. As seen in the third line, we learn that decoder (base+CL+GD) brings 4.8% R@10, and 7.1% MRR@10 increments on top of the second line. The improvement in metrics in this step is significant.

Prompt Guide The fourth line shows that models within the prompt technique get a bit better performance than without versions. Keyword-based prompt learning proves its appeal.

Interaction Between Query and Event The fifth row in Table 3 shows that we focused on the impact of adding correlation learning between queries and events (the results generated by the decoder module) without considering prompt learning (which would be the full EER model if it were added). The results are obvious - the direct interaction of queries and events effectively contributes to encoder performance, with metrics improving, compared to the third row.

3.2.3. Prompt Search

We try to find suitable prompt templates and mainly explore two different types of templates: hand-craft and continuous. Prompt search experimental results are shown in Table 4. Among them, the template “In `[X]`, the subject is `[MASK]`, the object is `[MASK]`, the action is `[MASK]`” performs best. We analyzed that longer and more specific templates strongly imply the key information of a given title, i.e. what the event is, and therefore this hand-craft template is more competitive.

3.3. Case Study

To visually illustrate how EER works, we list two typical cases in Figure 4 for qualitative analysis.

In the first case, since the rest of the words in the query are included in the title except for the term “50”, both semantic-based baselines consider the case as relevant, appearing similar to the BM25 algorithm without capturing the huge semantic inconsistency caused by the subtle differences between the terms. In contrast, EER can focus on the fact that the subject of the event in the title is inconsistent with the query and thus makes a distinction.

In the second case, there are synonym pairs such as “废水 (sewage)” and “污水 (wastewater)”, which the semantic-based model should have taken advantage of. However, the information about the query in the title is very dispersed (not continuous but scattered), and there is also redundant infor-

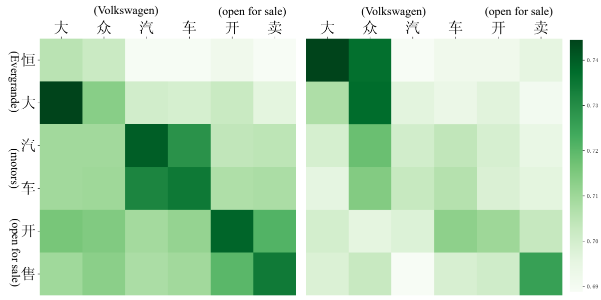


Figure 5: Distribution of Roberta (a) and our method (b).

mation "中方坚决反对 (China firmly opposes it)" to form interference, which leads to very asymmetric information between the query and the title. Baselines did not make a correct judgment. On the other hand, the event generated by the title is ["日本 (Japan)", "将排放 (will discharge)", "福岛核污水 (Fukushima nuclear sewage)"]. Shorter, more focused information facilitates matching to the query, with EER labeled correctly.

Attention Distribution To verify the fusion effect of the decoder module, we plot the attention weight distribution of EER compared to Roberta. As shown in Figure 5(a), Roberta is more likely to focus on the matching of similar tokens and underestimate the inconsistent parts. In contrast, in Figure 5(b), with the help of the decoder module, the attention distribution becomes more reasonable, especially the weight between "恒大(Evergrande)" and "大众(Volkswagen)" increases significantly. This indicates that EER simultaneously emphasizes different parts of the sentence pair.

4. Related Work

4.1. Information Retrieval

Realtime Search Information retrieval (IR) is a classic NLP task and is widely used in a variety of information-sharing scenarios - after all, people always need to find information. From the perspective of commercial search engine functionality, information retrieval can be categorized into various modes such as knowledge retrieval, product retrieval, code retrieval, and so on (Sølvberg et al., 1992; li et al., 2004; Sachdev et al., 2018; Zhang et al., 2023). Among these modes, some have been extensively and deeply explored by many researchers, while others have not. For example, real-time retrieval has not received much attention. Many events are happening all over the world every moment and are being reported and shared, for example, on Twitter there are hundreds of thousands of tweets every second (Busch et al., 2012), which makes real-time retrieval very important and used to satisfy the attention of users on new events.

Retrieval Paradigm For a long time, researchers have done a lot of exploration. For example, the classical unsupervised approach BM25 mainly focuses on the degree of lexical matching to respond to the match between the query and the document. Neural network models have also been widely used in information retrieval. DSSM (Huang et al., 2013) uses a deep learning network to map query and document into a semantic space of the same dimension, thus obtaining a low-dimensional semantic vector representation of the utterance sentence embedding, which is used to predict the semantic similarity of two sentences. Poly-encoder (Humeau et al., 2019) employs multiple independent encoders, each focusing on processing different information, to solve the bi-encoder's low matching quality problem and the slow matching speed of interactive cross-encoders such as ARC-II. The subsequent Colbert (Khattab and Zaharia, 2020) structure is relatively streamlined, introducing a late interaction architecture. By delaying and preserving this fine-grained similarity, the ability to pre-compute document representations offline is gained, greatly speeding up queries. Yang et al. (2023) proposes to use event extensions to assist retrieval, requiring additional information to be added.

Sentence Embedding Improved NLU technology leads to better sentence representation, which is crucial for information retrieval using vector representations. Sentence-BERT (Reimers and Gurevych, 2019), by using specific fine-tuning techniques, can generate semantically rich sentence embedding representations that achieve better performance in tasks such as text matching. Sentence-T5 (Ni et al., 2022a) adds the decode module for sentence embedding, exploring a variety of representations. Su et al. (2023) tries to let the model generate sentence vectors suitable for downstream tasks by giving different instructions to the model, which improves the performance of sentence embedding through more diversified data. In the era of large language models, there have also been some explorations (Jiang et al., 2023) of sentence embedding representations with generative models, however, due to the difference between NLU and NLG, related work is still in its infancy. It is important to note that models with large numbers of parameters together with huge amounts of data can cost a lot of money and cause more carbon emissions, so lightweight and low-cost modeling studies are still of great practical relevance.

4.2. Event Extraction

Event extraction (Hogenboom et al., 2011) is the task of organizing natural text into structured events, that is, extracting specific events that occurred at a specific time and place and involved one or more actors, each associated with a set of attributes.

Traditional methods (Ji and Grishman, 2008; Hong et al., 2011; Li et al., 2013) rely on human-designed features and rules to extract events. The event extraction model based on neural networks (Nguyen and Grishman, 2015; Nguyen et al., 2016) uses multiple model paradigms for modeling through automatic feature learning. Among them, the most common classification-based method considers event extraction as classifying given trigger and argument candidates into different labels (Feng et al., 2016; Liu et al., 2018; Lai et al., 2020; Wang et al., 2021). Another sequence tagging method (Chen et al., 2018; Ding et al., 2019; Ma et al., 2020; Guzman-Nateras et al., 2022) performs EE by tagging each word according to a specific tagging pattern such as BIO (Ramshaw and Marcus, 1995). With the research on machine reading comprehension tasks, the EE task paradigm has also been transformed into MRC to solve (Wei et al., 2021; Zhou et al., 2022). This approach employs a span prediction paradigm to predict event triggers and the start and end positions of argument spans. In addition, there are also some works (Huang et al., 2022; Zeng et al., 2022) exploring generating EE result sequences through conditional generation models as well as combining MRC and generative tasks (Deng et al., 2022a). With the development of huge language models, some studies (Gao et al., 2023; Wei et al., 2023) have also discussed the performance of ChatGPT on EE.

5. Conclusion

This paper describes an embedding-based approach, called EER, designed to improve semantic retrieval performance in real-time search. By uniquely utilizing a generative decoder module, our model provides a deeper understanding of the event information implicit in documents, thus enhancing query event matching and significantly reducing the "semantic drift" problem faced in real-time search. We have conducted extensive experiments and analysis to demonstrate the effectiveness of EER. Meanwhile, compared with the currently widely deployed models in real-world scenarios, our model does not bring additional costs because the model parameters are unchanged in the inference stage. Recently LLM has made a big splash in retrieval with its excellent performance, while the high inference cost constrains the wide application of LLM, and of course there are some ongoing works exploring the cost reduction. We believe that our proposed method will bring more thinking perspectives to the field of information retrieval at present.

6. Bibliographical References

Phil Bradley. 2009. Search engines: Real-time search.

Michael Busch, Krishna Gade, Brian Larson, Patrick Lok, Samuel Luckenbill, and Jimmy Lin. 2012. Earlybird: Real-time search at twitter. In *2012 ieee 28th international conference on data engineering*, pages 1360–1369. IEEE.

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.

Yubo Chen, Hang Yang, Kang Liu, Jun Zhao, and Yantao Jia. 2018. Collective event detection via a hierarchical and bias tagging networks with gated multi-level attention mechanisms. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1267–1276, Brussels, Belgium. Association for Computational Linguistics.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljacic, Shang-Wen Li, Scott Yih, Yoon Kim, and James Glass. 2022. DiffCSE: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle, United States. 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. 2022a. 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.

Haolin Deng, Yanan Zhang, Yangfan Zhang, Wangyang Ying, Changlong Yu, Jun Gao, Wei Wang, Xiaoling Bai, Nan Yang, Jin Ma, et al. 2022b. 2event: 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.

Ning Ding, Ziran Li, Zhiyuan Liu, Haitao Zheng, and Zibo Lin. 2019. Event detection with trigger-aware lattice neural network. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 347–356, Hong Kong, China. Association for Computational Linguistics.

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

Xiaocheng Feng, Lifu Huang, Duyu Tang, Heng Ji, Bing Qin, and Ting Liu. 2016. A language-independent neural network for event detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 66–71, Berlin, Germany. Association for Computational Linguistics.

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, Huan Zhao, Changlong Yu, and Ruifeng Xu. 2023. Exploring the feasibility of chatgpt for event extraction. *arXiv preprint arXiv:2303.03836*.

Luis Guzman-Nateras, Minh Van Nguyen, and Thien Nguyen. 2022. Cross-lingual event detection via optimized adversarial training. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5588–5599, Seattle, United States. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Frederik Hogenboom, Flavius Frasincar, Uzay Kaymak, and Franciska De Jong. 2011. An overview of event extraction from text. *DeRiVE@ ISWC*, pages 48–57.

Yu Hong, Jianfeng Zhang, Bin Ma, Jianmin Yao, Guodong Zhou, and Qiaoming Zhu. 2011. Using cross-entity inference to improve event extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1127–1136, Portland, Oregon, USA. Association for Computational Linguistics.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. *Advances in neural information processing systems*, 27.

Kuan-Hao Huang, I-Hung Hsu, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. Multilingual generative language models for zero-shot cross-lingual event argument extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4633–4646, Dublin, Ireland. Association for Computational Linguistics.

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.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *International Conference on Learning Representations*.

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.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *Proceedings of ACL-08: HLT*, pages 254–262, Columbus, Ohio. Association for Computational Linguistics.

Ting Jiang, Shaohan Huang, Zhongzhi Luan, Deqing Wang, and Fuzhen Zhuang. 2023. Scaling sentence embeddings with large language models. *arXiv preprint arXiv:2307.16645*.

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

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 39–48, New York, NY, USA. Association for Computing Machinery.

Viet Dac Lai, Tuan Ngo Nguyen, and Thien Huu Nguyen. 2020. Event detection: Gate diversity and syntactic importance scores for graph convolution neural networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5405–5411, Online. Association for Computational Linguistics.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 73–82, Sofia, Bulgaria. Association for Computational Linguistics.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning.

Zhanjun li, Min Liu, and Karthik Ramani. 2004. Review of product information retrieval: Representation and indexing. volume 4.

Fangyu Liu, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2021. Fast, effective, and self-supervised: Transforming masked language models into universal lexical and sentence encoders. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1442–1459, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xiao Liu, Zhunchen Luo, and Heyan Huang. 2018. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256, Brussels, Belgium. 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*.

Jie Ma, Shuai Wang, Rishita Anubhai, Miguel Ballesteros, and Yaser Al-Onaizan. 2020. Resource-enhanced neural model for event argument extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3554–3559, Online. Association for Computational Linguistics.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 300–309, San Diego, California. Association for Computational Linguistics.

Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 365–371, Beijing, China. Association for Computational Linguistics.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022a. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, and Yinfei Yang. 2022b. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and Rabab Ward. 2016. Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(4):694–707.

Lance Ramshaw and Mitch Marcus. 1995. Text chunking using transformation-based learning. In *Third Workshop on Very Large Corpora*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

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

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021. Contrastive learning with hard negative samples. In *International Conference on Learning Representations (ICLR)*.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2020. Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, 8:264–280.

Saksham Sachdev, Hongyu Li, Sifei Luan, Seohyun Kim, Koushik Sen, and Satish Chandra. 2018. Retrieval on source code: a neural code search. pages 31–41.

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*, pages 101–110.

Ingeborg Sølvberg, Inge Nordbø, and Agnar Aamodt. 1992. Knowledge-based information retrieval. *Future Generation Computer Systems*, 7(4):379–390.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. One embedder, any task: Instruction-finetuned text embeddings.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

Zilu Tang, Muhammed Yusuf Kocyigit, and Derry Tanti Wijaya. 2022. AugCSE: Contrastive sentence embedding with diverse augmentations. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 375–398, Online only. Association for Computational Linguistics.

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

Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. 2014. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1393.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Ziqi Wang, Xiaozhi Wang, Xu Han, Yankai Lin, Lei Hou, Zhiyuan Liu, Peng Li, Juanzi Li, and Jie Zhou. 2021. CLEVE: Contrastive Pre-training for Event Extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6283–6297, Online. Association for Computational Linguistics.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Kaiwen Wei, Xian Sun, Zequn Zhang, Jingyuan Zhang, Guo Zhi, and Li Jin. 2021. Trigger is not sufficient: Exploiting frame-aware knowledge for implicit event argument extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4672–4682, Online. Association for Computational Linguistics.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, et al. 2023. Zero-shot information extraction via chatting with chatgpt. *arXiv preprint arXiv:2302.10205*.

Xing Wu, Chaochen Gao, Liangjun Zang, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2022. ESimCSE: Enhanced sample building method for contrastive learning of unsupervised sentence embedding. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3898–3907, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding.

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.

Nan Yang, Shusen Zhang, Yannan Zhang, Xiaoling Bai, Hualong Deng, Tianhua Zhou, and Jin Ma. 2023. Event-driven real-time retrieval in web search. *arXiv preprint arXiv:2312.00372*.

Qi Zeng, Qiusi Zhan, and Heng Ji. 2022. EA^{2}E: Improving consistency with event awareness for document-level argument extraction. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2649–2655, Seattle, United States. Association for Computational Linguistics.

Yanan Zhang, Weijie Cui, Yangfan Zhang, Xiaoling Bai, Zhe Zhang, Jin Ma, Xiang Chen, and Tianhua Zhou. 2023. Event-centric query expansion in web search. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 464–475, Toronto, Canada. Association for Computational Linguistics.

Jie Zhou, Qi Zhang, Qin Chen, Qi Zhang, Liang He, and Xuanjing Huang. 2022. A multi-format transfer learning model for event argument extraction via variational information bottleneck. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1990–2000, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.