End-to-End Deep Attentive Personalized Item Retrieval for Online Content-sharing Platforms

Jyun-Yu Jiang*
Department of Computer Science, University of California, Los Angeles, CA, USA
jyunyu@cs.ucla.edu

Tao Wu
Google Inc., Mountain View, CA, USA
iotao@google.com

Georgios Roumpos
Google Inc., Mountain View, CA, USA
roumposg@google.com

Heng-Tze Cheng
Google Inc., Mountain View, CA, USA
hengtze@google.com

Xinyang Yi
Google Inc., Mountain View, CA, USA
xinyang@google.com

Ed Chi
Google Inc., Mountain View, CA, USA
edchi@google.com

Harish Ganapathy
Google Inc., Mountain View, CA, USA
gharish@google.com

Nitin Jindal
Google Inc., Mountain View, CA, USA
nitinjindal@google.com

Pei Cao
Google Inc., Mountain View, CA, USA
pei@google.com

Wei Wang
Department of Computer Science, University of California, Los Angeles, CA, USA
weiwang@cs.ucla.edu

ABSTRACT

Modern online content-sharing platforms host billions of items like music, videos, and products uploaded by various providers for users to discover items of their interests. To satisfy the information needs, the task of effective item retrieval (or item search ranking) given user search queries has become one of the most fundamental problems to online content-sharing platforms. Moreover, the same query can represent different search intents for different users, so personalization is also essential for providing more satisfactory search results. Different from other similar research tasks, such as ad-hoc retrieval and product retrieval with copious words and reviews, items in content-sharing platforms usually lack sufficient descriptive information and related meta-data as features. In this paper, we propose the end-to-end deep attentive model (EDAM) to deal with personalized item retrieval for online content-sharing platforms using only discrete personal item history and queries. Each discrete item in the personal item history of a user and its content provider are first mapped to embedding vectors as continuous representations. A query-aware attention mechanism is then applied to identify the relevant contexts in the user history and construct the overall personal representation for a given query. Finally, an extreme multi-class softmax classifier aggregates the representations of both query and personal item history to provide personalized search results. We conduct extensive experiments on a large-scale real-world dataset with hundreds of million users from a large video media platform at Google. The experimental results demonstrate that our proposed approach significantly outperforms several competitive baseline methods. It is also worth mentioning that this work utilizes a massive dataset from a real-world commercial content-sharing platform for personalized item retrieval to provide more insightful analysis from the industrial aspects.

KEYWORDS

Item retrieval, personalization, attention mechanism, online content-sharing platforms, real-world log analysis.

ACM Reference Format:
Jyun-Yu Jiang, Tao Wu, Georgios Roumpos, Heng-Tze Cheng, Xinyang Yi, Ed Chi, Harish Ganapathy, Nitin Jindal, Pei Cao, and Wei Wang. 2020. End-to-End Deep Attentive Personalized Item Retrieval for Online Content-sharing Platforms. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3366423.3380051

1 INTRODUCTION

Nowadays, online content-sharing platforms, such as music streaming system, photo and video sharing platform, and online e-commerce, have already become one of the most indispensable media in our lives [6]. However, enormous amounts of users are also accompanied with myriad uploaded contents. To ease the burden of discovering suitable contents from copious corpora, item search becomes one of the most essential functions to derive relevant items to certain queries and satisfy users' information needs.

Compared to ad-hoc search tasks [28], queries in item retrieval are usually short and vague while the user search intents can be

*Work done while interning at Google.

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380051

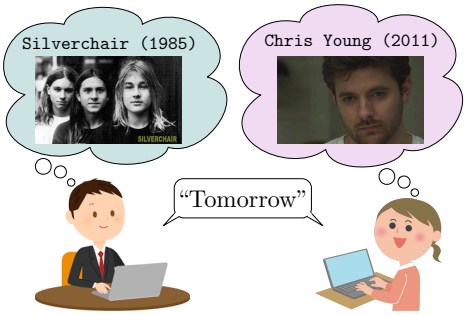

Figure 1: An example that users have different search intents with the same query for item retrieval.

more ambiguous. Figure 1 illustrates an example of an ambiguous query and distinct search intents for two users. More specifically, a certain query can be simultaneously relevant to multiple items while users can have distinct search intents for the query. Hence, the search results should be personalized for different users with their information needs. Moreover, the shortage of descriptive information for both users and items further increases the difficulty of personalization. Hence, personalized item retrieval remains an important research problem, especially for commercial platforms with millions of daily users.

One of the feasible solutions for personalized retrieval is to exploit user history because user historical behaviors can reveal user interests while the idea has already been studied in some research tasks, such as product search and ad-hoc search. For example, previous studies [2, 3, 16] summarize the reviews of purchased products into continuous user features for personalized product search. The words in clicked documents can be utilized to recognize user search intents for personalized ad-hoc search [5, 9, 34, 36]. However, there are few existing studies that address personalized item retrieval with user history while most of the previous works focus on utilizing descriptive document contents. Moreover, personalized ad-hoc and product search methods require descriptive information like reviews that are usually unavailable for items in content-sharing platforms. Hence, although existing methods have demonstrated the effectiveness of descriptive information, it is necessary for personalized item retrieval to obtain user features without any additional information of items in user history.

Without any descriptive information, machine learning models require to learn continuous representations for items as the inputs. To derive decent item representations, some previous studies [14] propose to pre-train the item embeddings and then learn the retrieval models with fixed item representations. However, pre-trained embeddings and multi-stage approaches lead to several drawbacks for item retrieval in real-world content-sharing platforms. First, a massive amount of new items are uploaded to the system every day so that the pre-training needs to be frequently started over to avoid cold-start problems. Second, fixed item embeddings degenerates the retrieval model so that the item representations cannot be flexibly optimized with queries and the objective of the retrieval task. Last but not least, multi-stage approaches can be too complicated to be integrated into sophisticated real-world production pipelines with numerous components. Therefore, an end-to-end approach for item retrieval is required for industrial content-sharing platforms.

Table 1: Comparisons between different personalized search tasks.

Personalized Task	Descriptive Information	Meta Information
Ad-hoc Search	✓ (documents)	✗
Web Search	✓ (web pages)	✗
Microblog Search	✓ (tweets)	✓ (hashtags)
Product Search	✓ (product reviews)	✓ (categories)
Item Search	✗	✓ (content providers)

In this paper, we propose the end-to-end deep attentive model (EDAM) to address the problems of personalized item retrieval for online content-sharing platforms. Without any descriptive information, we learn a continuous representation for each item and each content provider so that the query-aware attention mechanism can derive historical item and content provider representations from personal item history. In addition, we propose to utilize external key embeddings for estimating item attention weights in a different latent space. The sequential knowledge in user history can be also learned from preserving item locality with context items. Experiments on a large-scale dataset from a real-world commercial content-sharing platform demonstrate that EDAM significantly outperforms conventional baseline methods in related personalized search tasks across different evaluation metrics and history lengths.

In the literature, although none of the previous studies focuses on personalized item retrieval for online content-sharing platforms, personalized product search [2, 3, 16] is one of the most related tasks that consider using descriptive information like product reviews. More precisely, the descriptive information can interpret and link both users and products. The structured information [11, 12, 27, 32, 42] and context images [13] can be also applied into personalization. However, all of them rely on descriptive information. Some studies [4, 20, 23, 41] conduct feature engineering and learn a separate ranking model. Personalized listing search [14, 17] is also related to our work, but they highly count on heterogeneous meta data and pre-trained embeddings. In addition to personalized product search, personalization in ad-hoc search [5, 9, 18, 29, 34, 36–38] and microblog search [31, 40] are also relevant to personalized item retrieval. However, all of the existing methods require descriptive information while some models need to be separately learned. Table 1 summarizes the comparisons between different personalized search tasks. Item recommendation with queries [7, 8, 10, 25, 26, 33, 35] and neural information retrieval [15, 21, 30, 43] can be also treated as related tasks to this work.

Our contributions can be summarized as:

- To the best of our knowledge, this work is the pioneer of personalized item search retrieval with queries for online content-sharing platforms without considering any descriptive information. In addition, this is the first study using the datasets of real-world commercial media-sharing platforms for experiments.
- We propose the end-to-end deep attentive model (EDAM) for personalized search item retrieval. The embeddings of both items and their providers can be appropriately learned from user history, thereby deriving historical representations with query-aware

attention mechanism and external item key embeddings. The sequential knowledge in user history can be also learned by locality preservation. In addition, the proposed end-to-end framework can be easily integrated into real-world production systems.
- Experiments were conducted on a dataset from one of the largest online content-sharing platforms. The experimental results indicate that EDAM significantly outperforms several conventional baselines. An in-depth analysis also shows the robustness of EDAM and its components.

2 END-TO-END DEEP ATTENTIVE MODEL FOR PERSONALIZED ITEM RETRIEVAL

In this section, we first formally define the objective of this paper, and then introduce our proposed approach, end-to-end deep attentive model (EDAM) to address the task of personalized item retrieval for online content-sharing platforms.

2.1 Problem Statement

In this paper, we focus on personalized item retrieval using only query and personal item history like watched videos and listened musics. Suppose that $\boldsymbol{V}$ and $\boldsymbol{C}$ are the corpora of items and content providers, where the content provider c of an item v is denoted as $C(v) \in \boldsymbol{C}$. Each query q is composed of a set of terms $T(q) = \{t_1, \cdots, t_{|q|}\}$, where t_i is the i-th term of q; $|q|$ is the number of terms in q. The profile of a user u can be represented by the personal item history as a set of accessed items $H^V(u) \subset \boldsymbol{V}$ and the set of corresponding content providers $H^C(u) = \{C(v) \mid v \in H^V(u)\} \subset \boldsymbol{C}$. For a user u and a query q, $R(q,u) \subset \boldsymbol{V}$ indicates the corresponding set of items that are relevant to the query. Given a user u and a query q, our goal is to rank all of the items in $\boldsymbol{V}$ so that the relevant items $R(q,u)$ can be ranked as high as possible. Note that the task is extremely difficult because only personal item history is available while none of meta-data and descriptive information is granted for items and content providers.

2.2 Framework Overview

Figure 2 shows the illustration of the proposed framework personalized item retrieval with user history for online content-sharing platforms. Items and content providers in the user history are first mapped to item embeddings and provider embeddings while the query embedding is derived by aggregating the embeddings of query terms. With the query-aware attention mechanism, we compute the importance of each provider and each item, thereby obtaining the ultimate representations of historical items and content providers. In addition, we propose to utilize external item key memory for better estimation of item importance. Finally, after aggregating the representations of query and user history, the personalized search results can be derived by a softmax function over the candidate items. Moreover, the item key embeddings can be improved by an auxiliary classification task with query embeddings while the sequential knowledge in user history can be learned by locality preservation for item and provider embeddings.

2.3 Query-aware Attention with External Key Memory for User History Modeling

To utilize the knowledge in the user history, we propose query-aware attention with external key memory to model user history. More precisely, an embedding-based model derives continuous representations in latent spaces for history items and content providers.

Query Embedding. For the given query, we derive a continuous bag-of-terms representation as the query embedding by aggregating term embeddings due to the production efficiency. Formally, the query embedding $\boldsymbol{q} \in \mathbb{R}^d$ of the query q can be computed as:

$$\boldsymbol{q} = \frac{\sum_{t_i \in T(q)} \boldsymbol{t_i}}{|q|},$$

where $\boldsymbol{t_i} \in \mathbb{R}^{d_T}$ is the d-dimensional embedding of the term t_i in q. Note that the query embedding method can be simply replaced with other approaches, but we focus on user history modeling in this paper. In addition, bag-of-terms approach is robust for rare queries in real-world production systems and utilized in various previous studies [3].

User History Modeling with Query-aware Attention. As shown in previous studies [2, 3], user history can be useful for personalization. However, many of the activities in user history can be irrelevant to user search intents. In the item retrieval task, the query plays one of the most essential roles and directly represents search intents of the user. Hence, we utilize the query information for user history modeling with query-aware attention. More specifically, two continuous representations are derived to indicate relevant items and content providers in the user history.

Take the historical item representation as an example. In this paper, we estimate the importance of each item in the user history with the scaled dot-product attention [39]. For each item in user history $v \in H^V(u)$, the attention weight $\alpha(v,q)$ as the importance with the query q can be computed as follows:

$$\alpha(v, q) = \frac{\exp\left(\boldsymbol{q}^T \boldsymbol{v}/\sqrt{d}\right)}{\sum_{v' \in H^V(u)} \exp\left(\boldsymbol{q}^T \boldsymbol{v'}/\sqrt{d}\right)},$$

where $\boldsymbol{q}$ and $\boldsymbol{v}$ are d-dimensional query and item embeddings. The historical item representation $\boldsymbol{h_V}$ can then be represented as the weighted sum of individual item embeddings as $\boldsymbol{h_V} = \sum_{v \in H^V(u)} \alpha(v,q) \cdot \boldsymbol{v}$. Similarly, the representation for historical content providers $\boldsymbol{h_C}$ can be derived as follows:

$$\boldsymbol{h_C} = \sum_{c \in H^C(u)} \alpha(c, q) \cdot \boldsymbol{c}, \text{ and } \alpha(c, q) = \frac{\exp\left(\boldsymbol{q}^T \boldsymbol{c}/\sqrt{d}\right)}{\sum_{c' \in H^V(u)} \exp\left(\boldsymbol{q}^T \boldsymbol{c'}/\sqrt{d}\right)},$$

where $\boldsymbol{c}$ is the embedding for the content provider c.

External Key Embeddings for Item Attention. Generally, the query-aware attention projects historical items and content providers onto the latent query embedding space so that the embedding similarity can be treated as the importance scores. However, the query embedding space can be inappropriate to represent items and content providers. Moreover, the embedding spaces of different entities can be different. Although some studies [2, 3] apply non-linear transformations to cast embeddings into the same space for estimating attention weights, it can be better to independently model representations and estimate attention weights.

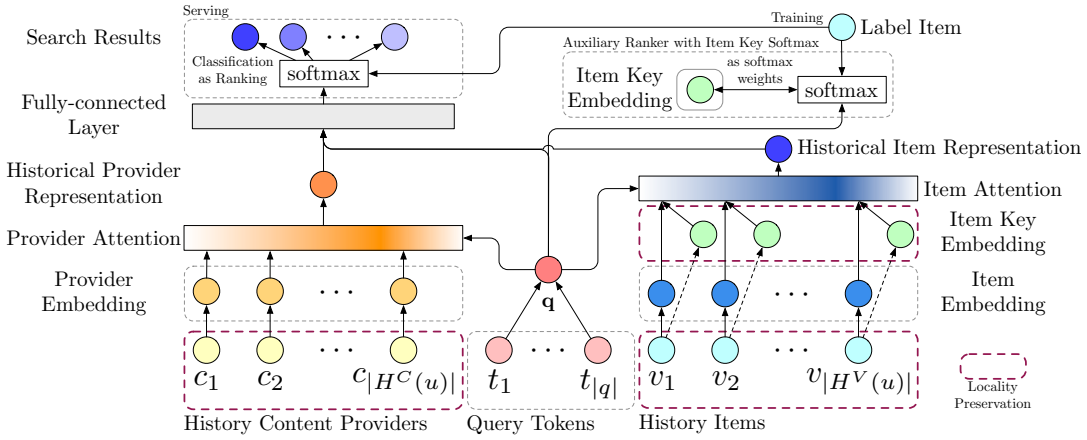

Figure 2: Illustration of the proposed framework, end-to-end deep attentive model (EDAM), for personalized item retrieval.

In this work, we propose to use additional external key embeddings for estimating item attention weights. For each item $v \in \boldsymbol{V}$, instead of utilizing the item embedding $\boldsymbol{v}$, we independently learn an external key embedding $\boldsymbol{k_v}$ in the query embedding space to compute the attention weight as follows:

$$\alpha_k(v, q) = \frac{\exp\left(\boldsymbol{q}^T \boldsymbol{k_v}/\sqrt{d}\right)}{\sum_{v' \in H^V(u)} \exp\left(\boldsymbol{q}^T \boldsymbol{k_{v'}}/\sqrt{d}\right)}.$$

Therefore, the historical item representation $\boldsymbol{h_V}$ can be re-written as $\boldsymbol{h_V} = \sum_{v \in H^V(u)} \alpha_k(v, q) \cdot \boldsymbol{v}$. Note that we do not learn external key embeddings for content providers because they do not lead to improvements as discussed in Section 3.

Finally, to capture the knowledge of both query and user history and model their interactions, the ultimate features $\boldsymbol{h}$ for deriving search results can be computed with a fully-connected layer $\boldsymbol{h} = \text{ReLU}(\boldsymbol{W_h h_0} + \boldsymbol{b_h})$, where $\boldsymbol{h_0} = [\boldsymbol{q}; \boldsymbol{h_V}; \boldsymbol{h_C}]$ concatenates the query embedding and the representations of items and content providers in user history; $\boldsymbol{W_h}$ and $\boldsymbol{b_h}$ represent the layer weights and biases for d_h hidden units; $\text{ReLU}(\cdot)$ is the rectified linear unit as the activation function.

2.4 Classification as Ranking

To derive the ranking results, we follow the previous industrial approach [8] to pose the ranking problem as a task of extreme multi-class classification with the ultimate features $\boldsymbol{h}$. More precisely, given a query q and a user u, we aim to calculate a probabilistic score $P(v \mid q, u)$ for each candidate item $v \in \boldsymbol{V}$ as the estimated relevance to the query.

Given the ultimate features $\boldsymbol{h}$, we can derive the logits $\boldsymbol{x} \in \mathbb{R}^{|\boldsymbol{V}|}$ for multi-class classification with a fully-connected layer as $\boldsymbol{x} = \boldsymbol{W_s h}$, where $\boldsymbol{W_s}$ represents the weights for obtaining logits. Finally, the relevance scores $P(v \mid q, u)$ can be computed with a softmax function as:

$$P(v \mid q, u) = \frac{\exp(x_v)}{\sum_{v'} \exp(x_{v'})},$$

where x_v is the corresponding logit in $\boldsymbol{x}$ for the item v. The search results can then be generated by ranking the relevance scores. Note that we do not learn biases for computing the logits because the search results can be approximated by nearest neighbor search for the efficiency of serving real-world production systems. It is also consistent with existing industrial approaches [8].

2.5 Auxiliary Ranker with Item Key Softmax

When item key embeddings are crucial for estimating the importance of each item in user history, they can only be jointly and implicitly optimized with each other through complicated computations for item attention. To learn better item key embeddings, we propose an additional auxiliary ranker for regularization.

Since item key embeddings share the same latent space with query embeddings, item key embeddings can be also relatively applied for estimating relevance. The auxiliary task aims to estimate the relevance scores $P(v \mid q, \{\boldsymbol{k_i}\})$ with only the query q and the item key embeddings of all items $\{\boldsymbol{k_i}\}$. Here we propose the item key softmax to address the auxiliary task and sharpen item key embeddings. Formally, the relevance score $P(v \mid q, \{\boldsymbol{k_i}\})$ to the query q for the item v can be computed by replacing the weights of a softmax with item key embeddings as:

$$P(v \mid q, \{\boldsymbol{k_i}\}) = \frac{\exp(\boldsymbol{q}^T \boldsymbol{k_v})}{\sum_{v' \in \boldsymbol{V}} \exp(\boldsymbol{q}^T \boldsymbol{k_{v'}})},$$

where $\boldsymbol{q}$ is the query embedding; $\boldsymbol{k_v}$ is the item key embedding of the item v. Finally, the item key embeddings can be more informative if they are also capable of directly indicating the relevance to queries. Moreover, the auxiliary task can also be jointly addressed with the major retrieval task introduced in Section 2.4 so that the item key embeddings can be trained with both item attention and item key softmax.

2.6 Locality Preservation

The sequential user behaviors can also indicate the relationships between items and content providers. In other words, the contexts of items in user history can also be beneficial to learn their embeddings. In this work, we conduct locality preservation for local patterns of items in user history with a continuous bag-of-words (CBOW) model as an additional regularization task.

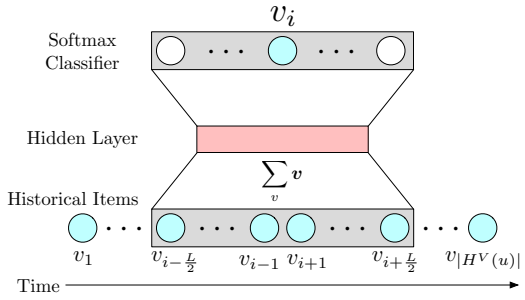

Figure 3: Schema of the continuous bag-of-words (CBOW) model for locality preservation with historical items.

Figure 3 shows an example of the CBOW model for items in user history. Given L context items of a certain item v_i, locality preservation assumes that the embeddings of context items are capable of inferring the item v_i. More formally, we aim to maximize the following objective computed by a softmax function as:

$$P\left(v_i \mid \boldsymbol{v}_{\boldsymbol{i-\frac{L}{2}}}, \boldsymbol{v}_{\boldsymbol{i-1}}, \boldsymbol{v}_{\boldsymbol{i+1}}, \boldsymbol{v}_{\boldsymbol{i+\frac{L}{2}}}\right) = \frac{\exp(\boldsymbol{s}_{\boldsymbol{v_i}}^T \boldsymbol{x_i})}{\sum_{v \in \boldsymbol{V}} \exp(\boldsymbol{s}_{\boldsymbol{v}}^T \boldsymbol{x_i})},$$

where $\boldsymbol{s_j}$ derives the logit for the item j; $\boldsymbol{x_i}$ is the summation of the embeddings of context items as $\boldsymbol{x_i} = \sum_{i-\frac{L}{2} \le j \le i+\frac{L}{2}, j \ne i} \boldsymbol{v_j}$. Similarly, the item key embeddings $\boldsymbol{k_i}$ can be also regularized by locality preservation based on the sequences of items in user history as shown in Figure 2.

2.7 Multi-task Learning and Optimization

Multi-task learning is applied to simultaneously optimize the objectives of different components in EDAM, including (1) classification as ranking, (2) the auxiliary ranker, and (3) locality preservation. Each component has a corresponding loss jointly optimized with the losses of other components.

For classification as ranking and the auxiliary ranker, the tasks solve extreme multi-class classification problems with shared training data. Hence, we utilize the cross-entropy [19] between the predicted distributions and the gold standard $\boldsymbol{y}$ as the loss functions. Formally, the losses of two tasks can be computed as:

$$L_{\text{rank}} = -\sum_{v \in \boldsymbol{V}} y_v \log P(v \mid q, u), L_{\text{aux}} = -\sum_{v \in \boldsymbol{V}} y_v \log P(v \mid q, \{\boldsymbol{k_i}\}),$$

where y_v indicates if v is the label item in the gold standard $\boldsymbol{y}$.

For locality preservation, it can be treated another extreme multi-class classification task for each item or content provider in user history. Hence, the locality preservation loss for different embeddings can be represented as:

$$L_{\text{locality\_item}} = \sum_{v_i \in H^V(u)} \sum_{v_j \in \boldsymbol{V}} \mathbb{1}[v_i = v_j] \log P\left(v_j \mid \boldsymbol{v}_{\boldsymbol{i-\frac{L}{2}}}, \boldsymbol{v}_{\boldsymbol{i-1}}, \boldsymbol{v}_{\boldsymbol{i+1}}, \boldsymbol{v}_{\boldsymbol{i+\frac{L}{2}}}\right),$$

$$L_{\text{locality\_item\_key}} = \sum_{v_i \in H^V(u)} \sum_{v_j \in \boldsymbol{V}} \mathbb{1}[v_i = v_j] \log P\left(v_j \mid \boldsymbol{k}_{\boldsymbol{i-\frac{L}{2}}}, \boldsymbol{k}_{\boldsymbol{i-1}}, \boldsymbol{k}_{\boldsymbol{i+1}}, \boldsymbol{k}_{\boldsymbol{i+\frac{L}{2}}}\right),$$

where the overall loss for locality preservation can be defined as:

$$L_{\text{locality}} = L_{\text{locality\_item}} + L_{\text{locality\_item\_key}}.$$

Finally, the objective of multi-task learning combines the loss functions of different components as $L = L_{\text{rank}} + L_{\text{auxiliary}} + L_{\text{locality}}$.

Efficient Optimization. To efficiently train the model with millions of items and content providers in corpora, we rely on sampling negative classes as candidates from the background distribution to avoid exhausting computations [22]. More specifically, for each training instance, the cross-entropy is minimized with the class of the true label and several negative sampled classes. Practically, sampling several thousands of negative classes can lead to more than 100 times speedup over the conventional optimization in the production systems as shown in previous studies [8].

For the task of locality preservation, it is also time-consuming to enumerate all individual items and content providers in user history. Hence, in each training epoch, we stochastically sample an item and a content provider for optimizing the objectives. In other words, we manually conduct stochastic gradient descent for the part of learning locality preservation, thereby reaching several hundred times speedup over enumerating all possible candidates.

3 EXPERIMENTS

In this section, we conduct extensive experiments and in-depth analysis to verify the performance of our proposed approach.

3.1 Experimental Settings

Experimental Dataset. The experiments of this paper are conducted based on user logs of a large video media platform at Google with videos as items and channels as content providers.The dataset consists of 400 most recent accessed items of 184M users, where some of the items are accessed after issuing queries. To alleviate the impact of rare items, items are replaced with an out-of-vocabulary (OOV) item if the items are not among the top 1M items. Similarly, content providers are taken over by an OOV provider if they are not in the list of the top 400K providers.

Label Items and History Selection. The items associated with queries are treated as the label items that are relevant to the corresponding queries. Note that OOV items will not be selected as labels to prevent both training and evaluation from the noises caused by the ambiguity. To avoid the temporal leakage in the logs, we follow previous work [8] to derive the contexts as personal item history. Figure 4 shows the illustration of label items and an example of selecting personal item history for a given query. For instance, v_i is not a label item without a corresponding query. In contrast, v_j is a label item because it is accessed after the query q_j. Given the label item v_j with a query q_j, the selected item history is considered as $\{v_1, v_2, \cdots, v_j\}$.

Training and Evaluation. For evaluation, we randomly sample 10% of users and their logs as testing data while the data of the remaining 90% of users are considered as the training dataset. To reduce the bias of diligent users with more label items, we only adopt the last label item of each user in the testing dataset for evaluation. On the contrary, in each epoch, we independently sample a label item for each training user so that popular users would not be over-trained with more label items. Moreover, different label items of a user can be examined over training epochs.

Competitive Baselines. Although none of the existing works focuses on personalized item retrieval for online content-sharing

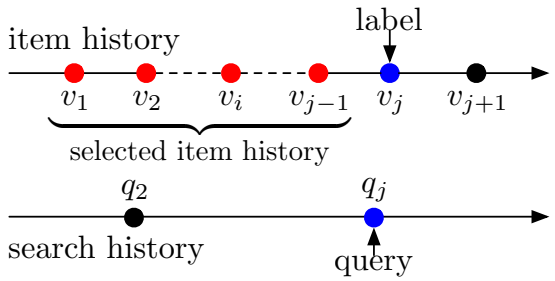

Figure 4: Illustration of label items and the selection of personal item history for a given query.

platforms without descriptive information, methods for other retrieval tasks can be modified by replacing the encoders of descriptive information with embeddings as a workaround. In our experiments, query embedding model (QEM) [3], hierarchical embedding model (HEM) [3], zero-attention model (ZAM) [2], attentive convolutional neural networks (ACNN) and recurrent neural networks (ARNN) [16] are considered as the comparative baseline methods.

Evaluation Metrics. For evaluation, we adopt success rate at top-k (SR@k) [28] to evaluate the performance of models. More precisely, SR@k denotes the percentage of the label items that can be found in the top-k ranked items.

Implementation Details. The model is implemented by Tensorflow [1] and optimized by Adam [24] with an initial learning rate of 1e-5. The embedding dimension d and the number of hidden units d_h are set as 128 and 256 after fine-tuning. For all of the baselines, we also fine-tune all hyper-parameters for fair comparisons and evaluation.

3.2 Experimental Results

Overall Evaluation. Figure 5 shows the performance of different methods. For the baselines, QEM performs the worst because it only considers query information while HEM achieves better performance by exploiting user history. AEM and ZAM are the best baselines with the attention that appropriately identifies important items and providers in history. ACNN and ARNN perform worse because they over-emphasize the sequential information, which is not as essential as the relations between history and the query for personalized search. Our proposed EDAM significantly outperforms all baselines. This is because the external item key embeddings can more appropriately model the query-history relations while locality preservation properly learns sequential knowledge.

Length of User History. We then analyze the performance with different lengths of user history. Figure 6 demonstrates the SR@1 scores of methods over different numbers of items in user history. For all methods using user history, the improvements against QEM are greater with more items in user history. When the number of historical items is limited, all baselines exploiting user history perform worse than QEM. In contrast, our proposed EDAM consistently outperforms all baselines over different history lengths. It shows that EDAM is capable of deriving essential information from user history across different situations of user history.

Ablation Study. Here we conduct an ablation study to demonstrate the effectiveness of different components in EDAM. Table 2

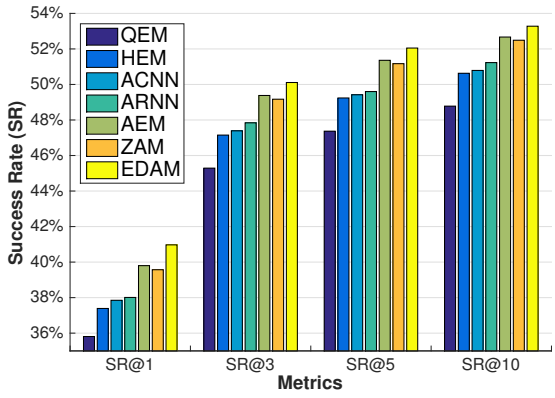

Figure 5: Success rates of methods at different positions.

Table 2: The SR@1 scores of EDAM with and without the auxiliary ranker (AR) and locality preservation (LP).

Method	Length of User History				
	Overall	[0, 50]	[51, 100]	[101, 200]	[201, 400]
EDAM	**0.4097**	**0.3297**	**0.3718**	**0.3822**	**0.4196**
-AR	0.3973	0.3031	0.3522	0.3696	0.4089
-LP	0.4039	0.3143	0.3591	0.3729	0.4155

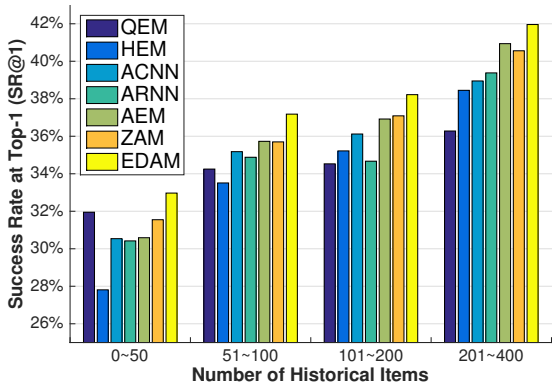

Figure 6: SR@1 scores of methods with different lengths of user history.

depicts the SR@1 scores of EDAM with and without the auxiliary ranker (AR) and locality preservation (LP). The results show that both AR and LP are consistently beneficial across different history lengths while AR plays a more important role for EDAM. Especially, AR leads to greater improvements for shorter user history. It further demonstrates that the ability of EDAM to model personal information with only limited data as shown in Figure 6.

Content Provider Key Embeddings. In addition to historical items, we also attempt to apply external key memory into modeling content providers for personalization. Table 3 shows the SR@1 scores of ZEM and EDAM using two different key embeddings. Although item key embeddings lead to significant improvements, external key memory does not work for modeling content providers. This can be because label items of a content provider can be relevant

Table 3: The SR@1 scores of ZEM and EDAM using item key embeddings or content provider key embeddings.

Method	Length of User History				
	Overall	[0, 50]	[51, 100]	[101, 200]	[201, 400]
ZEM	0.3957	0.3155	0.3570	0.3709	0.4056
EDAM (Item)	**0.4097**	**0.3297**	**0.3718**	**0.3822**	**0.4196**
EDAM (Provider)	0.3808	0.3106	0.3513	0.3608	0.3892

to different queries so that the learned embeddings are noisy. Hence, EDAM only adopts the item key embeddings. In order to model the relations between the query and user history, the query-aware attention mechanism with external key memory is proposed to derive the representations of historical and content providers. The sequential knowledge can also be learned by preserving the item locality in history.

4 CONCLUSIONS

In this paper, we propose EDAM to address the problems of personalized item retrieval for online content-sharing platforms without any descriptive information based on the query-aware attention mechanism with external key memory and locality preservation. Experimental results and analysis on the large-scale dataset from a real-world commercial online content-sharing platform also demonstrate the effectiveness and the robustness of EDAM. The insights can be concluded as follows: (1) user history is helpful for personalized item retrieval; (2) learning external key item embeddings for estimating attention weights is beneficial, especially for the users with shorter item history; (3) sequential information in user history is sensitive for item retrieval so that EDAM with locality preservation outperforms baselines of sequence models such as ARNN.

REFERENCES

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. 265–283.

[2] Qingyao Ai, Daniel N Hill, SVN Vishwanathan, and W Bruce Croft. 2019. A Zero Attention Model for Personalized Product Search. *arXiv preprint arXiv:1908.11322* (2019).

[3] Qingyao Ai, Yongfeng Zhang, Keping Bi, Xu Chen, and W Bruce Croft. 2017. Learning a hierarchical embedding model for personalized product search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 645–654.

[4] Kamelia Aryafar, Devin Guillory, and Liangjie Hong. 2017. An Ensemble-based Approach to Click-Through Rate Prediction for Promoted Listings at Etsy. In *Proceedings of the ADKDD'17*. ACM, 10.

[5] Paul N Bennett, Ryen W White, Wei Chu, Susan T Dumais, Peter Bailey, Fedor Borisyuk, and Xiaoyuan Cui. 2012. Modeling the impact of short-and long-term behavior on search personalization. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*. ACM, 185–194.

[6] Jean Burgess and Joshua Green. 2018. *YouTube: Online video and participatory culture*. John Wiley & Sons.

[7] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, and Tat-Seng Chua. 2017. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. ACM, 335–344.

[8] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*. ACM, 191–198.

[9] Mariam Daoud, Lynda Tamine-Lechani, and Mohand Boughanem. 2008. Learning user interests for a session-based personalized search. In *Proceedings of the second international symposium on Information interaction in context*. ACM, 57–64.

[10] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, et al. 2010. The YouTube video recommendation system. In *Proceedings of the fourth ACM conference on Recommender systems*. ACM, 293–296.

[11] Huizhong Duan and ChengXiang Zhai. 2015. Mining coordinated intent representation for entity search and recommendation. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, 333–342.

[12] Huizhong Duan, ChengXiang Zhai, Jinxing Cheng, and Abhishek Gattani. 2013. A probabilistic mixture model for mining and analyzing product search log. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. ACM, 2179–2188.

[13] Anjan Goswami, Naren Chittar, and Chung H Sung. 2011. A study on the impact of product images on user clicks for online shopping. In *Proceedings of the 20th international conference companion on World wide web*. ACM, 45–46.

[14] Mihajlo Grbovic and Haibin Cheng. 2018. Real-time personalization using embeddings for search ranking at airbnb. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 311–320.

[15] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. ACM, 55–64.

[16] Yangyang Guo, Zhiyong Cheng, Liqiang Nie, Yinglong Wang, Jun Ma, and Mohan Kankanhalli. 2019. Attentive long short-term preference modeling for personalized product search. *ACM Transactions on Information Systems (TOIS)* 37, 2 (2019), 19.

[17] Malay Haldar, Mustafa Abdool, Prashant Ramanathan, Tao Xu, Shulin Yang, Huizhong Duan, Qing Zhang, Nick Barrow-Williams, Bradley C Turnbull, Brendan M Collins, et al. 2019. Applying deep learning to Airbnb search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1927–1935.

[18] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).

[19] Geoffrey E Hinton and Ruslan R Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *science* 313, 5786 (2006), 504–507.

[20] Yujing Hu, Qing Da, Anxiang Zeng, Yang Yu, and Yinghui Xu. 2018. Reinforcement learning to rank in e-commerce search engine: Formalization, analysis, and application. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 368–377.

[21] 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*. ACM, 2333–2338.

[22] Sebastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On using very large target vocabulary for neural machine translation. In *53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL-IJCNLP 2015*. Association for Computational Linguistics (ACL), 1–10.

[23] Shubhra Kanti Karmaker Santu, Parikshit Sondhi, and ChengXiang Zhai. 2017. On application of learning to rank for e-commerce search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 475–484.

[24] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[25] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 1419–1428.

[26] Sheng Li, Jaya Kawale, and Yun Fu. 2015. Deep collaborative filtering via marginalized denoising auto-encoder. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, 811–820.

[27] Soon Chong Johnson Lim, Ying Liu, and Wing Bun Lee. 2010. Multi-facet product information search and retrieval using semantically annotated product family ontology. *Information Processing & Management* 46, 4 (2010), 479–493.

[28] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. 2010. Introduction to information retrieval. *Natural Language Engineering* 16, 1 (2010), 100–103.

[29] Nicolaas Matthijs and Filip Radlinski. 2011. Personalizing web search using long term browsing history. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 25–34.

[30] Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 1291–1299.

[31] Makbule Gulcin Ozsoy, Kezban Dilek Onal, and Ismail Sengor Altingovde. 2014. Result diversification for tweet search. In *International Conference on Web Information Systems Engineering*. Springer, 78–89.

[32] Nish Parikh and Neel Sundaresan. 2011. Beyond relevance in marketplace search. In *Proceedings of the 20th ACM international conference on Information and knowledge management*. ACM, 2109–2112.

[33] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*. AUAI Press, 452–461.

[34] Xuehua Shen, Bin Tan, and ChengXiang Zhai. 2005. Implicit user modeling for personalized search. In *Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM, 824–831.

[35] Börkur Sigurbjörnsson and Roelof Van Zwol. 2008. Flickr tag recommendation based on collective knowledge. In *Proceedings of the 17th international conference on World Wide Web*. ACM, 327–336.

[36] David Sontag, Kevyn Collins-Thompson, Paul N Bennett, Ryen W White, Susan Dumais, and Bodo Billerbeck. 2012. Probabilistic models for personalizing web search. In *Proceedings of the fifth ACM international conference on Web search and data mining*. ACM, 433–442.

[37] Bin Tan, Xuehua Shen, and ChengXiang Zhai. 2006. Mining long-term search history to improve search accuracy. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 718–723.

[38] Yury Ustinovskiy and Pavel Serdyukov. 2013. Personalization of web-search using short-term browsing context. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. ACM, 1979–1988.

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[40] Jan Vosecky, Kenneth Wai-Ting Leung, and Wilfred Ng. 2014. Collaborative personalized twitter search with topic-language models. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM, 53–62.

[41] Chen Wu, Ming Yan, and Luo Si. 2017. Ensemble methods for personalized e-commerce search challenge at CIKM Cup 2016. *arXiv preprint arXiv:1708.04479* (2017).

[42] Jun Yu, Sunil Mohan, Duangmanee Pew Putthividhya, and Weng-Keen Wong. 2014. Latent dirichlet allocation based diversified retrieval for e-commerce search. In *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, 463–472.

[43] Hamed Zamani, Michael Bendersky, Xuanhui Wang, and Mingyang Zhang. 2017. Situational context for ranking in personal search. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 1531–1540.