Improving Web Image Search with Contextual Information

Xiaohui Xie
BNRist, DCST, Tsinghua University
Beijing, China
xiexh_thu@163.com

Jiaxin Mao
BNRist, DCST, Tsinghua University
Beijing, China
maojiaxin@gmail.com

Yiqun Liu*
BNRist, DCST, Tsinghua University
Beijing, China
yiqunliu@tsinghua.edu.cn

Maarten de Rijke
University of Amsterdam
Amsterdam, The Netherlands
derijke@uva.nl

Qingyao Ai
School of Computing
University of Utah
Salt Lake City, UT
aiqy@cs.utah.edu

Yufei Huang
DCST, Tsinghua University
Beijing, China
huangyf16@mails.tsinghua.edu.cn

Min Zhang
BNRist, DCST, Tsinghua University
Beijing, China
z-m@tsinghua.edu.cn

Shaoping Ma
BNRist, DCST, Tsinghua University
Beijing, China
msp@tsinghua.edu.cn

ABSTRACT

In web image search, items users search for are images instead of Web pages or online services. Web image search constitutes a very important part of web search. *Re-ranking* is a trusted technique to improve retrieval effectiveness in web search. Previous work on re-ranking web image search results mainly focuses on intra-query information (e.g., human interactions with the initial list of the current query). Contextual information such as the query sequence and implicit user feedback provided during a search session prior to the current query is known to improve the performance of general web search but has so far not been used in web image search. The differences in result placement and interaction mechanisms of image search make the search process rather different from general Web search engines. Because of these differences, context-aware re-ranking models that have originally been developed for general web search cannot simply be applied to web image search.

We propose CARM, a *context-aware re-ranking model*, a neural network-based framework to re-rank web image search results for a query based on previous interaction behavior in the search session in which the query was submitted. Specifically, we explore a hybrid encoder with an attention mechanism to model intra-query and inter-query user preferences for image results in a two-stage structure. We train context-aware re-ranking model (CARM) to jointly learn query and image representations so as to be able to deal with the multimodal characteristics of web image search.

Extensive experiments are carried out on a commercial web image search dataset. The results show that CARM outperforms state-of-the-art baseline models in terms of personalized evaluation metrics. Also, CARM combines the original ranking can improve the original ranking on personalized ranking and relevance estimation. We make the implementation of CARM and relevant datasets publicly available to facilitate future studies.

CCS CONCEPTS

• **Information systems → Users and interactive retrieval**.

KEYWORDS

Web image search, Search result re-ranking, User session

ACM Reference Format:
Xiaohui Xie, Jiaxin Mao, Yiqun Liu, Maarten de Rijke, Qingyao Ai, Yufei Huang, Min Zhang, and Shaoping Ma. 2019. Improving Web Image Search with Contextual Information. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM '19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3357384.3358011

1 INTRODUCTION

Web image search is a vital part of web search. Textual queries with an image search intent are the most popular type of query on mobile phone devices and the second most popular on desktop and tablet devices [28]. While the performance of web image search engines has improved considerably in recent years [7, 36], there remains considerable room for improvement [22]. Existing work aimed at improving the performance of web image search engines, attempts to reorder visual documents based on the information manifested in the initial result list or other knowledge sources. In the contenxt of text-to-image search scenarios,[1] re-ranking methods can be grouped into three major categories [22]: (1) self-re-ranking methods that extract relevant visual patterns from the initial list and re-rank results based on image similarity graph or cluster [12]; (2) crowd-re-ranking methods that use results from multiple image resources (e.g., results from different search

*Corresponding author

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11…$15.00
https://doi.org/10.1145/3357384.3358011

[1] Content-based image search is not discussed in this paper.

engines or expanding results based on query suggestion) [18, 39]; and (3) interactive re-ranking methods that require human interaction during the re-ranking process in which users can provide complementary requirments or annotate results [33].

Although the methods mentioned above achieve promising performance improvements, they do not consider contextual information, i.e., query sequence and implicit user feedback before the current query in a search session, which has been shown to be beneficial for general web search re-ranking [15, 20, 41].[2] We hypothesize that incorporating contextual information can further improve the ranking performance of web image search. Figure 1 shows a real-world example of a web image search session from a commercial search engine. In this search session, the user first

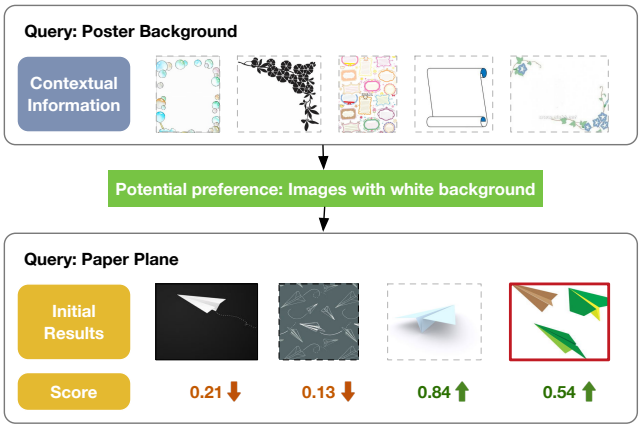

Figure 1: A real-world example of a web image search session. Based on the clicked/hovered images in the search engine result page (SERP) of the previous query "poster background" (contextual information), results for the current query "paper plane" can be re-ranked. The image highlighted with a red box is the image clicked by the user in the initial list for the second query. (The scores are explained in the text.)

submits a query "Poster Background" and then clicks or hovers their cursor on several image results. Then, this user issues another query "Paper Plane" and receives a result list from the search engine. After examining the returned results, the image highlighted with a red box is clicked by the user. In this case, the user wants to find images for making a poster and already has some requirements about these images; following the taxonomy due to [35] the user appears to have a "Locate/Acquire" intent. From the images clicked or hovered by the user in query "poster background," we can infer that the one aspect of user preference in this task may be white background. This preference may result in a click on the image with a white background in query "Paper Plane."

It is hard to apply standard context-aware models developed for general web search to web image search due to differences in interaction mechanisms and result placements between these two search scenarios. In web image search, items users search for are images instead of documents with text content. Hence, determining whether query terms appear in previous documents, which is required by most session search models in general web search [20, 41], is simply not applicable. Also, image results are self-contained, in the sense that users do not have to click the results to view the landing page as in general web search, which leads to click sparsity problem [36]. The sparsity of clicks in image search generates a challenge to training click-based models for general web search re-ranking [15, 20]. And, naturally, the multi-modal nature of web image search needs to be taken into account, with methods that are able to bridge textual queries and visual results. To date, there has been very little work on re-ranking image search results using contextual information.

We propose a novel neural network-based framework, *context-aware re-ranking model* (CARM) to incorporate contextual information so as to re-rank image results. Specifically, we consider "cursor hovering" as an additional feedback signal for user preferences and propose a two-stage structure to generate a preference representation based on previous clicked or hovered images and issued queries. In the first stage, image results are mapped to dense vectors through an embedding layer to represent user preferences for the current query (i.e., intra-query user preferences). In the second stage, we use recurrent neural networks (RNNs) and an attention mechanism to model the preference by considering sequential behavior in the same search session (i.e., inter-query user preferences). The query text is also mapped to dense vectors through a trainable embedding layer. Given an image in the result list that needs to be re-ranked, we obtain a context-aware score by measuring the similarity between the representation of this image and user preference representation. For instance, returning to the example in Figure 1, we show re-ranking scores provided by CARM in red (down) and green (up). CARM assigns higher scores to images with a white background, which aligns with the user preferences picked from the context.

Below, we report on extensive experiments on a commercial web image search dataset. Our results demonstrate that the proposed CARM model outperforms state-of-the-art baselines in terms of personalized evaluation metrics. We also show that incorporating contextual information can improve the original ranking on personalized ranking and relevance estimation.

The key technological contributions of this work are:

- We formally define the problem of context-aware image re-ranking within web image search scenarios.
- We propose a novel web image search re-ranking model, named CARM, that considers contextual information. CARM explore a hybrid encoder to better model user preference and jointly learns image and query representation to tackle the multimodal issue.
- We conduct extensive experiments to test the performance of CARM. Experimental results demonstrate that CARM performs effectively. We make the implementation of CARM and related datasets publicly available to facilitate future studies.

[2] A *search session* is defined as an uninterrupted sequence of activity in the system. The session ends when the user is inactive for more than the predefined number of minutes [9]. We set 30 minutes for this number, following [31].

2 RELATED WORK

2.1 Image search re-ranking

Image search re-ranking aims to reorder image results based on multimodal cues, which may be specific visual patterns from the initial search results or knowledge obtained from different sources [7, 18, 22, 27]. Most existing work on image search re-ranking focuses on the current query or the current result list of the query. *Self re-ranking* methods mine relevant visual patterns from the initial list. For instance, Jain and Varma [12] hypothesize that images clicked in response to a query are most relevant to the query and employ Gaussian Process regression to predict a re-ranking score for each image. *Crowd re-ranking* methods attempt to select and fuse candidate results from multiple resources to generate the final result list. For instance, Liu et al. [18] construct a set of visual words on the basis of local image patches collected from multiple image search engines and formalize re-ranking as an optimization problem based on mined patterns among visual words.

Interactive re-ranking methods involve user interactions (i.e., human labor and feedback) to refine search results. For example, Wang and Hua [33] propose an image search system, image search by color map, that enables users to specify color distributions in the desired images. This system provides a way to enable users to indicate their visual expectation. Although methods the reviewed above are promising, they do not consider user preferences encoded in the past history (i.e., contextual information), which have been shown to be valuable for search result re-ranking [20].

Besides the methods listed above, Sang et al. [27] capture user preferences on the basis of annotations and the participation of interest groups of image search users on Flickr. Cui et al. [7] build a user-image interest graph on the photo sharing platform and use the graph to re-rank search results. However, these models require a user profile that includes tag information or group information, which is not available in web image search.

2.2 Web search session search

In general web search, a lot of research has been devoted to session search. The Text REtrieval Conference (TREC) session track [5, 6] has built a standard protocol (datasets, experimental and evaluation settings) for multiple query-response interactions in web search. Models considering content information (query and document) and user interaction have been developed. Zhang et al. [41] utilize query change as a new form of relevance feedback for better session search. Similar to Zhang et al. [41], Guan et al. [10] propose a query change retrieval model (QCM) and model the entire session as a Markov Decision Process.

Besides considering relevance, user session level diversity has also been used to re-rank search results [25]. Prior work has also investigated session-level evaluation metrics to better reflect user satisfaction in a search session. For instance, Session-based DCG (sDCG) [14] assumes that the documents at a lower position and retrieved by a later query are less likely to be read by users, hence, these documents have less influence on session-level satisfaction. Luo et al. [19] propose cube test (CT) which takes the information nugget and importance into account; the gain of a result is discounted if the same nugget has been encountered in previous results.

Due to users' different and unique interactions with web image search engines when compared with general web search engines, it is hard to apply standard session models that have been shown to be useful for general web search to image search in a straightforward manner. There exists very little work on utilizing contextual information to improve web image search.

2.3 Session-based recommendation

Contextual information in the form (estimated) user intent and preference is also being used in session-based recommendations. Here, the recommender system recommends based on the behavior of users in the current browsing session. Recently, RNNs have been used to model variable-length session data. Hidasi et al. [11] are the first apply RNNs to session-based recommendation with remarkable results. Furthermore, Tan et al. [30] study an extension to this RNN framework and Li et al. [17] propose neural attentive recommendation machine (NARM) to both model the user's sequential behavior and capture the user's main purpose. Ren et al. [24] integrate a regular neural recommendation approach in an encoder-decoder structure with a repeat recommendation mechanism that can choose items from a user's history.

While models developed for session-based recommendation can provide insights into the design of contextual re-ranking models for web image search, most session-based recommendation models do not consider query information which has been shown to be valuable for understanding user preference [41]. In this paper, we propose CARM, which jointly models queries and image results to better capture the inter-query preferences of search users.

3 METHOD

We first introduce the context-aware image re-ranking task. Then we describe the proposed CARM in detail.

3.1 Problem definition

Context-aware image re-ranking is the task of re-ranking image results of the current query based on user preference. We hypothesize that query sequence and implicit user feedback during a search session prior to the current query to some extent encode user preference. As shown in [36], user clicks and cursor hovering can be useful signals for relevance in image search scenarios. In this paper, we also consider these two types of implicit user feedback as signals of user preference.

Given a search session with n consecutive queries $\langle Q_0, Q_1, \ldots, Q_{n-1}\rangle$, on the SERP of a query Q_i $(0 \leq i \leq n-1)$, m_i images are hovered or clicked (i.e., implicit user feedback is collected). Let $\langle I_0, I_1, \ldots, I_{m_i-1}\rangle$ denotes these m_i images. Assuming we want to reorder image results for the query Q_k, where $1 \leq k \leq n-1$, to ensure there exists a context, we build a model M so that for any given image I' on the SERP of the query Q_k, we can generate a context-aware score for this image, which can be formulated as:

$$S_c = M(I') = f(I'|U, Q_k), \tag{1}$$

where U is the representation of user preference implicit in contextual information. Besides testing the performance of uncovering user preference, we also want to evaluate how well the model M can

improve the original ranking. Hence, we can combine the context-aware score S_c and the original ranking score S_o together using a trade-off parameter λ to obtain a final re-ranking score S:

$$S = \lambda S_c + (1 - \lambda) S_o. \tag{2}$$

Both S and S_c can be used to reorder the original result list. We will further describe details of how to compute these scores and evaluate their re-ranking performance in the following sections.

3.2 Context-aware re-ranking model

We show the proposed CARM framework in Figure 2. Two trainable embedding layers map the query content and image ID into dense vectors. A two-stage encoding architecture captures both intra-query and inter-query user preferences. The context-aware re-ranking score S_c can then be computed according to the similarity of the generated user preference and target image.

3.2.1 Embedding layers. In this paper, we use two trainable embedding layers for query and image respectively. Since the number of possible queries in web image search is very large, we define a projection function δ to combine word-level embeddings to form a query-level embedding e_Q as in [1], which can be formulated as:

$$e_Q = \delta(w_Q | w_Q \in Q) = \tanh\left(W \cdot \frac{\sum_{w_Q \in Q} w_Q}{|Q|} + b\right), \tag{3}$$

where w_Q is the word-level embedding, $|Q|$ is the number of words in query Q, and $W \in R^{\alpha \times \alpha}$ and $b \in R^{\alpha}$ are learnable parameters. We aggregate and average word-level embeddings first and adopt a non-linear projection layer over the averaged word-level embeddings to obtain the query-level embedding. Ai et al. [1] demonstrate that considering non-linear relations between queries and words is beneficial. We leave investigating other sophisticated methods to combine word-level embeddings as future work. For the image embedding layer, we directly map the image ID to a dense vector. We use pre-trained weights to initiate embedding layers which are shown to be beneficial for the text-image task [34]. For the word embedding, we adopt a large-scale embedding corpus [29]. For the image embedding e_I, we use representations of images at the penultimate layer of the pre-trained ResNet-34 model. Both query and image embeddings are trainable during the training process. To bridge the gap between query and image representation, we use a non-linear projection layer to map 512-dimensional image vectors to 200-dimensional vectors that have the same length as query vectors. Through trainable embedding layers, we map text-based queries and visual-based image results into the same latent space.

3.2.2 Two-stage encoding architecture. We use a two-stage encoding architecture to extract user preferences from contextual information. The first stage focuses on intra-query information, that is, combining preferred images of users to represent intra-query preference. In the second stage, RNN and a query-based attention layer are designed to capture inter-query preference.

In the first stage, for a given query Q_i in the context, we have m_i vectors $\langle e_{I_0}, e_{I_1}, \ldots, e_{I_{m_i-1}} \rangle$ to represent features of preferred images among search results of this query (i.e., clicked/hovered (CH) images). We use P_{Q_i} to denote user preferences extracted from implicit feedback on a SERP produced for query Q_i. In this paper, we formulate P_{Q_i} as:

$$P_{Q_i} = \sum_{j=0}^{m_i-1} e_{I_j}. \tag{4}$$

Based on Eq. 4, we can obtain a set of intra-query preference representation $\langle P_{Q_1}, P_{Q_2}, \ldots, P_{Q_n} \rangle$.

In the second stage, we apply an RNN with Gated Recurrent Units (GRU) to model the sequential structure among consecutive queries. We use RNN with GRU rather than standard RNNs in this paper for two reasons: (1) Hidasi et al. [11] demonstrate that a GRU can outperform standard Long Short-Term Memory (LSTM) units. (2) A GRU can better deal with the vanishing gradient problem [17]. We define h_t as the hidden state that holds information for the current unit at time t. For RNNs with GRU, h_t can be updated as follows:

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h'_t, \tag{5}$$

where h_{t-1} is the hidden state at time $t-1$ and h'_t is the current hidden state before update; $\odot$ is element-wise multiplication. In Eq. 5, an update gate z_t is needed to combine these two types of content, which is given by:

$$z_t = \sigma(W^{(z)} P_{Q_t} + U^{(z)} h_{t-1}), \tag{6}$$

where σ is a sigmoid activation function; P_{Q_t} is query-level preference, which is plugged into the network unit at time t. The hidden state h'_t can be computed as:

$$h'_t = \tanh[W P_{Q_t} + r_t \odot U h_{t-1}], \tag{7}$$

where the reset gate r_t is calculated as:

$$r_t = \sigma(W^{(r)} P_{Q_t} + U^{(r)} h_{t-1}). \tag{8}$$

Through the GRU component, we obtain a set of hidden states $\langle h_1, h_2, \ldots, h_n \rangle$ which encode both query-level preference and sequential information between consecutive queries.

Session search is a complex search task that may involve multiple subtasks [20, 23]. A subtask is to seek content covering one of aspects (subtopics) on a shared theme. Given a target query Q_T for which search results are planned to be re-ranked, user preferences encoded in previous queries that belong to the same subtask as Q_T should be emphasized. Consider, for example, a search session that consists of three queries "Sports bag", "Basketball shoe" and "Backpack". The theme for this search session could be "Sports equipment". The first and third query belong to a same subtopic "bag". When we plan to re-rank results of the third query based on contextual information, it might be beneficial to put more weight on preference information encoded in results for the first query than in the second one. We apply a query-based attention layer to dynamically select and linearly combine hidden states generated by RNNs to form the representation of the overall user preference encoded in the context before the target query Q_T. We write U_T to denote the overall user preference modeled by our model which is used to re-rank results of the target query Q_T. Then, U_T is given by:

$$U_T = \sum_{i=1}^{n} \frac{e^{\eta_i}}{\sum_{j=1}^{n} e^{\eta_j}} \cdot h_i \tag{9}$$

$$\eta_i = f(Q_i, Q_T) = e_{Q_i} \odot e_{Q_T}, \tag{10}$$

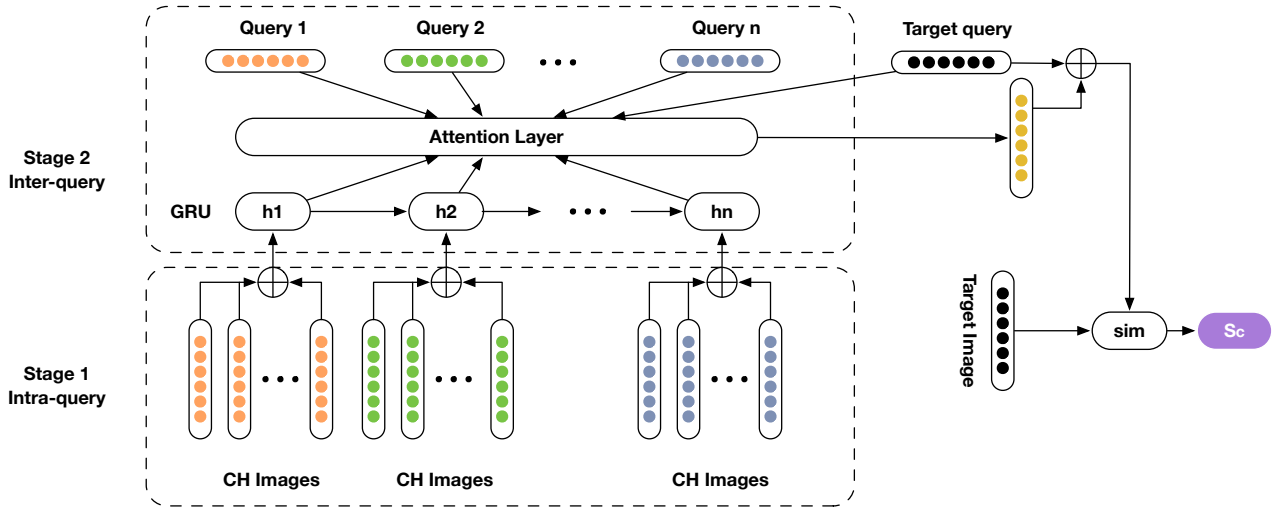

Figure 2: The proposed CARM framework. Through the trainable image embedding layer and the query embedding layer, image ID and query ID are mapped to dense vectors respectively. A two-stage encoding architecture extracts user preference based on the query and clicked/hovered (CH) image information. Context-aware score S_c is generated by considering user preference, target query and target image in a latent space.

where η is a query-based function which measures the alignment between the target query Q_t and the candidate query Q_i in the context. We apply element-wise multiplication to embedding vectors of Q_i and Q_T.

3.2.3 Context-aware score. Given a target query Q_T and an image result I' for Q_T, we calculate the context-aware score S_c of I' as follows:

$$S_c = \text{sim}(U_T + e_{Q_T}, I'), \tag{11}$$

where sim is a function that measures the similarity between two vectors in a latent space. Ai et al. [1], Van Gysel et al. [32] show that cosine similarity yields better performance on measuring similarity between latent representations. Hence, we also apply this measure of which results ranges from -1 (exactly opposite) to 1 (exactly the same). Besides user preference representation U_T, we also incorporate the embedding vector of the target query e_{Q_T} into the score computation as in [1]. Based on Eq. 11, we cannot only model the similarity between user preference and visual content of the given image, but also capture the query intent of the target query.

3.2.4 Training loss. We use pairwise loss [3] to learn parameters of the proposed CARM. We leave investigating other learning methods (e.g., list-wise loss) as future work. Let I_i and I_j be image results for the target query. Based on Eq. 11, we can obtain context-aware scores S_{ci} and S_{cj} for these two images, respectively. Let $I_i \triangleright I_j$ denote the event that I_i should be ranked higher than I_j (i.e., I_i is preferred by the search user than I_j in results of Q_T). The two outputs of CARM S_{ci} and S_{cj} are mapped to a learnable probability P_{ij} which can be computed as:

$$P_{ij} = P(I_i \triangleright I_j) = \frac{1}{1 + e^{-(S_{ci} - S_{cj})}}. \tag{12}$$

Let $\overline{P_{ij}}$ denote the actual probability that I_i is more preferred than I_j. We then apply the cross entropy function to form the pairwise loss function, which is given by:

$$L = -\overline{P_{ij}} \log(P_{ij}) - (1 - \overline{P_{ij}}) \log(1 - P_{ij}) + \beta \sum \xi^2, \tag{13}$$

where β is the strength of L2 regularization and ξ are parameters needed to be estimated during training process. To note here, we use both clicked and hovered images (CH images) when modeling user preference in the context while we only use clicked images to represent preferred images for the target query.

4 EXPERIMENTAL SETUP

In this section, we introduce our experimental settings for context-aware web image re-ranking. We describe the dataset and give details about our data partitions. We also introduce baseline models against which the proposed CARM is compared. Details of evaluation metrics and training settings are also provided.

4.1 Dataset

To the best of our knowledge, there is no publicly available dataset that is suitable for the context-aware image re-ranking task. We create a dataset by randomly sampling data from a commercial search log. To note here, a *query session* refers to search behavior of one query, while a *search session* consists of several query sessions. Using the user ID for identification, we group consecutive query sessions of the same user without interruption into a single search session. The search session ends when the user is inactive for more than 30 seconds. Also, since the target of our re-ranking task is to order results that are preferred by search users to a higher rank, we need to know which results are favored by the real-world users during the training and testing process. Hence, we keep search sessions of which the last query receives at least one click. Furthermore, we discard search sessions with only one query.

We show statistics of the dataset in Table 1. A total of 33,732 search sessions are used in our experiment. Besides the number of

Table 1: Statistics of the datasets used in our experiments ("#" refers to "number of"). A H (C) Qsession is a query session that has at least one hover (H) (click (C)) action. We do not consider the last query of a session when we count H (C) Qsessions.

#Sessions	#Distinct queries	#H Qsessions	#C Qsessions
33,732	18,311	36,340	17,214

search sessions and distinct queries, we also calculate the number of query sessions that have at least one hover (H Qsession) or at least one click (C Qsession). Since the last query of each search session contains at least one click due to the requirement of the task, we only consider queries before the last one of each search session when we calculate H Qsession and C Qsession. From Table 1, we see that the number of queries that receive at least one click is less than the number of queries with at least one hover. Xie et al. [36] also show that hovering data tends to be richer than click data and that cursor hovering can be an additional signal used for ranking in web image search. Thus, we consider clicked images and hovered images as users' preferred images in the context. We also show the distribution of search session length (i.e., the number of queries) of our dataset in Figure 3. Over 70% of the search sessions contain two queries, indicating that in a real-world web image search environment users tend to submit a single query reformulation.

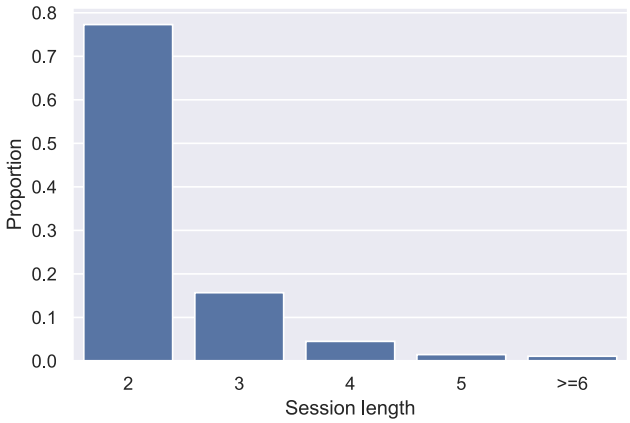

Figure 3: Distribution of search session length (The number of queries in a search session) of our dataset.

Furthermore, since our task is to re-order results for a given query on the basis of contextual information, we also use queries with at least one click before the last query and after the first query of a search session as the target query during the training and testing process.

We split all search sessions into training, validation and test sets at a ratio of 7:1:2.

4.2 Baseline models

To evaluate the performance of modeling contextual information, we compare the proposed CARM against existing context-aware models developed for general web search, web image search and session-based recommendation, respectively. We also compare the re-ranked result list obtained by using the combination score S ($S = \lambda S_c + (1-\lambda)S_o$) to the original ranked list to see whether contextual information can improve the original ranking.

4.2.1 Social-sensed image re-ranking model (SIRM). Cui et al. [7] propose an image re-ranking model that considers social relevance to measure the relevance of an image for a user's interest. Given a user U_i, the user interest is represented by a representative image set O_i; *social relevance* of an image I_j is then calculated as:

$$\chi(U_i, I_j) = \sum_{(I_k, P_{ik}) \in O_i} P_{ik}\phi(I_k, I_j)\delta_\rho(\phi(I_k, I_j)) \tag{14}$$

$$\phi(I_k, I_j) = \frac{|C(I_k) \cap C(I_j)|}{|C(I_k) \cup C(I_j)|}, \tag{15}$$

where P_{ik} is the transition probability of a user-image pair, which Cui et al. [7] calculate based on user profiles in Flickr. However, in web image search, user profiles are not available, which is one of the challenges of our paper. Thus, we set $P_{ik} = 1$ if I_k is clicked/hovered in the context, and $P_{ik} = 0$ otherwise. Following [7], we regard an image as a document and represent it as a bag-of-visual-words; $C(I_i)$ is the visual word set of image I_i. The function ϕ counts the co-occurrence of visual words in two images; ρ is a threshold to determine the value of δ, that is, $\delta_\rho = 1$ if $\phi(I_k, I_j) \geq \rho$ and $\delta_\rho = 0$, otherwise. Since no difference between different threshold settings is demonstrated in [7], we set ρ to be zero in this paper.

4.2.2 Rocchio model. The Rocchio model has been used for text categorization [15] and wide range of other information retrieval tasks, including web session search [20] and semantic image retrieval [21]. Basically, the Rocchio model is used to measure how close a document vector is to a keyword vector. In our task, a "document" refers to a given image I_j and the "keyword" refers to the user preference U_i. According to [15], U_i can be computed as the difference of the averages of the vectors w.r.t. positive example images (clicked or hovered images), D_{pi}, and negative example images (images not clicked or hovered), D_{ni}:

$$U_i = \frac{1}{|D_{pi}|} \sum_{I_k \in D_{pi}} \frac{\overline{I_k}}{\left\|\overline{I_k}\right\|} - \frac{1}{|D_{ni}|} \sum_{I_k \in D_{ni}} \frac{\overline{I_k}}{\left\|\overline{I_k}\right\|}, \tag{16}$$

where $\overline{I_k}$ is the bag-of-visual-words vector of image I_k and $\left\|\overline{I_k}\right\|$ is the Euclidean norm of the vector $\overline{I_k}$. After obtaining the representation of the user preference U_i, the Rocchio model computes the similarity score between U_i and the given image I_j as:

$$\text{sim}(\overline{I_j}, U_i) = \frac{\overline{I_j} \odot U_i}{\left\|\overline{I_j}\right\| \cdot \|U_i\|}. \tag{17}$$

Then, the similarity score can be used to re-rank the result list of the target query.

4.2.3 Neural Attentive Recommendation Machine (NARM). Li et al. [17] propose NARM with an encoder-decoder architecture to address the session-based recommendation problem. NARM uses a GRU to form a global encoder and a local encoder. The global encoder is used to model users' sequential features while the local encoder is used to capture the users' main purpose. Instead of using

query-level attention as we do in this paper, NARM uses an item-level attention mechanism that allows the decoder to determine a weighted combination of different parts of the input sequence.

We re-implement NARM according to the published paper [17] although some changes have been made for the training process. Specifically, we obtain a context-aware score from the similarity layer of NARM and train the model using the same pair-wise loss described in Section 3 instead of the original list-wise loss.

4.2.4 Original ranked list. This baseline simply returns the original search result ranked list produced after issuing a query by the commercial search engine from which our data was collected. The original ranking score S_{oi} of a given image I_i is defined as:

$$S_{oi} = 1 - \frac{rank(I_i)}{N}, \tag{18}$$

where $rank(I_i)$ is the rank of I_i in the original result list. We define the *rank* in a grid-based result panel by mapping tuple positions (row, column) to a numerical value as in [36], that is, following left-to-right and top-to-bottom order. Finally, N is the number of images on the SERP being considered.

4.3 Evaluation metrics

To assess the performance of our context-aware re-ranking model, we use several evaluation metrics, which can be divided into two groups. Evaluation metrics in the first group are mainly click-based. We want to test whether CARM can better model user preferences based on contextual information. The second group focuses on result relevance. Since we incorporate query intent into the calculation of the re-ranking score (see Eq. 11), we want to test the ability of CARM to estimate relevance.

The first group consists of Average Rank, Rank Scoring, Recall@k and MRR.

4.3.1 Average Rank. The Average rank metric has been used to measure the quality of personalized search [8] and context-aware image re-ranking [7]. The *average rank* of a query q is defined as:

$$AvgRank_q = \frac{1}{|O_q|} \sum_{I \in O_q} rank(I), \tag{19}$$

where O_q denotes the set of clicked images on query q. The final average rank on test set of target queries S is computed as:

$$AvgRank = \frac{1}{|S|} \sum_{q \in S} AvgRank_q. \tag{20}$$

A smaller average rank value indicates better re-ranking performance in terms of user preference.

4.3.2 Rank Scoring. Breese et al. [2] propose a rank scoring metric to evaluate the effectiveness of collaborative filtering systems. Dou et al. [8] use it to measure the performance of personalized web search. Given a query q, the *rank scoring* metric is computed as:

$$RS_q = \sum_j \frac{\delta(q, j)}{2^{(j-1)/(\alpha-1)}}. \tag{21}$$

Here, j is the rank of an image in the result list of query q; $\delta(q, j) = 1$ if image j is clicked and $\delta(q, j) = 0$ otherwise; α is the viewing half life. We use a half life of 10 images in this paper. The final score for an experiment over a test set S is:

$$RS = 100 \cdot \frac{\sum_q RS_q}{\sum_q RS_q^{max}}. \tag{22}$$

Here, RS_q^{max} is the maximum possible utility obtained when all images that have been clicked appear at the top of the ranked list. A larger rank scoring value indicates better performance of context-aware re-ranking.

4.3.3 Recall@K. Recall is the fraction of the results that are relevant to the query that are successfully retrieved. It has been used to measure the performance of session-based recommendation [17]. In this paper, we regard clicked images as relevant results and examine two different cut-offs: $K = 5$ and $K = 10$. A larger Recall value indicates better performance.

4.3.4 MRR. Mean Reciprocal Rank (MRR) is the average of reciprocal ranks of the desire items. MRR takes the rank of the item into consideration. Let $rank_i$ refer to the position of the first desired image (i.e., clicked image) for the query q_i. Then, MRR can be computed as:

$$MRR = \frac{1}{|S|} \sum_{q \in S} \frac{1}{rank_i}, \tag{23}$$

where S is the set of all target queries.

The second group of metrics that we consider includes Normalized Discounted Cumulative Gain (NDCG) and variant versions of NDCG considering grid-based behavior assumptions introduced in [37].

4.3.5 NDCG@K. We apply NDCG [13] to measure the performance of relevance estimation. For a ranked list of images, the DCG score is defined as:

$$DCG@K = \sum_{i=1}^{k} \frac{r_i}{\log_2(i+1)}, \tag{24}$$

where r_i is the relevance score at position i and K is the depth of the ranked list of images. Then, the NDCG@K score can be obtained by normalizing DCG@d using ideal DCG@d, which measures the perfect ranking. We show results of NDCG with two different cut-offs: $K = 5$, $K = 10$ and $K = 15$ as in [36].

4.3.6 NDCG-MB/SD/RS. Xie et al. [37] propose three grid-based assumptions (i.e., Middle bias (MB), Slower decay (SD) and Row skipping (RS)) to derive new grid-based evaluation metrics. Specifically, they revise the representations of continuation and stopping probability of search users by considering grid-based information. Grid-based behavior assumptions are closer to real user behavior and grid-based metrics can better reflect user satisfaction. We use grid-based NDCG scores to test the ability of CARM on estimating relevance of query-image pairs. We use $K = 10$ as the default cut-off setting for these metrics. The parameters (e.g., row skipping probability) of grid-based assumptions are set to the value which performs best performance described in [37].

4.4 Training settings

CARM uses 200-dimensional embeddings for queries and 512-dimensional embeddings for images. We use Adadelta [38] as the optimization algorithm, with the initial learning rate set to 0.1. The

mini-batch size is fixed at 128. For NARM and CARM, we apply the same epoch number and report the best results in the following sections. For CARM and all baseline models, we use both clicked and hovered images as the input and re-rank results in top 5 rows on SERP. We use one GRU layer in CARM, where the hidden size of GRU is fixed at 200. A Nvidia Titan X GPU is used to train all deep models. We share the source code of CARM and all baselines.[3]

5 RESULTS AND ANALYSIS

We can use different scores to reorder the result list (i.e., the context-aware score S_c or the combination score $S = \lambda S_c + (1 - \lambda)S_o$). Since the output of ***CARM*** is the context-aware score S_c, we write ***CARM+OR*** for using the combination score when reporting experimental results where ***OR*** denotes the original ranking.

We aim to answer the following three research questions:

(RQ1) Is CARM able to model contextual information better than other context-aware baseline models?
(RQ2) Is CARM able to improve original ranking in terms of personalized metrics?
(RQ3) How do different settings (e.g., trade-off-parameter and session length) affect the performance of CARM?
(RQ4) Is CARM also able to improve the performance of original ranking in terms of relevance estimation?

5.1 Comparison against baselines

To answer RQ1, we first compare the CARM model against other context-aware methods, i.e., the Rocchio model, SIRM, and NARM. The results of all methods in terms of five click-based evaluation metrics (i.e., Average ranking, Ranking score, Recall@5, Recall@10 and MRR) are shown in Table 2.

From Table 2, we have the following observations:

(1) In terms of all evaluation metrics, the proposed CARM significantly outperforms the baselines, which demonstrates that CARM can better model user preference on the basis of contextual information in web image search scenarios.
(2) Both NARM and CARM are neural network-based frameworks. Compared to bag-of-visual-words models (i.e., the Rocchio model and SIRM), NARM and CARM can better extract representative visual features from images since they enable embedding layers to be trained.
(3) NARM considers item-level attention to model users' main purpose while CARM forms the overall user preference by considering query-level attention. Our results demonstrate that query-level attention is more expressive than item-level attention and results in better performance of CARM.

5.2 Comparisons among different parameter settings

By using the combination score S to reorder image results, we want to test whether contextual information can further improve the original ranking in terms of personalized metrics to answer RQ2. We also investigate how different settings affect the performance of CARM to answer RQ3. We show results using Average Rank as the evaluation metric in these comparisons.

[3] https://github.com/THUxiexiaohui/Context-aware-Re-ranking-Model

Table 2: Context-aware re-ranking performance in terms of click-based evaluation metrics. ** (*): The difference between the baseline model and CARM is significant with p-value < 0.01 (0.05).

Model	Rocchio model	SIRM	NARM	CARM
AvgRank	14.50**	12.75**	11.26**	**10.70**
RankScore	42.30**	47.88**	52.59**	**54.87**
Recall@5	0.174**	0.241**	0.322**	**0.357**
Recall@10	0.350**	0.427**	0.502**	**0.543**
MRR	0.154**	0.216**	0.259**	**0.280**

In Figure 4 we show the performance of CARM+OR with different settings for the trade-off parameter λ. Recall that if $\lambda = 0$, then CARM+OR degrades to the original ranking model and CARM+OR coincides with CARM when $\lambda = 1$. From Figure 4, we have the following observations:

(1) Compared to the original ranking ($\lambda = 0$), all settings of CARM+OR ($0.1 \leq \lambda \leq 0.9$) achieve better performance in terms of Average Ranking, which demonstrates that incorporating contextual information can improve the original ranking.
(2) CARM+OR achieves the best performance when $\lambda = 0.3$. However, the differences between different settings of the trade-off parameter, especially from 0.3 to 0.9 are not significant, which means that CARM+OR is not sensitive to this parameter. The reason might be that different search sessions receive different benefits from contextual information. For some search sessions, contextual information plays a more important role, in which case a larger value of λ can result in larger improvements for these search sessions while less so for others. Thus, setting different values of λ for different search sessions may be helpful, as has already been shown in session search for general web search [4]. We leave further investigation of the trade-off parameter as our future work.
(3) By only considering contextual information, CARM is not able to outperform the original ranking, which demonstrates that biases (e.g., position bias and appearance bias) in the original ranking might affect the search behavior of users, confirming [36, 37]. Also, CARM only considers query and image features while the original ranking takes more sophisticated features such as the surrounding text of images into consideration.[4]

Next, we fix λ to 0.3 and compare CARM+OR against the original ranking with different search session lengths in Figure 5. The results show that CARM+OR outperforms the original ranking over all search session lengths. The largest improvement ($(9.85 - 8.70)/9.85 = 11.6\%$) is observed when the search session length equals 4. We also observe that search sessions with a session length larger than 5 receive smaller improvements than other search sessions. The reason can be that the number of search sessions with long session length is small.

[4] This observation is similar to observation made in many other re-ranking settings, especially in a production setting: the original ranker is highly optimized and possibly exploits a broad range of ranking features already; the re-ranker generates a new signal that may not be available to the original ranker (yet) and, hence, adds to it but cannot outperform it as a stand-alone ranker. An "old" example of this phenomenon is relevance feedback [see, e.g., 16]; a more recent example concerns neural re-ranking of results retrieved by lexical methods [see, e.g., 32].

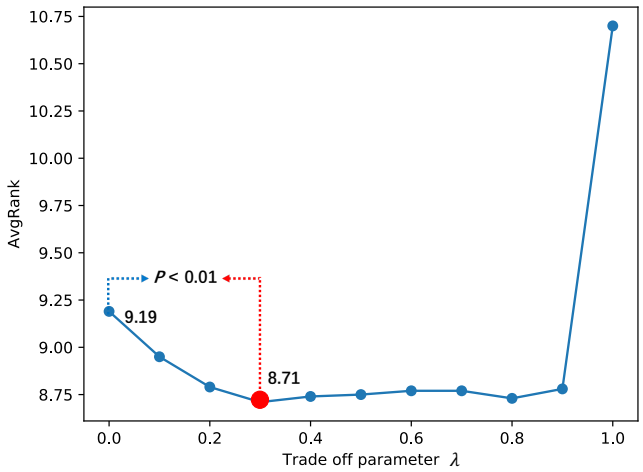

Figure 4: The performance of CARM+OR with different settings of the trade-off parameter λ. When $\lambda = 0$, the output of CARM+OR coincides with the original ranking score S_o. When $\lambda = 1$, CARM+OR coincides with CARM.

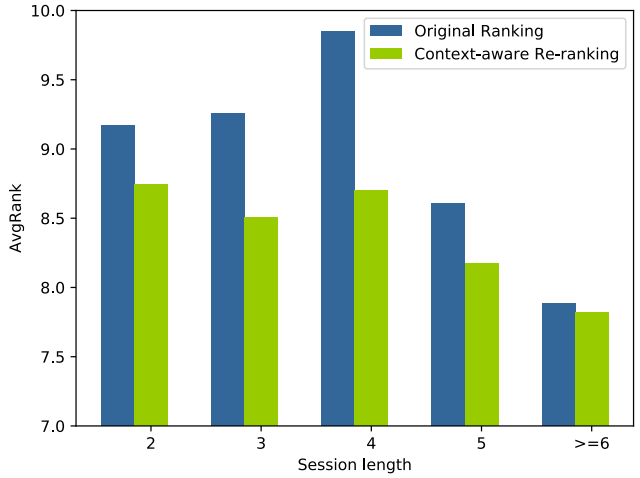

Figure 5: The performance of CARM+OR and original ranking with different search session lengths (The number of queries in a search session).

5.3 Evaluation of result relevance

The training target of CARM is to make sure that images clicked (i.e., preferred) by search users have a higher re-ranking score than images without click while the topical relevance of query-image pairs is not explicitly modeled. For a search service, personalization and result relevance are both important: personalization should not be achieved at a cost of relevance. We further conduct experiments to test whether CARM can preserve or even improve the overall result relevance of the original ranking in order to answer RQ4.

We randomly sample 300 search sessions from the dataset; after filtering out pornographic searches, 281 distinct queries and around 8,000 images are annotated. For each query-image pair topical relevance judgement, at least three editors are recruited to provide annotations based on the instructions illustrated in [40]. The Fleiss Kappa scores among annotators are higher than 0.5, which leads to substantial agreement. A 4-point scale judgement for each query-image pair is gathered: *Irrelevant* (0), *Somewhat relevant* (1), *Fairly relevant* (2), and *Highly relevant* (3).

Table 3: Relevance estimation performance in terms of NDCG@5, @10, @15 and NDCG with grid-based assumptions (i.e., MB, SD and RS). The cut-off K of grid-based NDCG is set to 10.

NDCG	@5	@10	@15	MB	SD	RS
Original rank	0.928	0.931	0.931	0.928	0.930	0.929
CARM+OR	**0.937**	**0.932**	**0.934**	**0.935**	**0.935**	**0.935**

On the basis of the annotation data, we compare CARM+OR against the original ranking in terms of NDCG with different cut-offs (@5, @10 and @15). Xie et al. [37] show that grid-based metrics can better reflect user satisfaction, hence, we also report results on grid-based NDCGs (MB, SD, and RS). The cut-off K of grid-based NDCGs is set to 10. The results are shown in Table 3.

We see that CARM+OR achieves slight improvements over the original ranking on all metrics, which demonstrates that contextual information can benefit relevance estimation to some extent. Also, the difference between the original rank and CARM+OR is larger in terms of grid-based NDCGs@10 compared to list-based NDCG@10. We also calculate the proportion of ties with the threshold parameter 0.01 [26] of NDCG@10 and NDCG-MB, respectively. The proportion of ties for NDCG@10 is 0.572 and for NDCG@MB it is 0.534, which means that NDCG with grid-based assumptions has a stronger discriminative power than NDCG with list-based assumptions in image search scenarios. The reason for the limited improvements in Table 3 can be that there is a gap between the user preference that is the optimization target of CARM and the topical relevance that is annotated by external editors [40].

6 CONCLUSION & FUTURE WORK

In this paper, we have formally defined the problem of image re-ranking using contextual information in web image search. We have proposed a novel web image search re-ranking model, named *context-aware re-ranking model* (CARM). Specifically, we map text-based queries and visual-based images to dense vectors in a latent space using trainable embedding layers. We explore a two-stage structure to better model user preference. In the first stage, we combine features of preferred images for a particular query to obtain intra-query preferences. In the second stage, we use a hybrid encoder with a query-based attention mechanism to capture inter-query sequential behavior of search users. CARM not only models context-aware user preferences but also captures query intent of the target query when it calculates the context-aware score. We train CARM to jointly learn query and image representations so as to deal with the multimodal nature of web image search.

Extensive experiments are conducted on a commercial web image search dataset. We find that (1) compared to state-of-the-art baseline models, CARM can better model user preference on the basis of contextual information in web image search scenarios; (2) CARM can further improve the original ranking in personalized ranking, that is, ordering preferred images of search users at higher

ranks; (3) CARM is not sensitive to the combination parameter that combines the context-aware score and the original ranking score, which demonstrates that contextual information may have a different impact on different search sessions; and (4) incorporating contextual information can improve the original ranking in terms of relevance estimation, which results in better relevance ranking.

CARM can be used to improve the performance of web image search. It can easily be transferred to other search environments that have contextual information (i.e., user interaction and query sequence) such as web search, product search or video search.

Through our experiments, we have obtained a better understandings about the advantages and the limitations of CARM. The limitations guide interesting directions for future work: (1) We use a fix parameter to combine the context-aware score and the original ranking score. Since different search sessions might benefit differently from contextual information, a more sophisticated way to determine the trade-off parameter for a given search session needs further investigation. (2) Parts of the design of CARM can be improved. For example, when modeling intra-query user preference, it might be beneficial to consider grid-based information and differences between click and hovering signals.

ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Program of China (2018YFC0831700), Natural Science Foundation of China (Grant No. 61622208, 61732008, 61532011), Ahold Delhaize, the Association of Universities in the Netherlands (VSNU), and the Innovation Center for Artificial Intelligence (ICAI). All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

REFERENCES

[1] Qingyao Ai, Yongfeng Zhang, Keping Bi, Xu Chen, and W Bruce Croft. 2017. Learning a hierarchical embedding model for personalized product search. In *SIGIR*. ACM, 645–654.

[2] John S Breese, David Heckerman, and Carl Kadie. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *UAI*. Morgan Kaufmann Publishers Inc., 43–52.

[3] Christopher Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N Hullender. 2005. Learning to rank using gradient descent. In *ICML*. 89–96.

[4] Fei Cai and Maarten de Rijke. 2016. Selectively personalizing query autocompletion. In *SIGIR*. ACM, 993–996.

[5] Ben Carterette, Paul Clough, Mark Hall, Evangelos Kanoulas, and Mark Sanderson. 2016. Evaluating retrieval over sessions: The TREC session track 2011-2014. In *SIGIR*. ACM, 685–688.

[6] Ben Carterette, Evangelos Kanoulas, Mark Hall, and Paul Clough. 2014. *Overview of the TREC 2014 session track*. Technical Report. Dept. Computer and Information Sciences, University of Delaware, Newark.

[7] Peng Cui, Shao-Wei Liu, Wen-Wu Zhu, Huan-Bo Luan, Tat-Seng Chua, and Shi-Qiang Yang. 2014. Social-sensed image search. *ACM Transactions on Information Systems (TOIS)* 32, 2 (2014), 8.

[8] Zhicheng Dou, Ruihua Song, and Ji-Rong Wen. 2007. A large-scale evaluation and analysis of personalized search strategies. In *WWW*. ACM, 581–590.

[9] Daniel Gayo-Avello. 2009. A survey on session detection methods in query logs and a proposal for future evaluation. *Information Sciences* 179, 12 (2009), 1822–1843.

[10] Dongyi Guan, Sicong Zhang, and Hui Yang. 2013. Utilizing query change for session search. In *SIGIR*. ACM, 453–462.

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

[12] Vidit Jain and Manik Varma. 2011. Learning to re-rank: query-dependent image re-ranking using click data. In *WWW*. ACM, 277–286.

[13] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 422–446.

[14] Kalervo Järvelin, Susan L Price, Lois ML Delcambre, and Marianne Lykke Nielsen. 2008. Discounted cumulated gain based evaluation of multiple-query IR sessions. In *ECIR*. Springer, 4–15.

[15] Thorsten Joachims. 1996. *A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization*. Technical Report. Department of Computer Science, Carnegie-Mellon University.

[16] Victor Lavrenko and W. Bruce Croft. 2001. Relevance Based Language Models. In *SIGIR*. ACM, 120–127.

[17] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural attentive session-based recommendation. In *CIKM*. ACM, 1419–1428.

[18] Yuan Liu, Tao Mei, and Xian-Sheng Hua. 2009. CrowdReranking: Exploring multiple search engines for visual search reranking. In *SIGIR*. ACM, 500–507.

[19] Jiyun Luo, Christopher Wing, Hui Yang, and Marti Hearst. 2013. The water filling model and the cube test: multi-dimensional evaluation for professional search. In *CIKM*. ACM, 709–714.

[20] Jiyun Luo, Sicong Zhang, and Hui Yang. 2014. Win-win search: Dual-agent stochastic game in session search. In *SIGIR*. ACM, 587–596.

[21] Joao Magalhaes and Stefan Rueger. 2007. High-dimensional visual vocabularies for image retrieval. In *SIGIR*. ACM, 815–816.

[22] Tao Mei, Yong Rui, Shipeng Li, and Qi Tian. 2014. Multimedia search reranking: A literature survey. *ACM Computing Surveys (CSUR)* 46, 3 (2014), 38.

[23] Karthik Raman, Paul N Bennett, and Kevyn Collins-Thompson. 2013. Toward whole-session relevance: exploring intrinsic diversity in web search. In *SIGIR*. ACM, 463–472.

[24] Pengjie Ren, Zhumin Chen, Jing Li, Zhaochun Ren, Jun Ma, and Maarten de Rijke. 2019. RepeatNet: A Repeat Aware Neural Recommendation Machine for Session-based Recommendation. In *AAAI*. AAAI, 4806–4813.

[25] Pengjie Ren, Zhumin Chen, Jun Ma, Shuaiqiang Wang, Zhiwei Zhang, Zhaochun Ren, and Tinghuai Ma. 2018. User session level diverse reranking of search results. *Neurocomputing* 274 (2018), 66–79.

[26] Tetsuya Sakai. 2006. Evaluating evaluation metrics based on the bootstrap. In *SIGIR*. ACM, 525–532.

[27] Jitao Sang, Changsheng Xu, and Dongyuan Lu. 2012. Learn to personalized image search from the photo sharing websites. *IEEE Transactions on Multimedia* 14, 4 (2012), 963–974.

[28] Yang Song, Hao Ma, Hongning Wang, and Kuansan Wang. 2013. Exploring and exploiting user search behavior on mobile and tablet devices to improve search relevance. In *WWW*. ACM, 1201–1212.

[29] Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In *NAACL*. 175–180.

[30] Yong Kiam Tan, Xinxing Xu, and Yong Liu. 2016. Improved recurrent neural networks for session-based recommendations. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. ACM, 17–22.

[31] Bartłomiej Twardowski. 2016. Modelling contextual information in session-aware recommender systems with neural networks. In *RecSys*. ACM, 273–276.

[32] Christophe Van Gysel, Maarten de Rijke, and Evangelos Kanoulas. 2016. Learning latent vector spaces for product search. In *CIKM*. ACM, 165–174.

[33] Jingdong Wang and Xian-Sheng Hua. 2011. Interactive image search by color map. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3, 1 (2011), 12.

[34] Liwei Wang, Yin Li, and Svetlana Lazebnik. 2016. Learning deep structure-preserving image-text embeddings. In *CVPR*. 5005–5013.

[35] Xiaohui Xie, Yiqun Liu, Maarten de Rijke, Jiyin He, Min Zhang, and Shaoping Ma. 2018. Why people search for images using web search engines. In *WSDM*. ACM, 655–663.

[36] Xiaohui Xie, Jiaxin Mao, Maarten de Rijke, Ruizhe Zhang, Min Zhang, and Shaoping Ma. 2018. Constructing an interaction behavior model for web image search. In *SIGIR*. ACM, 425–434.

[37] Xiaohui Xie, Jiaxin Mao, Yiqun Liu, Maarten de Rijke, Yunqiu Shao, Zixin Ye, Min Zhang, and Shaoping Ma. 2019. Grid-based evaluation metrics for web image search. (2019).

[38] Matthew D Zeiler. 2012. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701* (2012).

[39] Zheng-Jun Zha, Linjun Yang, Tao Mei, Meng Wang, Zengfu Wang, Tat-Seng Chua, and Xian-Sheng Hua. 2010. Visual query suggestion: Towards capturing user intent in internet image search. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 6, 3 (2010), 13.

[40] Fan Zhang, Ke Zhou, Yunqiu Shao, Cheng Luo, Min Zhang, and Shaoping Ma. 2018. How well do offline and online evaluation metrics measure user satisfaction in web image search?. In *SIGIR*. ACM, 615–624.

[41] Sicong Zhang, Dongyi Guan, and Hui Yang. 2013. Query change as relevance feedback in session search. In *SIGIR*. ACM, 821–824.