Denoising Attention for Query-aware User Modeling

Elias Bassani[1], Pranav Kasela[1,2], and Gabriella Pasi[1]

[1]University of Milano-Bicocca, Milan, Italy
e.bassani3@campus.unimib.it, gabriella.pasi@unimib.it
[2]ISTI-CNR, Pisa, Italy
pranav.kasela@unimib.it

Abstract

Personalization of search results has gained increasing attention in the past few years, also thanks to the development of Neural Networks-based approaches for Information Retrieval. Recent works have proposed to build user models at query time by leveraging the Attention mechanism, which allows weighing the contribution of the user-related information w.r.t. the current query. This approach allows giving more importance to the user's interests related to the current search performed by the user.
In this paper, we discuss some shortcomings of the Attention mechanism when employed for personalization and introduce a novel Attention variant, the Denoising Attention, to solve them. Denoising Attention adopts a robust normalization scheme and introduces a filtering mechanism to better discern among the user-related data those helpful for personalization. Experimental evaluation shows improvements in MAP, MRR, and NDCG above 15% w.r.t. other Attention variants at the state-of-the-art.

1 Introduction

The past few years have witnessed an increasing interest in Neural models for tackling various tasks of Information Retrieval (Guo et al., 2020; Kasela et al., 2024), among which Personalized Search. Two of the main challenges of Personalized Search are *how* and *when* personalization should take place. First, not all the data gathered to represent specific users' preferences are equally related to each of the users' queries, as users usually have multiple and diverse interests. Second, personalization is not always beneficial to the retrieval process (Teevan et al., 2008) as it could cause the information need expressed by the user to be misinterpreted by the system, thus decreasing effectiveness. A recent trend in Personalized Search (Lu et al., 2020; Zhou et al., 2020b) is *query-aware user modeling*, which consists in building a representation of the user preferences, i.e., the user model, at query time, based on various sources of user interests and by giving more importance to those related to the current search performed by the user. Since a user is typically interested in different and even *unrelated* topics, a desirable property for defining reliable personalization models is the ability to discern between beneficial and noisy user-related information on a query basis. Previous works in this context relied on the Attention mechanism (Bahdanau et al., 2015) to weigh the contribution of distinct sources of user-related information on a query basis. Despite the increasing use of the Attention mechanism in user modeling, there is still a lack of an in-depth analysis of its effects on personalization, as well as a comparison with simpler operators in this context.

In this paper, we first describe and analyze the Attention mechanism when used for query-aware user modeling, by highlighting some shortcomings related to its use of the Softmax function (Section 3). To overcome these limitations, in Section 4, we propose the Denoising Attention mechanism, an Attention variant designed to filter out noisy user-related information and produce a balanced representation of the user interests w.r.t. the current search. In Section 5, we introduce the task of Personalized Results Re-Ranking and the framework we employed in our experimental evaluation. Then, we present the research questions we addressed and describe the experimental setup (Section 6). Finally, in Section 7, we compare the Denoising Attention with other Attention variants at the state-of-the-art, evaluating both their effectiveness and their robustness. Results clearly show the advantages of Denoising Attention and the importance of the filtering mechanism it implements. We share all the code to reproduce the experimental evaluation, and we make available the implementation of Denoising Attention for future works. [1]

[1] www.github.com/AmenRa/denoising-attention

Findings of the Association for Computational Linguistics: NAACL 2024, pages 2368–2380
June 16-21, 2024 ©2024 Association for Computational Linguistics

2 Related Work

Personalization of search results has received considerable attention from both academia and industry. The definition of user models is the core issue in Personalized Search. Early user modeling approaches relied on click-based features (Bennett et al., 2012; Dou et al., 2007; Teevan et al., 2008, 2011), content-based features (Matthijs and Radlinski, 2011; Teevan et al., 2008), social network analysis (Carmel et al., 2009), language models (Tan et al., 2006; Sontag et al., 2012), topic modeling (Harvey et al., 2013; Carman et al., 2010; Xu et al., 2008), ontologies (Sieg et al., 2007; Pretschner and Gauch, 1999), and other sources of user-related information to build user representations. Researchers have also applied Representation Learning (Bengio et al., 2013) techniques to build semantic vector representations of queries, documents, and user-related information for personalizing search results (Li et al., 2014; Song et al., 2014; Zhang et al., 2020; Vu et al., 2017; Braga et al., 2023).

Several recent works employed on the Attention mechanism (Bahdanau et al., 2015) to weigh and aggregate the available user-related information on a query basis. These models take advantage of the diverse interests that a user may have to conduct query-aware personalization. For example, previous works (Ge et al., 2018; Lu et al., 2020; Yao et al., 2020b) relied on Attention to build session-based user models for personalizing subsequent searches. (Zhou et al., 2020b) employed the Attention to enhance a personalization model based on user re-finding behavior. (Zhong et al., 2020) leveraged the Attention to weigh user-related terms for Personalized Query Suggestion. (Jiang et al., 2020) proposed an *attentive* Personalized Item Retrieval model that estimates the importance of each item in the user history. Despite the increasing application of Attention for user modeling, previous publications did not present an in-depth analysis of its behavior and effects on personalization. The sole exception is represented by the Zero Attention Model (Ai et al., 2019). This Attention variant was defined to allow the retrieval model to avoid personalization when no source of user information is related to her current search, which is not possible using the standard Attention formulation, as we will discuss in Section 3.2. Despite promising results, successive works (Bi et al., 2020b,a; Jiang et al., 2020) have shown that the Zero Attention Model performs inconsistently. In this paper, we address the lack of in-depth analysis of the Attention mechanism when applied for personalization and propose a novel Attention variant to overcome some limitations highlighted by our study of such a mechanism.

3 Preliminaries on Query-aware User Modeling

Users usually have diverse interests in multiple domains. Not all those preferences are equally relevant to a specific user's information need. For example, if a user is looking for a new book to read, her fashion-related preferences probably do not matter for personalizing the results of her current query. *Query-aware User Modeling* consists in building a user model at query time, based on previously gathered sources of user interest, by giving more importance to those related to the current search performed by the user. In the literature, the definition of a user model with the previous characteristics has been provided by relying on the Attention mechanism (Bahdanau et al., 2015), which allows weighing the contribution of the user-related data w.r.t. the current search query. In this section, we first describe the Attention mechanism as it is usually employed in the context of Personalized Search. Then, we discuss some of its shortcomings when used for personalization.

3.1 Attention Mechanism

The Attention mechanism aims at computing a context vector by weighing the available contextual information w.r.t. a given input. In Personalized Search, the context vector is interpreted as the *user's context vector*, i.e., the *user model*; the contextual information is intended as the *user's contextual information*, i.e. the available user-related information, and the input is the *search query*. We assume that the user-related information is extracted from textual documents. At query time, the Attention mechanism weighs the vector representations of these documents w.r.t. the query vector and aggregates them to produce the user model employed in the personalization process. This mechanism comprises three steps aiming to build the context vector: *scoring*, *normalization*, and *aggregation*.

Scoring First of all, an *alignment model* a is used to score how well the representations of the user-related documents match with the input query:

$$e_{\boldsymbol{q},\boldsymbol{d}} = a(\boldsymbol{q}, \boldsymbol{d}) \quad (1)$$

where, $\boldsymbol{d} \in \mathbb{R}^m$ and $\boldsymbol{q} \in \mathbb{R}^n$ are the vector representations of a user document and a given query, respectively, and $e_{\boldsymbol{q},\boldsymbol{d}} \in \mathbb{R}$ is the score computed by the alignment model $a : \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}$. The alignment model can be the dot-product, the cosine similarity, or a parameterized function.

Normalization The second step of the Attention mechanism consists in the normalization of the scores computed by the alignment model to generate a probability distribution of the contextual information. The normalized scores are commonly called *attention weights*. This step is usually accomplished by employing the Softmax function:

$$\alpha(\boldsymbol{q}, \boldsymbol{d}) = \frac{\exp(e_{\boldsymbol{q},\boldsymbol{d}})}{\sum_{\boldsymbol{d'} \in \boldsymbol{D_u}} \exp(e_{\boldsymbol{q},\boldsymbol{d'}})} \tag{2}$$

where, exp is the exponential function, $\boldsymbol{D_u}$ is the set of all the documents related to the user u, and $\alpha(\boldsymbol{q}, \boldsymbol{d}) \in \mathbb{R}$ is the *attention weight* of $\boldsymbol{d}$ w.r.t. $\boldsymbol{q}$.

Aggregation The final step consists in the aggregation of the contextual information to produce the context vector $\boldsymbol{u}$, which, in our case, represents the user model. This process is carried out by summing the user document vector representations weighed by their corresponding attention weights:

$$\boldsymbol{u} = \sum_{\boldsymbol{d} \in \boldsymbol{D_u}} \alpha(\boldsymbol{q}, \boldsymbol{d}) \cdot \boldsymbol{d} \tag{3}$$

3.2 Attention-based User Modeling Shortcomings

Although the Attention mechanism allows building user models at query time, it is affected by some shortcomings when employed for personalization. Specifically, these issues are related to the use of Softmax in the normalization step. Softmax was proposed as a continuous and differentiable generalization of *Arg max* (Bridle, 1989) and is primarily employed in classifiers to compute a probability distribution over the output classes. Because of the usage of the exponential function, Softmax tends to select one among n options. Therefore, Softmax-based user modeling approaches naturally tend to skew the user representation towards a single user document, i.e., the one that best *aligns* with the query. Such characteristics are usually not ideal for personalization as also other user documents could concur to a more informed and balanced representation of the user interests and preferences. A possible solution could be to constrain the alignment model's output so that the normalization step cannot produce an *overly narrow* probability distribution of the contextual information. However, if, for example, we constrain the alignment scores near zero by using the cosine similarity as the alignment model, the Softmax normalization will *overly smooth* the attention weights, thus causing noisy information to flow into the user model. Moreover, the user information source whose alignment score with the query is negative, indicating very low relatedness, would get a positive attention weight. Lastly, as the Softmax normalizes its input into a probability distribution, it follows that the attention weights from Eq. 2 are all positive and sum up to 1. Even if all the alignment scores were zero or negative, the attention weights would *all* be positive and sum to 1. *For example*, given the following vector of alignment scores $[0.0, 0.0, 0.0, 0.0]$, by applying Eq. 2 for normalization we obtain the following attention weights $[0.25, 0.25, 0.25, 0.25]$. The same happens when all the alignment scores are negative: $[-7.0, -3.0, -1.0, -2.0] \rightarrow [0.0016, 0.0899, 0.6641, 0.2443]$. In the context of personalization, this means that the user's context vector will never be zero, causing personalization to be performed even when no source of user-related information is related to her current search. In such cases, personalization could hurt the effectiveness of the search engine instead of improving it.

4 Denoising Attention Mechanism

As extensively discussed in Section 3.2, the principal issues of the Attention are related to its normalization step, described in Section 3.1, and specifically to the use of the Softmax function to produce the attention weights. To counteract these issues, we need a mechanism able to avoid *overly narrowing* or *overly smoothing* the attention weights, which can cause the model either to focus only on a single source of user-related information or to reduce the diversity of their estimated importance. Moreover, this mechanism should finely filter out noisy contextual information, thus preventing it from flowing into the user model. Finally, it should zero out the user's context vector when personalization could harm the retrieval process, i.e., when *all* the user-related information is noisy or irrelevant with respect to the current search. In this regard, we propose the Denoising Attention mechanism. The Denoising Attention mechanism departs from the Softmax function by adopting a more straight-

forward and robust normalization scheme, and it introduces a filtering mechanism based on ReLU (Nair and Hinton, 2010) and a threshold value. To complement those changes, we rely on a cosine similarity-based alignment model to evaluate the relatedness of the sources of user-related information w.r.t. the current search.

Scoring For an alignment model to act in a complementary way with the changes introduced in the next paragraph, we need a function $a(\boldsymbol{q}, \boldsymbol{d})$ bounded between 0 and 1, as an unbounded function would make it difficult to control which information flows into the user model. To compute the alignment scores $e_{\boldsymbol{q},\boldsymbol{d}}$, we then rely on a cosine similarity-based function bounded in $[0, 1]$:

$$e_{\boldsymbol{q},\boldsymbol{d}} = a(\boldsymbol{q}, \boldsymbol{d}) = \frac{\cos(\boldsymbol{q}, \boldsymbol{d}) + 1}{2} \quad (4)$$

Filtering We propose to extend the Attention mechanism with an information filter. First, we employ a threshold t to *negativize* the alignment scores of the user data loosely related to the query:

$$shifted\_e_{\boldsymbol{q},\boldsymbol{d}} = e_{\boldsymbol{q},\boldsymbol{d}} - \sigma(t) \quad (5)$$

where σ is the Sigmoid function, which allows us to constrain t in $[0, 1]$ during training. Second, we apply ReLU to the shifted alignment scores. What makes ReLU convenient in the personalization context is its ability to zero out negative values, in our case, the scores of noisy user-related information:

$$filter\_e_{\boldsymbol{q},\boldsymbol{d}} = \mathrm{ReLU}(shifted\_e_{\boldsymbol{q},\boldsymbol{d}}) \quad (6)$$

By combining these two operations, we can both control the information flow from the user data to the user model and filter out noisy user-related information that could harm the retrieval process. To overcome the well-known dying ReLU problem (Lu et al., 2019; Agarap, 2018) , we sum the user model to the query representation during training.

Normalization The second major change we propose to the Attention is to replace Softmax by a sum based normalization of the alignment scores into attention weights:

$$\alpha(\boldsymbol{q}, \boldsymbol{d}) = \frac{filter\_e_{\boldsymbol{q},\boldsymbol{d}}}{\max\left\{\sum_{\boldsymbol{d'} \in \boldsymbol{D_u}} filter\_e_{\boldsymbol{q},\boldsymbol{d'}}, \varepsilon\right\}} \quad (7)$$

where ε is a very low positive value required to avoid numerical instability.

The proposed filtering mechanism cannot work correctly with Softmax because the latter is translationally invariant (adding or removing the same amount to the input values does not change the outputs) and zeroing out negative values does not prevent it from producing positive attention weights, as discussed in Section 3.2. By ditching Softmax, our proposal does not suffer from those issues.

Aggregation The aggregation of the user-related information is performed as in Eq. 3.

Denoising Attention Weights To sum up, we propose to compute the weights for the user-related information as follows:

$$\alpha(\boldsymbol{q}, \boldsymbol{d}) = \frac{\mathrm{ReLU}\left(e_{\boldsymbol{q},\boldsymbol{d}} - \sigma(t)\right)}{\sum_{\boldsymbol{d'} \in \boldsymbol{D_u}} \mathrm{ReLU}\left(e_{\boldsymbol{q},\boldsymbol{d'}} - \sigma(t)\right)} \quad (8)$$

In contrast with the standard Attention formulation, Denoising Attention is able to 1) selectively filter out the noisy contextual information from the user-related data before aggregating them in the context vector, and 2) zero out the context vector when all the sources of user-related information are unrelated to her current search. Moreover, the combined use of our filtering mechanism and normalization function makes our Attention variant prone to avoid *overly narrow* or *overly smooth* attention weights. This way, the model preserves the estimated importance of the user-related information sources, thus composing a balanced representation of the user preferences related to the current query while filtering those unrelated. For a sake of comparison, the alignment scores $[0.7, 0.3, 0.1, -0.2]$ produce the attention weights $[0.3809, 0.2553, 0.2090, 0.1548]$ when fed to Eq. 2, whereas they produce the attention weights $[0.75, 0.25, 0.0, 0.0]$ when fed to Eq. 8 with $\sigma(t) = 0.1$.

5 Personalized Results Re-Ranking

In this section, we introduce the task we have considered for evaluating the proposed user modeling approach, i.e. Personalized Results Re-Ranking. Moreover, we describe the personalized re-ranking framework we employed for comparative evaluation. This framework allowed us to test different user modeling techniques with ease and isolate their impact from the other system components.

In Personalized Results Re-Ranking, a retrieval system (first stage retriever) computes a ranked list of documents in response to a search query. Then, a personalization component computes new relevance scores for the initially retrieved documents

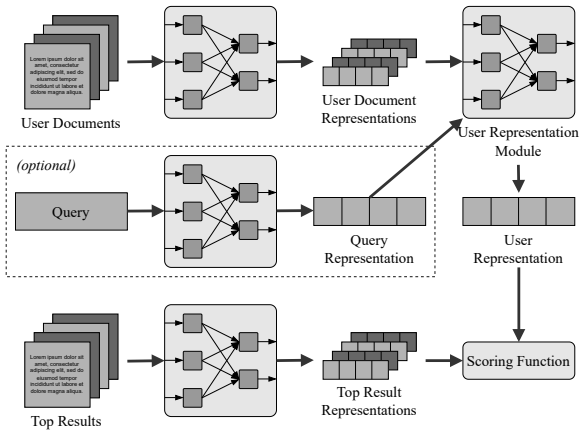


Figure 1: Personalized Results Re-Ranking Framework. The optional module is required only for query-aware user modeling.

by leveraging the user-related information. Finally, the personalized relevance scores are usually combined with those computed by the first stage retriever to re-rank the initially retrieved list of documents. Fig. 1 depicts the Personalized Results Re-Ranking Framework we relied on for comparing various user modeling techniques. The framework comprises two modules that generate the vector representations of the top-k results retrieved by the first stage retriever and those of the user-related documents. Once computed the user-related document representations, the *user representation module* aggregates them into the user model. In the case of query-aware user modeling, an additional module is employed to produce the query representation. Finally, a scoring function computes a personalized relevance score for each initially retrieved result by comparing its representation with that of the user. These scores are then combined with the first stage retriever's scores as follows:

$$final\_score = (1 - \lambda) \cdot a + \lambda \cdot b \qquad (9)$$

where, a and b are the relevance scores computed by the first stage retriever and the personalization model, respectively, and λ is a parameter that controls the influence of the two on the final score.

Albeit simple, the framework we implement for personalized re-ranking is functional to compare a user model based on Denoising Attention with state-of-the-art alternatives, isolating the user modeling approach's contribution to the overall system effectiveness. In the experiments presented in Section 7, we relied on TinyBERT (Jiao et al., 2020) followed by a mean pooling operation to embed both the top retrieved documents, the user information, and the query.

Table 1: Statistics of the employed datasets.

Web Search Dataset			
# documents	1 291 695	# users	30 166
# train queries	212 386	avg. query length	3.57 ± 1.51
# val queries	31 064	avg. relevants	1.15 ± 0.46
# test queries	36 052	avg. user docs	136.62 ± 134.17
Academic Search Dataset			
# documents	4 201 265	# users	63 738
# train queries	419 004	avg. query length	7.53 ± 2.64
# validation queries	4 241	avg. relevants	5.33 ± 5.11
# test queries	24 056	avg. user docs	53.59 ± 50.94

6 Experimental Setup

The experiments reported in this section aim to answer the following research questions:

RQ1 Are query-aware Attention-based user models more effective than static user models?

RQ2 Is Denoising Attention more effective at user modeling than other Attention variants?

RQ3 Is Denoising Attention more *robust*, i.e., less likely to decrease the system's effectiveness due to noisy user-related data, than other Attention variants?

To answer the research questions **RQ1** and **RQ2**, we conducted a comparative evaluation of the retrieval effectiveness of the personalized re-ranking pipeline described in Section 5 using several different user models. Then, to answer the research question **RQ3**, we compared the number of times the considered user models decreased the retrieval effectiveness of our first-stage retriever, BM25.

In the following, we present the datasets we employed for conducting our evaluations (Section 6.1), we introduce the baselines we have selected (Section 6.2), and we outline the training setup and evaluation procedure (Section 6.3).

6.1 Datasets

To conduct our experimental evaluations, we relied on two datasets that account for different search scenarios. We considered a Web Search dataset based on the AOL query log (Pass et al., 2006) and a synthetic dataset built following the procedure described by (Tabrizi et al., 2018) that simulates an Academic Search scenario. We describe both datasets in detail in the following sections.

6.1.1 Web Search Dataset

The AOL query log is one of the most known large-scale set of data for the evaluation of *session-based* personalization models (Ahmad et al., 2018, 2019; Yao et al., 2020a; Zhou et al., 2020a; Lu et al., 2020; Zhou et al., 2021; Yao et al., 2020b, 2022; Deng et al., 2022).

Retrieving documents and query logs As the documents are not provided with the query logs, we relied on *aolia-tools* (MacAvaney et al., 2022), which leverage the Internet Archive's Wayback Machine service, to retrieve contents similar to those seen by the users when the logs were collected. We identified and removed non-English documents by analyzing them using Google's CLD v3[2]. We discarded all the queries without related clicks, and those containing Internet domain references (e.g., *.com*, *.org*, etc.) or website names and queries shorter than three characters. For ethical reasons, we also discarded all the queries containing or pointing to adult or illegal contents. We removed non-alphanumeric characters from the queries, applied a spelling corrector (*SymSpell*[3]). To avoid introducing in the test set $\langle$*query, user, document*$\rangle$ triplets also present in the train set, we kept only the first appearance of such triplets by comparing their associated timestamps.

Training / Validation / Test Splits Following previous works (Sordoni et al., 2015; Ahmad et al., 2019), we considered the queries formulated in the first five weeks as a background set. We discarded all the queries from users with less than 20 associated queries in this set to ensure having enough data to conduct personalization. We then temporally split the remaining weeks' worth of queries. We used six weeks for training queries, one week for *validation* queries, and one week for test queries.

6.1.2 Academic Search Dataset

Due to the lack of a publicly available Domain-specific Search dataset for studying personalization, researchers have recently tackled personalization in Product Search scenarios relying on synthetic datasets built upon product reviews from a popular e-commerce platform (Ai et al., 2017). However, due to the low number of different queries present in these datasets, and their low quality (Bassani and Pasi, 2022), we did not employ them in our comparative evaluation. Instead, we followed the procedure described in (Tabrizi et al., 2018) to build an Academic Search dataset that allow us to test our Attention variant in a domain-specific search scenario. In particular, we relied on the *ArnetMiner*'s *Citation Network Dataset V12* (Tang et al., 2008), which makes available the metadata of $4\,894\,081$ academic papers.

Query Generation Following the approach described by (Tabrizi et al., 2018), we generated user-query-document triplets as follows: for each academic paper, we considered its title as a query, the list of its citations as the documents relevant to that query, and we assumed that the first author is the user issuing the query. We applied stop-word removal using the *NLTK*'s stop-words list and the *Krovetz stemmer*, to obtain queries that resemble real-world ones. Finally, we discarded all the generated queries whose related users have less than 20 associated documents, i.e., published papers. More details can be found in (Bassani et al., 2022).

Training / Validation / Test Splits We split the obtained dataset into training and test sets chronologically, i.e. by using the queries generated from papers published *after* 2018 as the test set. We then randomly split the training set to obtain a training set and a validation set, using a splitting ratio of $99:1$. We opted for a chronological training / test split instead of a random partitioning so that the dataset is closer to a real scenario, where all the searches in the test set happen after the searches in the training set. As we are interested only in results re-ranking, in both datasets, we discarded the queries for which BM25 does not retrieve any relevant document in the top 1000 results and we retain only the relevant documents present in the top 1000 results retrieved by BM25.

6.2 Baselines

In this section, we introduce the baselines employed in our comparative evaluation.
Attention: query-aware user model based on the standard Attention formulation.
Zero Attention: query-aware user model based on the Zero Attention strategy (Ai et al., 2019).
Multi-Head Attention: query-aware user model based on the Multi-Head Attention (Vaswani et al., 2017) with four Attention heads.
Mean: *static* user model that computes user representations as the arithmetic mean of the user-related documents' representations.

[2] https://github.com/google/cld3
[3] https://github.com/wolfgarbe/SymSpell

BM25: for reference, we also performed comparison with BM25 (Robertson and Walker, 1994), out first stage retriever.

We trained three variants for both the Attention-based and the Zero Attention-based user models by employing different alignment functions. The first variant employs the scaled-dot product, which relies on a temperature based softmax, popularized by the Transformer architecture (Vaswani et al., 2017). The second one uses the cosine similarity, similarly to our Denoising Attention. The last one relies on Additive Attention, a parametrized model (Bahdanau et al., 2015)

6.3 Setup & Evaluation Metrics

We relied on ElasticSearch for BM25, HuggingFace's Transformers for TinyBERT , and PyTorch for the implementation of all the neural models. We optimized BM25's $k1$ and b parameters on non-test data. BM25 scores were computed on the concatenation of documents' title and abstract. The training was done on an NVidia® RTX 2080 Ti GPU for 20 epochs using a hinge loss (Gao et al., 2021), with a margin of 0.1 and AdamW optimizer with learning rate of 5×10^{-5}, and batch size of 32. We train the model with hard negatives sampled from the top results retrieved by BM25 and in-batch random negatives. During training we randomly sampled 20 user documents to use for personalization, while during the evaluation, we used all the available user documents. After training, we fine-tuned the λ parameter of Eq. 9 and the Denoising Attention's threshold on the validation set. The re-ranking was done on the top 1000 results retrieved by BM25.

To evaluate the effectiveness of the compared models, we employed Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and Normalized Discounted Cumulative Gain (NDCG). MRR and NDCG were computed on the top 10 documents retrieved by each model, whereas MAP was computed on the top 100. Metrics computation and comparison were conducted using the Python library `ranx` (Bassani, 2022; Bassani and Romelli, 2022; Bassani, 2023).

7 Results and Discussion

In this section, we present the results of our comparative evaluation. First, we discuss the retrieval effectiveness of the personalized re-ranking pipeline described in Section 5 when considering different user modeling techniques. Then, we analyze the robustness of the compared user models, evaluating the probability they decrease the system's effectiveness in the presence of noisy user-related data. We remind the reader that the only difference between the compared personalization models is the technique used for defining the user model, while the other system's components are fixed.

Table 2: Effectiveness of all models. $*$ and † denote significant improvements in a Bonferroni corrected Fisher's randomization test with $p < 0.001$ over *Mean* and over all the baselines, respectively. Best results are highlighted in boldface.

Web Search Dataset						
Model	**Alignment**	**MAP@100**	**MRR@10**	**NDCG@10**	λ	$\sigma(t)$
BM25	—	0.245	0.238	0.280	—	—
Mean	—	0.282	0.276	0.329	0.2	—
Attention	Additive	0.281	0.276	0.328	0.2	—
	Cosine	0.287*	0.281*	0.335*	0.2	—
	Scaled-Dot	0.290*	0.285*	0.339*	0.2	—
Zero Attention	Additive	0.277	0.272	0.325	0.2	—
	Cosine	0.286*	0.281*	0.334*	0.2	—
	Scaled-Dot	0.290*	0.285*	0.338*	0.2	—
Multi-Head	Scaled-Dot	0.275	0.269	0.324	0.2	—
Denoising	Cosine-based	**0.338**†	**0.336**†	**0.393**†	0.4	0.7
Academic Search Dataset						
Model	**Alignment**	**MAP@100**	**MRR@10**	**NDCG@10**	λ	$\sigma(t)$
BM25	—	0.119	0.294	0.171	—	—
Mean	—	0.146	0.328	0.200	0.6	—
Attention	Additive	0.156*	0.340*	0.213*	0.6	—
	Cosine	0.151	0.332	0.206	0.6	—
	Scaled-Dot	0.157*	0.343*	0.214*	0.6	—
Zero Attention	Additive	0.155*	0.338	0.211*	0.6	—
	Cosine	0.150	0.330	0.204	0.6	—
	Scaled-Dot	0.156*	0.341*	0.212*	0.6	—
Multi-Head	Scaled-Dot	0.152	0.336	0.207	0.6	—
Denoising	Cosine-based	**0.179**†	**0.378**†	**0.241**†	0.6	0.6

7.1 Retrieval Effectiveness

As reported in Table 2, personalization improved the retrieval effectiveness of our first stage retriever, BM25, regardless of the user modeling mechanism employed, thus confirming the utility of personalization for both the considered datasets. Among the Attention-based baselines, only those employing the scaled-dot alignment model significantly improved over Mean on both the considered datasets. Those relying on the additive and the cosine alignment models achieved mixed results, sometimes even decreasing w.r.t. Mean. Despite the fact that it was introduced to overcome some of the Attention shortcomings, the Zero Attention-based user models generally achieved slightly worse results than their standard Attention-based counterparts. In this regard, our findings are consistent with results from previous works (Bi et al., 2020b,a; Jiang

et al., 2020). The results obtained by the standard Attention and the Zero Attention-based user models with cosine similarity as the alignment model show that constraining the alignment scores causes noisy information to leak into the user model, as discussed in Section 3.2. Finally, the Multi-Head Attention-based user model's results are among the lowest for both datasets. The additional complexity introduced by this approach did not deliver improvements over the other Attention-based models while introducing additional overhead. If we consider only the Attention-based user model with the scaled-dot alignment model, the obtained results positively answer our first research question, **RQ1**. However, this is not the case for all the other Attention baselines, which confirms the need for our investigation on the use of the Attention mechanism for query-aware personalization.

When employing the Denoising Attention-based user model, the results re-ranking pipeline achieved substantial improvements over all baselines, corroborating our intuitions about the shortcomings of the standard Attention formulation when it comes to personalization and the advantages brought by our proposal. In particular, Denoising Attention improves over the best-performing baseline of about 15% for each metric on both datasets. The obtained results clearly show the robustness of our proposed Attention variant to search scenarios with noticeable structural differences. For Web Search, it is fundamental to finely select the most promising user-related data for conducting personalization in order to improve over simple operations for building user models, such as averaging over the representations of the user-related data. In the case of Academic Search, user-related information is very focused and, therefore, it is easier to improve a user model that averages the representations of the user-related data. Nonetheless, Denoising Attention still exhibits significant advantages over the other Attention variants. The λ parameter has a huge impact on the final performances. In order to remove the contribution of the first stage ranker, we set the λ parameter to 1.0. In this experiment, we consider only the queries for which Denoising Attention outputs a non-zero user model and employs only the scores deriving from the comparisons between the user models and the documents to re-rank the initially retrieved BM25 result lists. The results are reported in Table 3. In the best case scenario (*Scaled-dot*), the Attention-based baselines increased over *Mean* by 11%, 13%, and 11% in MAP, MRR, and NDCG, respectively, on the *Web Search Dataset*, and by 34%, 30%, and 34% in MAP, MRR, and NDCG, respectively, on the *Academic Search Dataset*.

These results, which positively answer our second research question, **RQ2**, highlight the importance of correctly managing the user-related information in personalization and the potential of deepening this research area.

Table 3: Effectiveness of BM25 and those of the user models when used in isolation. $*$ and † denote significant improvements in a Bonferroni corrected Fisher's randomization test with $p < 0.001$ over *Mean* and over all the baselines, respectively. Best results are highlighted in boldface.

Web Search Dataset						
Model	**Alignment**	**MAP@100**	**MRR@10**	**NDCG@10**	λ	$\sigma(t)$
BM25	—	0.240*	0.233*	0.274*	—	—
Mean	—	0.136	0.120	0.157	1.0	—
Attention	Additive	0.136	0.120	0.155	1.0	—
	Cosine	0.141*	0.125*	0.166*	1.0	—
	Scaled-Dot	0.152*	0.137*	0.177*	1.0	—
Zero Attention	Additive	0.125	0.108	0.144	1.0	—
	Cosine	0.148*	0.132*	0.169*	1.0	—
	Scaled-Dot	0.153*	0.138*	0.177*	1.0	—
Multi-Head	Scaled-Dot	0.128	0.111	0.148	1.0	—
Denoising	Cosine-based	**0.264**†	**0.256**†	**0.312**†	1.0	0.7
Academic Search Dataset						
Model	**Alignment**	**MAP@100**	**MRR@10**	**NDCG@10**	λ	$\sigma(t)$
BM25	—	0.120*	0.295*	0.172*	—	—
Mean	—	0.068	0.160	0.094	1.0	—
Attention	Additive	0.090*	0.205*	0.123*	1.0	—
	Cosine	0.076*	0.172*	0.103*	1.0	—
	Scaled-Dot	0.091*	0.208*	0.125*	1.0	—
Zero Attention	Additive	0.086*	0.195*	0.117*	1.0	—
	Cosine	0.075*	0.171*	0.103*	1.0	—
	Scaled-Dot	0.088*	0.201*	0.120*	1.0	—
Multi-Head	Scaled-Dot	0.074*	0.172*	0.101*	1.0	—
Denoising	Cosine-based	**0.143**†	**0.319**†	**0.194**†	1.0	0.6

7.2 Robustness

As shown in Table 4, to evaluate the *robustness* of the considered user models, we considered the number of times personalization decreased BM25 effectiveness in terms of MAP@100. Quite surprisingly, the Attention-based user models are often more harmful than Mean, although more effective in general, as previously reported. Conversely, the Denoising Attention-based user model is substantially less harmful than all the other user models on both datasets. Compared to the Denoising Attention-based user model, the best baselines on the Web Search Dataset and the Academic Search Dataset decreased the retrieval effectiveness of BM25 for 38% and 8% more queries, respectively. The much more significant difference we

Table 4: Number of times (and ratios) personalization decreased BM25 effectiveness in terms of MAP@100 (lower is better). Best results are in boldface.

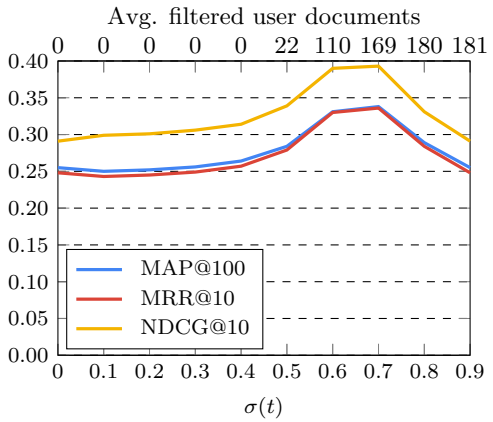
Model	Alignment	Web Search Dataset	Academic Search Dataset
Mean	—	10 798 (30%)	6 165 (26%)
Attention	Additive	11 157 (31%)	6 076 (25%)
	Cosine	9 877 (27%)	6 580 (27%)
	Scaled-Dot	9 426 (26%)	5 954 (25%)
Zero Attention	Additive	11 508 (32%)	6 201 (26%)
	Cosine	10 234 (28%)	6 708 (28%)
	Scaled-Dot	9 356 (26%)	6 131 (25%)
Multi-Head	Scaled-Dot	12 049 (33%)	6 366 (26%)
Denoising	Cosine-based	**6 780 (19%)**	**5 509 (23%)**

registered on the Web Search Dataset than the Academic Search Dataset is due to the different nature of those datasets. In the former dataset, user-related data accounts for the many different interests each user may have. Thus, personalization is likely to harm the retrieval process if a filtering mechanism for the user information is not employed. In the latter dataset, user preferences are limited to fewer topics. Given the obtained results, we conclude that the Denoising Attention-based user model is much more robust than the other considered user models regardless of the search scenario, positively answering our research question **RQ3**.

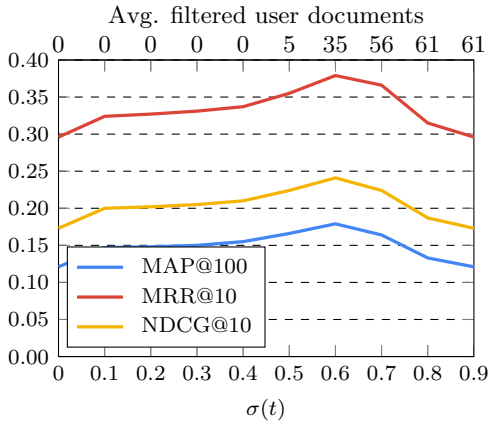
7.3 Model Analysis

In this section, we evaluate the Denoising Attention-based user model performances for various threshold values.

Figures 2a and 2b show the performances of the results re-ranking pipeline with the Denoising Attention-based user model for different threshold values on the considered datasets. The figures also report the average number of filtered user documents for each considered threshold value. On average, the test queries have 181 and 61 associate user-related documents in the *Web Search Dataset* and the *Academic Search Dataset*, respectively, while the average number of filtered ones for the best threshold values are 169 and 35, respectively. The different ratios of average filtered user-related documents are again due to the distinct nature of the two search scenarios and datasets. Our proposed approach is able to adapt to different search contexts thanks to the threshold parameter and our filtering mechanism. When the threshold is zero, which corresponds to not filtering any user-related document in our case, the model effectiveness is very low for both datasets. When the threshold is equal to 0.5, which corresponds to using the cosine similarity with no modification as our alignment model, the model still does not reach its full potential. These results highlight again the need for a filtering mechanism that can be tuned and modulated.



(a) Retrieval effectiveness of the personalized re-ranking pipeline with Denoising Attention-based user model on the *Web Search Dataset*



(b) Retrieval effectiveness of the personalized re-ranking pipeline with Denoising Attention-based user model on the *Academic Search Dataset*

Figure 2: Threshold analysis.

8 Conclusion

In this work, we have addressed some issues related to the use of the Attention mechanism for query-aware user modeling and proposed a novel user-data aggregation model called Denoising Attention, designed to solve the shortcomings of the standard Attention formulation and, in particular, filter out noisy user-related information. Experimental evaluation in two different search scenarios, namely Web Search and Academic Search, shows the benefits of our proposed approach over other Attention variants and highlights the potential of correctly managing the user-related information.

9 Limitations

Despite the significant improvements brought by our proposed Denoising Attention mechanism when in comes to query-aware personalization, some related problems are worth further study. The alignment model we employed, may be replaced by a parameterized function that could leverage additional information other than the representations of user-related documents and queries. For example, the dates associated with those documents might play a role in personalization, as documents written or consulted long before the query might be less relevant to personalization than more recent ones. Furthermore, the fixed value threshold parameter we employed could be sub-optimal in many cases. As shown by the difference in the threshold parameter values for the two considered datasets, different queries could benefit from more user-related information or require a finer selection of the user-related data employed in the personalization process. To conclude, the management of the user-related information during personalization is fundamental and far from being a solved issue, leaving room for further improvements.

References

Abien Fred Agarap. 2018. Deep learning using rectified linear units (relu).

Wasi Uddin Ahmad, Kai-Wei Chang, and Hongning Wang. 2018. Multi-task learning for document ranking and query suggestion. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Wasi Uddin Ahmad, Kai-Wei Chang, and Hongning Wang. 2019. Context attentive document ranking and query suggestion. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pages 385–394. ACM.

Qingyao Ai, Daniel N. Hill, S. V. N. Vishwanathan, and W. Bruce Croft. 2019. A zero attention model for personalized product search. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 379–388. ACM.

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, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 645–654. ACM.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Elias Bassani. 2022. ranx: A blazing-fast python library for ranking evaluation and comparison. In *Advances in Information Retrieval - 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10-14, 2022, Proceedings, Part II*, volume 13186 of *Lecture Notes in Computer Science*, pages 259–264. Springer.

Elias Bassani. 2023. ranxhub: An online repository for information retrieval runs. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 3210–3214. ACM.

Elias Bassani, Pranav Kasela, Alessandro Raganato, and Gabriella Pasi. 2022. A multi-domain benchmark for personalized search evaluation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, page 3822–3827, New York, NY, USA. Association for Computing Machinery.

Elias Bassani and Gabriella Pasi. 2022. A multi-representation re-ranking model for personalized product search. *Inf. Fusion*, 81:240–249.

Elias Bassani and Luca Romelli. 2022. ranx.fuse: A python library for metasearch. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 4808–4812. ACM.

Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. 2013. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):1798–1828.

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 *The 35th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '12, Portland, OR, USA, August 12-16, 2012*, pages 185–194. ACM.

Keping Bi, Qingyao Ai, and W. Bruce Croft. 2020a. A review-based transformer model for personalized product search. *CoRR*, abs/2004.09424.

Keping Bi, Qingyao Ai, and W. Bruce Croft. 2020b. A transformer-based embedding model for personalized product search. In *Proceedings of the 43rd International ACM SIGIR conference on research*

and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020, pages 1521–1524. ACM.

Marco Braga, Alessandro Raganato, and Gabriella Pasi. 2023. Personalization in bert with adapter modules and topic modelling. In *Italian Information Retrieval Workshop*.

John S. Bridle. 1989. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In *Advances in Neural Information Processing Systems 2, [NIPS Conference, Denver, Colorado, USA, November 27-30, 1989]*, pages 211–217. Morgan Kaufmann.

Mark James Carman, Fabio Crestani, Morgan Harvey, and Mark Baillie. 2010. Towards query log based personalization using topic models. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010*, pages 1849–1852. ACM.

David Carmel, Naama Zwerdling, Ido Guy, Shila Ofek-Koifman, Nadav Har'El, Inbal Ronen, Erel Uziel, Sivan Yogev, and Sergey Chernov. 2009. Personalized social search based on the user's social network. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009, Hong Kong, China, November 2-6, 2009*, pages 1227–1236. ACM.

Chenlong Deng, Yujia Zhou, and Zhicheng Dou. 2022. Improving personalized search with dual-feedback network. In *WSDM '22: The Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event / Tempe, AZ, USA, February 21 - 25, 2022*, pages 210–218. ACM.

Zhicheng Dou, Ruihua Song, and Ji-Rong Wen. 2007. A large-scale evaluation and analysis of personalized search strategies. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 581–590. ACM.

Luyu Gao, Zhuyun Dai, Tongfei Chen, Zhen Fan, Benjamin Van Durme, and Jamie Callan. 2021. Complement lexical retrieval model with semantic residual embeddings. In *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part I*, volume 12656 of *Lecture Notes in Computer Science*, pages 146–160. Springer.

Songwei Ge, Zhicheng Dou, Zhengbao Jiang, Jian-Yun Nie, and Ji-Rong Wen. 2018. Personalizing search results using hierarchical RNN with query-aware attention. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 347–356. ACM.

Jiafeng Guo, Yixing Fan, Liang Pang, Liu Yang, Qingyao Ai, Hamed Zamani, Chen Wu, W. Bruce Croft, and Xueqi Cheng. 2020. A deep look into neural ranking models for information retrieval. *Inf. Process. Manag.*, 57(6):102067.

Morgan Harvey, Fabio Crestani, and Mark James Carman. 2013. Building user profiles from topic models for personalised search. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, pages 2309–2314. ACM.

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 *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 2870–2877. ACM / IW3C2.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4163–4174. Association for Computational Linguistics.

Pranav Kasela, Gabriella Pasi, Raffaele Perego, and Nicola Tonellotto. 2024. Desire-me: Domain-enhanced supervised information retrieval using mixture-of-experts. In *Advances in Information Retrieval*, pages 111–125, Cham. Springer Nature Switzerland.

Xiujun Li, Chenlei Guo, Wei Chu, Ye-Yi Wang, and Jude Shavlik. 2014. Deep learning powered in-session contextual ranking using clickthrough data. In *In Proc. of NIPS*.

Lu Lu, Yeonjong Shin, Yanhui Su, and George E. Karniadakis. 2019. Dying relu and initialization: Theory and numerical examples. *CoRR*, abs/1903.06733.

Shuqi Lu, Zhicheng Dou, Chenyan Xiong, Xiaojie Wang, and Ji-Rong Wen. 2020. Knowledge enhanced personalized search. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 709–718. ACM.

Sean MacAvaney, Craig Macdonald, and Iadh Ounis. 2022. Reproducing personalised session search over the AOL query log. In *Advances in Information Retrieval - 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10-14, 2022, Proceedings, Part I*, volume 13185 of *Lecture Notes in Computer Science*, pages 627–640. Springer.

Nicolaas Matthijs and Filip Radlinski. 2011. Personalizing web search using long term browsing history. In *Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM 2011,*

Hong Kong, China, February 9-12, 2011, pages 25–34. ACM.

Vinod Nair and Geoffrey E. Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 807–814. Omnipress.

Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A picture of search. In *Proceedings of the 1st International Conference on Scalable Information Systems, Infoscale 2006, Hong Kong, May 30-June 1, 2006*, volume 152 of *ACM International Conference Proceeding Series*, page 1. ACM.

Alexander Pretschner and Susan Gauch. 1999. Ontology based personalized search. In *11th IEEE International Conference on Tools with Artificial Intelligence, ICTAI '99, Chicago, Illinois, USA, November 8-10, 1999*, pages 391–398. IEEE Computer Society.

Stephen E. Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum)*, pages 232–241. ACM/Springer.

Ahu Sieg, Bamshad Mobasher, and Robin D. Burke. 2007. Web search personalization with ontological user profiles. In *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, CIKM 2007, Lisbon, Portugal, November 6-10, 2007*, pages 525–534. ACM.

Yang Song, Hongning Wang, and Xiaodong He. 2014. Adapting deep ranknet for personalized search. In *Seventh ACM International Conference on Web Search and Data Mining, WSDM 2014, New York, NY, USA, February 24-28, 2014*, pages 83–92. ACM.

David A. Sontag, Kevyn Collins-Thompson, Paul N. Bennett, Ryen W. White, Susan T. Dumais, and Bodo Billerbeck. 2012. Probabilistic models for personalizing web search. In *Proceedings of the Fifth International Conference on Web Search and Web Data Mining, WSDM 2012, Seattle, WA, USA, February 8-12, 2012*, pages 433–442. ACM.

Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, pages 553–562. ACM.

Shayan A. Tabrizi, Azadeh Shakery, Hamed Zamani, and Mohammad Ali Tavallaei. 2018. PERSON: personalized information retrieval evaluation based on citation networks. *Inf. Process. Manag.*, 54(4):630–656.

Bin Tan, Xuehua Shen, and ChengXiang Zhai. 2006. Mining long-term search history to improve search accuracy. In *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, August 20-23, 2006*, pages 718–723. ACM.

Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*, pages 990–998. ACM.

Jaime Teevan, Susan T. Dumais, and Daniel J. Liebling. 2008. To personalize or not to personalize: modeling queries with variation in user intent. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008*, pages 163–170. ACM.

Jaime Teevan, Daniel J. Liebling, and Gayathri Ravichandran Geetha. 2011. Understanding and predicting personal navigation. In *Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM 2011, Hong Kong, China, February 9-12, 2011*, pages 85–94. ACM.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Thanh Vu, Dat Quoc Nguyen, Mark Johnson, Dawei Song, and Alistair Willis. 2017. Search personalization with embeddings. In *Advances in Information Retrieval - 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, April 8-13, 2017, Proceedings*, volume 10193 of *Lecture Notes in Computer Science*, pages 598–604.

Shengliang Xu, Shenghua Bao, Ben Fei, Zhong Su, and Yong Yu. 2008. Exploring folksonomy for personalized search. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008*, pages 155–162. ACM.

Jing Yao, Zhicheng Dou, and Ji-Rong Wen. 2020a. Employing personal word embeddings for personalized search. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 1359–1368. ACM.

Jing Yao, Zhicheng Dou, and Ji-Rong Wen. 2022. Clarifying ambiguous keywords with personal word embeddings for personalized search. *ACM Trans. Inf. Syst.*, 40(3):43:1–43:29.

Jing Yao, Zhicheng Dou, Jun Xu, and Ji-Rong Wen. 2020b. Rlper: A reinforcement learning model for personalized search. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 2298–2308. ACM / IW3C2.

Han Zhang, Songlin Wang, Kang Zhang, Zhiling Tang, Yunjiang Jiang, Yun Xiao, Weipeng Yan, and Wenyun Yang. 2020. Towards personalized and semantic retrieval: An end-to-end solution for e-commerce search via embedding learning. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 2407–2416. ACM.

Jianling Zhong, Weiwei Guo, Huiji Gao, and Bo Long. 2020. Personalized query suggestions. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 1645–1648. ACM.

Yujia Zhou, Zhicheng Dou, and Ji-Rong Wen. 2020a. Encoding history with context-aware representation learning for personalized search. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 1111–1120. ACM.

Yujia Zhou, Zhicheng Dou, and Ji-Rong Wen. 2020b. Enhancing re-finding behavior with external memories for personalized search. In *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, pages 789–797. ACM.

Yujia Zhou, Zhicheng Dou, Yutao Zhu, and Ji-Rong Wen. 2021. PSSL: self-supervised learning for personalized search with contrastive sampling. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, pages 2749–2758. ACM.