INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 20, Issue: 4, Month: August, Year: 2025
Article Number: 6698, https://doi.org/10.15837/ijccc.2025.4.6698

CCC Publications

GBPS: A Group-Enhanced Neural Framework for Personalized Search

S. Wang, Z. Yuan, H. Zhang, B. Yang

Shijun Wang
School of Information
Renmin University of China, China
St. Zhongguancun 59, Haidian District, Beijing 100872, China
wangshijun@ruc.edu.cn

Zhe Yuan*
The Country and Area Studies Academy
Beijing Foreign Study University, China
No.2 Xisanhuan North Road, Haidian District, Beijing 100089, China
*Corresponding author: yuanzhe@bfsu.edu.cn

Han Zhang
China Unicom Research Institute
No.9 Shouti South Road, Haidian District, Beijing 100037, China
zhangh1550@chinaunicom.cn

Bin Yang
China Unicom Research Institute
No.9 Shouti South Road, Haidian District, Beijing 100037, China
researcher_yang@outlook.com

Abstract

Personalized search aims to improve user experience by tailoring search results to individual preferences. This paper introduces GBPS, a novel group-based personalized search framework that enhances search relevance by dynamically incorporating group user information. Unlike traditional approaches that rely on static, query-independent groupings based on shared clicks, GBPS forms user groups based on representation similarity, capturing deeper semantic connections between users. A hierarchical recurrent neural network (RNN) is employed to model user profiles, and multiple grouping strategies are explored to optimize representation similarities. Experiments on the AOL search log demonstrate that GBPS significantly outperforms some personalized search methods, with improvements of 1.65% in MAP, 1.65% in MRR, and 1.47% in P@1. These results highlight the potential of dynamically leveraging group information to enhance personalized search performance and address the limitations of static grouping methods.

Keywords: information retrieval, personalized search, grouping method.

1 Introduction

Issuing queries in a Web search engine is one of the most common approaches to obtain information for us in daily life. Given the same query, many search engines return the same result documents list regardless of which user issued the query. However, the same query may reveal completely different information needs for different users, especially for ambiguous queries [13, 29, 43]. This kind of situation is caused not only by the difference between users' intents when they enter the same query, but also by many unavoidable ambiguous words existing in natural language. For this reason, search personalization has been proposed. It aims to improve result quality and user satisfaction by returning more specific ranking results to each user based on the user's interests. Over these years, personalized search has been proved to give better document reranking list and promote user experience.

Many existing search results personalization models depend on click-based, topic-based and topology-based features extracted from the user's search history to analyze user interests [3, 4, 7, 8, 9, 10, 13, 15, 20, 26, 30, 33, 34, 36, 38, 42, 44]. Specifically, when a user issues a query, the user's previous search history log is used to help search result personalization. One's search history log contains a sequence of queries and clicked documents, clearly not all of the history interactions contribute the same to personalized reranking candidate documents. In fact, consistent with intuition, queries and clicked documents in the user's search history that are more similar to the current query often play a more important role in reranking documents lists.

However, one's effective search history that can contribute to search result personalization is sometimes sparse in practice. The lack of effectiveness can be caused by at least two aspects. First, for some users, their search history logs are not adequate for extracting sufficient features. Second, for some newly issued queries, their frequencies are low, and they are rarely or even never found in one's history log.

Previous studies show that a user profile can be enriched by using data from a group of users who share common interests [13, 17, 28, 31, 35, 38]. Generally, they use manually designed rules like commonly clicked documents to determine the user group. Despite achieving certain improvements, there still exist some drawbacks in these grouping methods. They are only able to get user interests covered in these limited features but miss other important semantic similarity information.

In this paper, we attempt to solve the problem by introducing a group of similar users' search histories to extract features when search results personalization is applied. Based on natural intuition, we assume that users similar to each other tend to act alike. Thus, if we can find a group of users who are similar to the current user, we can then use their search history logs to enhance the current user's search interaction history.

Specifically, we design a **G**roup-**B**ased **P**ersonalized **S**earch model, named GBPS, which take advantages of group users to promote personalized search. GBPS uses a hierarchical recurrent neural network (HRNN) [14] to sequentially build user profiles based on search history logs. Meanwhile, GBPS contains a set of user profiles representing users' long-term interests, and update it periodically. Given a target user's search history and her current query, GBPS finds several groups of users for the target user based on representation vector similarity. Then, multi-head attention is applied to combine group user profile representations. Afterward, GBPS matches them with the candidate documents. Finally, GBPS makes use of information provided by group users, together with other original features, to calculate the matching score between current query and candidate documents. By the way, the built-in HRNN model can be replaced by other personalized search models, as long as it can use search history to sequentially model user profiles.

The main contributions of our work are summarized as follows: (1) We attempt to boost the effect of search result personalization by introducing dynamically found group of users. (2) We propose a search personalization framework, GBPS, that can take advantage of a group of users' search history, as well as exploit other non-group-based personalization models. (3) We compare and analyze the effectiveness of using different groups of users when reranking documents.

The rest of the paper is organized as follows. Related works, splited into personalized search and group-based search, are introduced in Section 2. In Section 3, we introduce GBPS framework and elaborate each component. We introduce the dataset, the baselines used in our experiments, and detailed experimental settings in Section 4. In Section 5, we present and analyze the experimental results. We conclude our work and discuss about future work in Section 6.

2 Related Work

2.1 Personalized Search

Personalized search has been proved to effectively improve the quality of search engines [8]. The main aim of personalized search is to re-rank the results to meet the individual needs of different users, depending on the user's interests. The methods to represent and model user interests vary in different personalization algorithms. However, the general and common idea of them is to model users' preferences based on users' search behavior history.

In many traditional personalized search methods [3, 4, 7, 8, 9, 13, 15, 20, 26, 29, 30, 33, 34, 38], some heuristic rules are manually designed and applied on users' search behavior history to extract personalized features. These personalized features are exploited to analyze users' interests and preferences. Specifically, some works [13, 30] studied personalized click features. Some other works [4, 9, 15, 20, 26, 33, 34, 37, 38] extracted topics features or enhanced sequence representation from user's search history to predicted document relevance. Click features and topic features are combined and studied in some researches [3, 7, 38]. Bennett et al. [3] proposed SLTB, in which a lot of manually extracted features are combined including of the two types of features.

The key to personalized search is to model user's interests and preferences so as to clarify the user's information need behind current query. Provided with user's search history, traditional personalized search methods mainly focus on manually extracting personal features from user's own search history.

These years, deep learning has been used in personalized search [12, 14, 16, 18, 19, 21, 27, 40, 41, 46, 47, 48]. Deep learning can alleviate the problems with manually designed features, because of its powerful representation learning and interaction modeling capabilities. A hierarchical RNN (HRNN) [14] was proposed to model user's long-term interests and short-term interests separately and sequentially. HTPS [46] changed RNN in HRNN to term level transformer so as to learn context-aware representation from user search history. Apart from focusing on modeling users' search history representation, there are some works aimed at applying deep learning framework to personalized search. PSGAN [18] was a generative adversarial network framework to promote the training of deep personalized models which further enhances the personalized effect. Reinforcement learning was also applied in RLPer [41], whose authors believed that modeling user interests dynamically and updating the ranking strategy continuously would be helpful for personalization.

Aforementioned deep learning approaches no longer depend on manually designed features, however, still mine information from user's own limited search history. In this paper, we argue that it would be helpful for personalization if we exploit information from a group users similar with the target user.

2.2 Group-based Search

In most web search approaches, only the information provided by the user herself is used to create user profiles. There are also some methods [13, 17, 28, 31, 35, 38] that incorporate the preferences of a group of users to accomplish personalized search.

In [17], users' profile are constructed based on the modified collaborative filtering algorithm [6], so that a group of users can be figured out. CubeSVD [28] analyzed the correlation among users, queries, and web pages by sparse click-through data to improve personal web search. Dou et al. introduced G-Click model [13] which also uses click-through data and incorporates click histories of a group of users to personalized web search. In [31], the authors explored and tested different attributes to discover user group that is more valuable for personalized search. White et al. modeled search tasks identified within search sessions, and then mined other users' search behavior to complement and enhance search personalization [38]. However, the above works neglect the fact that users in a group may have different interests concerning different queries. Vu et al. [35] employed Latent Dirichlet Allocation (LDA) [5] to construct user profiles and dynamically form users grouped dependent on the input query to enrich the target user's profile.

Intuitively, we presume that dynamically grouping users can achieve better performance. In this paper, we explore the effectiveness of different user grouping methods, with and without respect to input query.

3 GBPS - A Framework for Group-Based Personalized Search

Personalized search models are required to tailor the candidate documents list taking user interests into account, and give higher priority to the documents that match the user's query intent and interests. As we stated in Section 2, most existing personalized search model uses extract features only from the target user. The minority models using group users are modeling users search history in a term-level straightforward way, such as mining re-finding behaviors in others' search history. Differently, we propose GBPS framework from the perspective of representation-based dynamic group formation to tackle this problem.

The GBPS framework uses a portable model that represent user search history as an inner model, and it can enhance this inner model with its powerful groupization. In our work, we extend HRNN [14] model for adjusting group formation, and choose this Extended-HRNN model to be the inner model, since the HRNN model is lightweight and easy to apply. Note that, as long as a model can embed user search history into vector representation, it can be extended and then served as the portable inner model.

Specifically, we design Extended-HRNN model as an extension of the HRNN model. We apply it as the non-group personalized search model wrapped in our GBPS framework. While user profile has only 2 forms in HRNN, short-term interests and long-term interests both with query-aware attention, user profile has 4 forms in Extended-HRNN as detailed described in 3.2.

Thus, the overall process of the GBPS framework is as follows: given a user and a newly issued query, the portable inner model (Extended-HRNN in our experiment) constructs the target user's four types of profile, that is long-term and short-term interests both with and without the current query-aware attention for the target user. Applying these four types of user profile and the current query as five groupization intents, GBPS framework forms five user groups respectively. Each user group contains the most top-K users whose static representation is cosine similar to the groupization intent. User representations from every group are then combined by multi-head attention mechanism which decide combination weights based on five groupization intents respectively. By matching the combined information from group users with candidate documents, and involving additional features, GBPS computes the final personalized reranking score.

3.1 Problem Definition

To start with, we formulate the problem with notations. Suppose that there are a set of N individual users, represented as $U = \{u_1, u_2, \dots u_N\}$. Whenever a new query q_t issued by a user u at current time t in M-th session, the user has corresponding search history H_t which includes long-term history and short-term history. Long-term history is represented by past search sessions $\{S_1, S_2, \dots, S_{M-1}\}$, where M is the index of the current session, Short-term history is represented by the current search sessions S_M. Thus, $H_t = \langle\{S_1, S_2, \dots, S_{M-1}\}, S_M\rangle$. Each session is defined as a sequence of queries and a list of corresponding documents for each query. The general denoting formula for m-th session is $S_m = \{\{q_{m,1}, d_{m,1,1}, \dots\}, \dots, \{q_{m,n_m}, d_{m,n_m,1}, \dots\}\}$, where n_m represents the total number of queries in the session S_m. Currently, the user u issues a query q_t, the original non-personalized search engine returns a candidate documents list $D = \{d_1, d_2, \dots\}$, where q_t is short for $q_{M,t}$ and d_i is short for $d_{M,t,i}$. At the moment t when query q_t is issued by user u, a group of users can be formed somehow determined in connection with q_t and H_t, denoted as G_t. We explain the idea and process of dynamically forming a group of users in 3.3 in detail. For each document, we want to compute a relevance score separately based on the issued query q_t, the target user's past search history H_t and the group of users G_t:

$$\begin{aligned}\text{score}(d_i) &= \text{score}(d_i|q_t, H_t, G_t)\\ &= \text{score}(d_i|q_t) + \text{score}(d_i|H_t) + \text{score}(d_i|G_t),\end{aligned} \tag{1}$$

where score($d_i|q_t$) represents the relevance between document d_i and query q_t, score($d_i|H_t$) represents the relevance between document d_i and the target user's search history H_t, score($d_i|G_t$) represents the relevance between document d_i and the group of users G_t. Finally, we rerank the candidate document list D according to the overall score(d_i) to produce a personalized ranking list.

As shown in Figure 1, we design a group-based personalized search framework that exploits a group of users to enhance the personalized search for the target user under the current query. Our proposed personalized search model GBPS mainly includes three parts: a non-group-based sequence deep learning model to construct long-term and short-term user interests, user group formation and combining group information to complete personalized search. Next, we're going to describe every part in GBPS model in details.

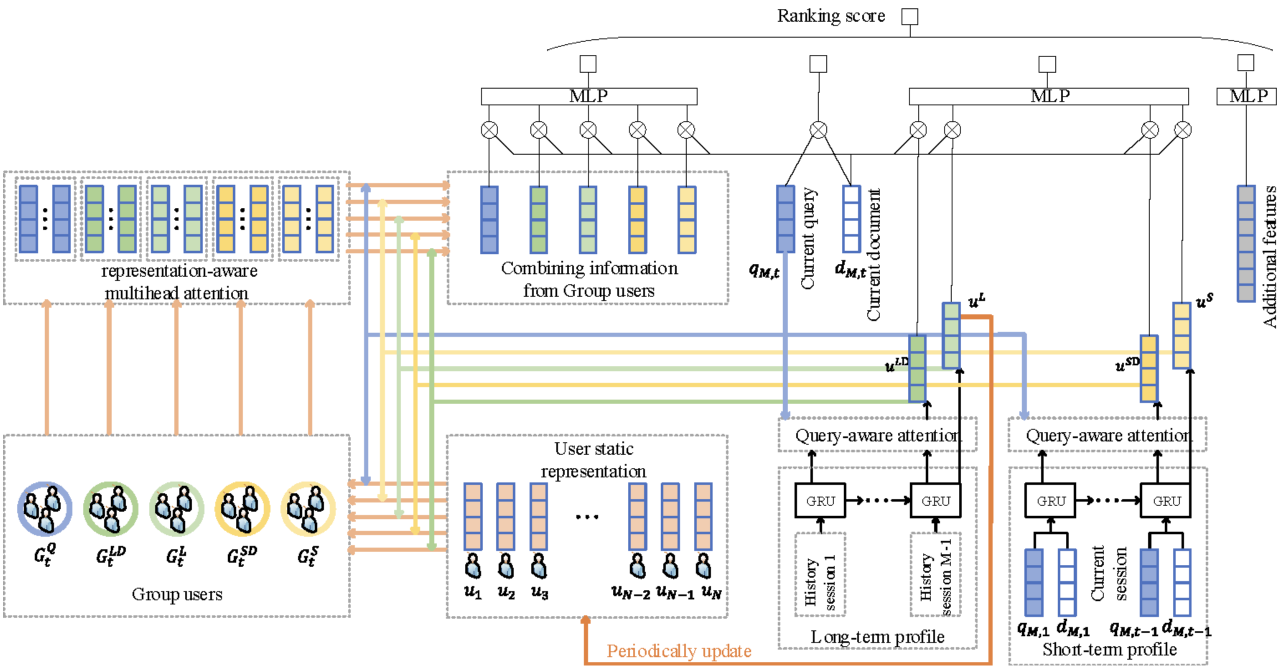

Figure 1: Structure of the GBPS framework. GBPS framework aims to enhance an inner personalized search model by utilizing user group information. Depicted in the bottom-right corner of the figure, it's the Extended-HRNN (modified from HRNN [14]) model functions as a portable inner model. Note that this inner model in GBPS framework can be replaced by other representation-based personalized search models.

3.2 Constructing a Single User Profile

We modify the hierarchical recurrent neural network model (HRNN) [14], called Extended-HRNN. Then we employ Extended-HRNN to learn user profiles from the SAT clicked documents extracted from user search histories.

We choose HRNN as the seed personalized search model to be modified as our inner model, because HRNN has some good qualities. HRNN is a representation-based model designed for the search personalization task, which takes into account the sequential information underlying previous queries and sessions. It mainly consists of two levels RNN to exploit the order of issued queries and past sessions, and automatically generate user profiles from historical data.

To be concrete, firstly HRNN splits a user search history into sessions $\{S_1, S_2, \dots, S_M\}$, where M is the index of current session. Each session not only contains a sequence of queries, but also a list of clicked documents corresponding to each query. Thus, $S_m = \{\{q_{m,1}, d_{m,1,1}, \dots\}, \dots, \{q_{m,n_m}, d_{m,n_m,1}, \dots\}\}$. Representation of a single session is calculated as the last latent state vector of a low-level intra-session RNN, which takes a sequence of query and SAT document pairs as input. In every step of the low-level RNN for the m-th session, latent state vector is calculated according to the inputs and previous state vectors, c:

$$h^1_{m,i} = f^1(h^1_{m,i-1}, q_{m,i}, d_{m,i}). \tag{2}$$

The representation of the current session $h^1_{M,t-1}$ when query q_t issued is considered as user short-term interest vector u^S. Representations of past sessions in search history proceed into a high-level inter-session RNN. The high-level RNN computes the sequence of user representations $\{h^2_1, \dots, h^2_{M-1}\}$ at the end of every session. Formally:

$$h^2_m = f^2(h^2_{m-1}, h^1_m). \tag{3}$$

Later, according to the input query q_t, an attention layer is applied on latent state vector of the high-level RNN to generate the long-term interest vector.

Extended-HRNN extends the calculation of the short-term user interest vector and the long-term user interest vector in HRNN, both by applying and not applying a query-aware attention.

We take in the query-aware long-term user interest vector from HRNN as our **long-term dynamic user interest** u^{LD}, formally:

$$u^{LD} = \mathrm{Att}(q_t, [h^2_1, \dots, h^2_{M-1}], [h^2_1, \dots, h^2_{M-1}]), \tag{4}$$

where $\mathrm{Att}(\cdot)$ is an attention layer following [32], q_t functions as attention query, $[h^2_1, \dots, h^2_{M-1}]$ functions as both attention key and attention value. Obviously, directly adopting the last latent state vector h^2_{M-1} is the simplest way to generate long-term user interest representation. Plus, the last latent state vector is independent to the current query, which makes it possible to reflect on the unbiased long-term user interest. Thus, we extend long-term user interest by using h^2_{M-1} as our **long-term static user interest** u^L.

Likewise, we take in the short-term user interest from HRNN as our **short-term static user interest** u^S, which is equal to $h^1_{M,t-1}$. Besides, the query-aware attention used in the high-level RNN in HRNN model has been proved to highlight different sessions of historical search behaviors dynamically. We believe that applying query-aware attention in the low-level RNN in current session can also assign proper weights to queries in current session, so as to avoid unnecessary noise. Therefore, we apply a query-aware attention layer on latent state vector in current session to calculate our **short-term dynamic user interest** u^{SD}, formally:

$$u^{SD} = \mathrm{Att}(q_t, [h^1_{M,1}, \dots, h^1_{M,t-1}], [h^1_{M,1}, \dots, h^1_{M,t-1}]), \tag{5}$$

where similarly, q_t functions as attention query in this attention layer, $[h^1_{M,1}, \dots, h^1_{M,t-1}]$ functions as both attention key and attention value.

In this way, given a user's search history and her current query, we can construct four vectors to represent this user's interest: long-term static user profile, long-term dynamic user profile, short-term static user profile and short-term dynamic user profile. We use "static" to refer to modeling sequence search history without current query-aware attention, and "dynamic" corresponding to using current query-aware attention ones.

3.3 User Group Formation

In general, we form user groups based on similarity between two representation vectors. For the target user who just issued a new query to be answered, we build her user profile in time as described in 3.2, constructing four vectors to represent her interest in four different aspects.

However, we can not afford to do the profile constructing process for every other user in the same time when the target user issue a query. Clearly, it is foreseeable that the huge amount of calculation and time cost are unbearable. As an alternative, we regard the long-term static user interest as the user's static representation. The long-term static user interest is sequentially constructed and can reflect the user's steady preference. More importantly, it's not biased by any certain issued query, since the construction process of long-term static user profile does not apply query-aware attention. Hence, we save the long-term static user interest vector u^L for every user to hard drive, and update them periodically.

In our GBPS framework, we group users based on the vector similarity of user representations and the current query. For a target user u and an input query q, we construct four types of user profile representations as shown in 3.2, and we have a vector representing the current query. Also, the issued query itself can be viewed as a strong sign of the user's current interest. Due to the characteristic of HRNN, the query representation vector is in the same semantic space as these four user profile representation vectors. This is the guarantee of making sense when we further compute the similarity between query representation and user profile representation. Thus, we have five representation vectors, each revealing a different aspect of the target user's interest.

For another user u_i to be considered, we have her stored static representation, that is her long-term static user interest vector calculated and saved most recently. Let $Sim(x, y)$ indicates a similarity function which computes the similarity between two vectors x and y with same dimensions. Verified by experiment, the cosine similarity function outperforms the L2 similarity function (the L2 similarity is computed as $1/(1 + l2 \quad distance)$).

Following [38], we use the term "groupization" indicating the long process of "applying some methods to all users' search histories then using a group of users' profiles to enhance the relevance and coverage of personalization search methods". We apply the aforementioned cosine similarity function to all users' static profiles trained by their corresponding search histories. Moreover, we use the term "groupization intent" referring to the representation vector of the target user which is calculated cosine similarity with every other user's static profile. The GBPS framework views these five representations revealing the target user's interest as five groupization intents.

With these five groupization intents, the similarity between the target user u and another user u_i can be calculated in five forms: $Sim(u^L, u_i)$, $Sim(u^{LD}, u_i)$, $Sim(u^S, u_i)$, $Sim(u^{SD}, u_i)$ and $Sim(q_t, u_i)$, where q_t refers to the representation of the query q issued at time t. By each form of similarity between the target user u and other users, a subgroup of similar users can be formed. Thus, we find five user subgroups for target user u and current query q defined by the five forms of similarity as above. Each user subgroup is extracted by ranking all other users in descending order of their similarity score in a corresponding way.

For a fixed target user u and a query q newly issued at time t, five subgroups each containing K similar users are formed, formally:

$$\begin{aligned} G_t^S &= \{u_i | \text{rank}(\text{Sim}(u^S, u_i)) \le K\}, \\ G_t^{SD} &= \{u_i | \text{rank}(\text{Sim}(u^{SD}, u_i)) \le K\}, \\ G_t^L &= \{u_i | \text{rank}(\text{Sim}(u^L, u_i)) \le K\}, \\ G_t^{LD} &= \{u_i | \text{rank}(\text{Sim}(u^{LD}, u_i)) \le K\}, \\ G_t^Q &= \{u_i | \text{rank}(\text{Sim}(q_t, u_i)) \le K\}, \end{aligned} \tag{6}$$

where rank(Sim(·)) is the rank of the cosine similarity score Sim(·).

In experiments, there exists some duplicate users from different subgroups some times. Those duplicate users from different subgroups are not deduplicated in the GBPS framework.

3.4 Combining Group Information to Complete Personalized Search

After obtaining of the K-nearest five subgroup of users who share common interests with the user u issued query q, we load their static user profiles saved in hard drive and joint these representation vectors altogether to a matrix. The matrix contains all the group information we extract from all other users with respect to the target user and the current query.

In order to exploit the group information, we apply five multi-head attention layers to fuse the group information. Each multi-head attention layer is corresponding to a different groupization intent, and combines all the group information with weight determined by a groupization intent. Thus, five concentrated representations for group information are computed. Taking the query as groupization intent for example, the concentrated representation corresponding to the current query is computed as follows:

$$g^Q = \text{Att}(q_t, [G_t^S, G_t^{SD}, G_t^L, G_t^{LD}, G_t^Q], [G_t^S, \ldots, G_t^Q]). \tag{7}$$

In this way, we combine group information to five concentrated representation: g^S, g^{SD}, g^L, g^{LD}, g^Q. These five representations are fed into a multi-layer perceptron (MLP) with $tanh(\cdot)$ as the activation function.

In addition to personalization scores calculated by our model, we also incorporate query-document relevance features and click-based features as additional features. Since the original query-document features are inaccessible in our dataset, here we use the original position of the document as a feature. Also, the click features include the total number of historical clicks on the candidate document by the user, the number of clicks on the candidate document under the input query by the user and the click entropy of the input query. The reason we incorporate click entropy is that the value of personalization varies a lot across different queries, and indiscriminately applying personalization on all queries could produce an adverse effect on the overall quality. These additional features are fed into a multi-layer perceptron (MLP) with $tanh(\cdot)$ as the activation function.

Note that our idea is to add group information in personalized search. So finally, we sum up the scores calculated by query-document pair, by user's history, and by group information as the final relevance score.

We choose a basic ranking algorithm, LambdaRank, to train the whole framework. We generate training pairs from query logs by treating the SAT-clicked documents as relevant samples and the others as irrelevant ones. Then we use the representations of document pairs to calculate the loss. Take a pair of relevant document d_i and irrelevant document d_j as an example. The loss function is the product of cross entropy between desired probabilities and predicted probabilities and the change of metrics, Δ, while swapping the positions of the two documents, defined as:

$$\text{loss} = \left(-\bar{p}_{ij}\log\left(p_{ij}\right) - \bar{p}_{ji}\log\left(p_{ji}\right)\right)|\Delta| \tag{8}$$

where p_{ij} represents the predicted probability that d_i is more relevant than d_j , and p_{ij} represents the real probability. Specifically, the predicted probabilities are computed by a logistic function:

$$p_{ij} = \frac{1}{1+\exp\left(-\left(\text{score}\left(d_i\right) - \text{score}\left(d_j\right)\right)\right)}. \tag{9}$$

4 Experimental Setup

In this section, we introduce the dataset used in our experiment, list the baselines compared with our model, list the evaluation metrics used and elaborate our model settings.

4.1 Dataset and Evaluation Metrics

We conduct our experiments on a non-personalized search logs, AOL search log [23]. The statistics of processed dataset are shown in Table 1.

Table 1: Statistics of the datasets.

Type	Train	Valid	Test
User Num	100,110	4,774	5,555
Query Num	598,812	66,879	70,763
Session Num	224,891	27,272	27,767
Avg Session Len	2.66	2.45	2.55
Avg History Len	50.56	64.67	68.18
Avg Click Num	1.11	1.09	1.12

AOL Dataset is a public available query log collected from March 1^{st} 2006 to May 31^{st}. Each record contains an anonymous user ID, a query text, query issued time, a clicked URL and the rank position of the URL in original ranking results. Wasi applied the pseudo labeling technique to construct dataset [2] using AOL search log, since AOL search log only has the clicked documents. In [2], the candidate documents for each query are selected from a fixed size window centered at the position where BM25 [24] ranks the recorded clicked documents. We follow the method in [2] to split user search log into sessions, segmenting with boundaries decided by the similarity between two consecutive queries. To ensure that every user has enough search history to build user profile, we set the first five weeks log as the history. We use the next six weeks log for model training, the remaining two weeks log for modeling validation and testing with the proportion 1:1. Same with [1, 2], five candidate documents are constructed for each query in the training and validation set, while 50 for testing set. We simply use title field in computing for documents.

The **metrics** used in our experiment are mean average precision (MAP), mean reciprocal rank (MRR) and precision@1 (P@1). Applying P@1 rather than Precision at other position is because the average click per query is only 1.11 in AOL dataset. These three are common metrics to measure the ranking quality to evaluate the performance of different models.

4.2 Baselines

We compare our model with adhoc search models, session search models and previous personalized search models. The original ranking and the baseline models are set as follows:

Original. Following [2], we take the ranking retrieved by **BM25** algorithm as a basic baseline.

KNRM [39]. It is a kernel-based neural ranking model for adhoc search. It extracts the features of interaction between query and document terms. The kernel-pooling is used to provide soft match signals for ranking.

Conv-KNRM [11]. It is an upgrade of the KNRM model. It adds a convolutional layer on the basis of KNRM to n-gram soft matches.

P-Click [13]. It is a personalized search model that reranks documents based on the number of clicks made by the same user under the same query in history. It aims to improve search results by satisfying users' widespread refinding behaviors.

SLTB [3]. It extracts 102 features from user search history, including click features, topical features, time features and positional features. It then aggregates these features to personalize the results via the LambdaMart algorithm.

HRNN [14]. It is a personalized search model using a hierarchical recurrent neural network with query-attention to build user profiles. It models the sequential information hidden in query logs and dynamically computes the user profile corresponding to the current query. The additional features in HRNN model are identical to SLTB features.

PSGAN [18]. It is a personalized framework that uses a hierarchical RNN and applies GAN as well. It generates queries that match the user's query intent better, and selects document pairs more valuable for learning user interests. In this paper, we take the variant PSGAN-D as our baseline, considering the cost of training.

PSTIE [21]. This model considers using time interval information in history to enhance the performance of the personalized search.

In this work, our GBPS model has not compared with three personalized search models: KEPS [19], HTPS [46], PEPS [40]. Here's our explanation. GBPS doesn't use external knowledge information, while KEPS conducts entity linking and uses external knowledge to enhance user profiles. GBPS builds user profiles from the sentence level, while HTPS models user profiles from the term level in finer granularity. GBPS is a representation-based matching model where queries, documents and users are represented in vector forms before matching with each other. However, PEPS matches representation vectors to compute the document relevance score with the interaction-based neural component KNRM. It appears that interaction-based models generally perform better than representation-based models [22, 45], thus we ignore the comparison between GBPS and PEPS, either.

4.3 Model Settings

To ensure the validity of user profile modeling, we filter out the users whose search log contain less than four sessions, thus about 26k valid users' search log is actually used. For every intent we use as a seed to format group, we find a set of the top 10 most similar users. Similar to [25], we initialize the word embedding matrix W with a pre-trained unsupervised model and keep it fixed during the training. As for the multi-head attention mechanism, we use 10 heads, and the dimension of each head is 50. Model optimization uses the Adam optimizer, with a learning rate of $1e-3$ and $\epsilon = 1e-5$. We use pairwise training, and the training loss is defined as cross-entropy. Besides, we adopt an early stop strategy to end the training when the average loss on the validation set stops decreasing in 3 continuous epochs.

5 Results and Analysis

5.1 Overall Performance

To begin with, we compare the overall performances of all baselines and GBPS. We train all models on the training set and then evaluate them on the testing set without any update. The overall results are reported in Table 3.

Table 2: Overall performance of all models on AOL dataset. "†" indicates significant improvements over all baselines with paired t-test at p < 0.01 level. The best results are shown in bold.

Model	MAP	MRR	P@1
Adhoc search model			
Original(BM25)	.2504	.2596	.1534
KNRM	.4291	.4391	.2704
Conv-KNRM	.4738	.4849	.3266
User profile based personalized search model			
P-Click	.4224	.4298	.3788
SLTB	.5072	.5194	.4657
HRNN	.5423	.5545	.4854
PSGAN	.5480	.5601	.4892
Group-based personalized search model			
GBPS	**.5805**†	**.5927**†	**.5114**†

Generally, all personalized search models improve the original ranking results greatly, indicating that personalization is helpful for promoting users' search experience. The increase of P-Click model validates the effectiveness of refinding behaviors. SLTB model realizes personalization by extracting various interest-related features from the search history. HRNN and PSGAN which all build user interest profiles with a hierarchical RNN achieve great results.

Compared with all the baselines list in Table 3, our GBPS model achieves best performance in terms of all the evaluation metrics, with t-test at $p < 0.01$ level.

5.2 Effectiveness of GBPS

Here, we contrast the effectiveness between HRNN model, our adjusted Extended-HRNN model and our proposed framework GBPS with Extended-HRNN as inner model.

Table 3: Overall performance of all models on AOL dataset. "†" indicates significant improvements over all baselines with paired t-test at p < 0.01 level. The best results are shown in bold.

Model	MAP	MRR	P@1
HRNN	.5423	.5545	.4854
Extended-HRNN	.5605	.5730	.5002
GBPS	**.5805**†	**.5927**†	**.5114**†

Specially compared with the closest baseline Extended-HRNN, our GBPS model still improves by 1.65% in MAP, 1.65% in MRR and 1.47% in P@1 metric. Since Extended-HRNN is the non-group form of our model, the result shows that the GBPS model do enhance user search history data and extract effective information from user group similar to the target user and the current query.

This result shows that GBPS framework inherit the capability of user interest profiles modeling from the Extended-HRNN, proves the effectiveness of GBPS framework. So, for a inner representation-based personalized search model, GBPS framework can enhance the personalized search result.

5.3 Ablation Analysis

The GBPS model includes five groupization approaches, forming user group based on five different grouping intents: long-term static user profile, long-term dynamic user profile, short-term static user profile, short-term dynamic user profile and the current query.

Note that the two dynamic user profiles are calculated by the current query-aware attention, and the current query is used already and alone. Thus, we focus more on the commonness between long-term static user profile and long-term dynamic user profile, and the commonness between short-term static user profile and short-term dynamic user profile. We regard two long-term user profiles as long-term type, because long-term user profiles represent user's stable preference, both static and dynamic. Similarly, we regard two short-term user profiles as short-term type, because short-term user profiles represent user's session interest, both static and dynamic. In this way, we classify these five grouping intents into three types: long-term type, short-term type and query type.

To figure out the role of each type of grouping intent for personalization, we perform several ablation experiments. We illustrate the experimental results in Table 4 and have some discussions.

Table 4: Results of ablation experiments.

GBPS Variant	MAP	MRR	P@1
GBPS	.5805	.5927	.5114
w/o Long-term	.5751	.5871	.5078
w/o Short-term	.5701	.5818	.5020
w/o Query	.5715	.5834	.5055

5.3.1 Long-term type groupization approach

In order to conform the respective effects of the long-term type groupization approaches, we strip off the groupization based on similarity compared with two long-term user profiles to experiment. In this case, we only use short-term user profiles and the current query to find user groups. As shown in Table 4, GBPS without long-term type groupization performs the best in all GBPS variants, which means long-term type groupization approach makes the least use of information from other users. The MAP, MRR, P@1 metrics drop 0.54%, 0.56%, 0.36% on AOL dataset. We analyze it most likely because a user's long-term type representations aim to model the user's long-term preference and does not contain the user's current information need, while the user's current information need is crucial for search task.

However, GBPS without long-term type groupization still outperforms all non-group-based baseline models in Table 4. This shows that long-term type groupization approach do extract useful information from other users, though long-term type groupization approach is not as effective as short-term type groupization and query type groupization approaches.

5.3.2 Short-term type groupization approach

Similarly, we abandon the groupization based on similarity compared with two short-term user profiles to experiment in order to study the effectiveness of short-term type groupization approach. We find the GBPS framework loses 1.04% in MAP, 1.09% in MRR and 0.94% in P@1 without the short-term type groupization approach on the AOL dataset. As shown in Table 4, GBPS without short-term type groupization performs the worst in all GBPS variants, which means the short-term

Table 5: Results of ablation experiments.

GBPS Variant	MAP	MRR	P@1
GBPS	.5805	.5927	.5114
GBPS-L	.5515	.5624	.4815
GBPS-LD	.5706	.5826	.5006
GBPS-S	.5473	.5584	.4778
GBPS-SD	.5641	.5756	.4951

type groupization approach makes the most use of information from other users compared with the other two approaches. We argue that this indicates that short-term user profiles are the best to represent the user's search intent, compared with long-term profiles and the current issued query. A user's short-term profiles aim to model the user's interest in the current session. In our framework, the short-term static user profile and the short-term dynamic user profile are all sequentially built using the search behaviors in the current session, thus gaining the ability to reflect the user's search intent well.

5.3.3 Query type groupization approach

We drop the groupization based on similarity compared with the current query in GBPS framework to analyze the contribution of query type groupization approach. In this case, GBPS degenerates to a query-independent grouping formation model. Without group formation by the query type approach, the MAP, MRR, Precision metrics drop to the second in all GBPS variants. Compared with the long-term type groupization approach, the query type groupization is more effective and can extract user groups more helpful for satisfying the target user's search intent. In the meantime, the query type groupization is less effective than the short-term type groupization approach. We analyze it most likely because some queries are ambiguous. User groupization by ambiguous query may lead to finding users sharing similar queries but not sharing similar interests. Thus, the query type groupization approach may drive down the performance slightly compared with the short-term type groupization approach. Vu et al. once argued that the grouping method needs to be dynamic and dependent on the input query (i.e., different input queries should return different groups of similar users) since fixed grouping methods may neglect the fact that users in a group may have different interests with respect to different topics [35]. We agree with Vu et al. on this point since their work constructed user profiles by employing Latent Dirichlet Allocation (LDA), while we still want to refine this statement when deep learning is widely used in information retrieval. We argue that it is not the query that matters in similar users groupization, but the groupization intent that can best represent the target user's search intent. Groupization intent needs to imitate the user's search intent. We presume that when a user's search intent can be well represented in an exploitable way, groupization can be promising to improve personalized search results.

5.4 Effect of Different User Profiles

To further explore the effect of different user profiles when using as groupization approach, we conduct several experiments as listed in Table 5. Different from the ablation experiments in 5.3, which drop different groupization parts from the overall GBPS model, here we choose to conduct each experiment in which only one groupization approach is applied. Take the 2nd and 3rd row in Table 5 for example, "GBPS-L" means only top-K similar users similar to the target user's long-term static profile are exploited, and "GBPS-LD" means user group is formed only similar with the target user's long-term dynamic profile.

Comparing GBPS-L and GBPS-LD, we find GBPS-LD outperforms GBPS-L. This indicates that the dynamic long-term user profile is better than the static long-term user profile. We get similar results when using the short-term profile: the dynamic version GBPS-SD outperforms GBPS-S. These results confirm that the query aware attention is able to improve the quality of user profiles. The dynamic user profiles embed the important user intent information contained in the current query. Using dynamic user profiles is able to find better similar users that have similar intent before.

Comparing the long-term profiles and short-term profiles, the long-term profile based methods GBPS-L and GBPS-LD outperform the short-term based ones GBPS-S and GBPS-SD. This indicates that using long-term user profiles are more effective than using short-term profiles.

6 Conclusion And Future Work

In this work, we propose a group-based personalized search framework GBPS which makes use of multiple dynamic groupization approaches to enrich a user profile for a bulit-in personalized search model. Firstly, we apply a non-group-based deep learning personalized search model to construct four types of user profiles. To further supplement the user's profile, several groups of users are formed by different groupization intents that include user profiles and the current query. The representations of these groups of users are combined through multi-head attention mechanism, and then match with candidate documents. After adding information from dynamically formed user groups in, the personalized search framework GBPS is built. Experimental results prove the effectiveness of our GBPS framework, confirming the improvement of GBPS framework on the basis of the built-in representation-based personalized search model.

In the future, it's better to explore different built-in non-group-based search models as the portable inner model in GBPS framework, to prove the effectiveness of GBPS more sufficiently. Besides, experiments on an inner model more advanced than HRNN are likely to have better results. In addition, it's promising to extend the GBPS framework to construct group users' profiles in a more refined and more timely way instead of using periodically saved long-term user profiles.

Funding

This research was funded by the China Postdoctoral Fellowship Program of CPSF (GZC20230287) and the Fundamental Research Funds for the Central Universities (2024QY004).

Author contributions

Conceptualization, W.S. and Y.Z.; methodology, W.S.; software, W.S.; validation, Y.Z. and Z.H.; formal analysis, W.S. and Y.Z.; investigation, W.S.; resources, Y.Z., Z.H. and Y.B; data curation, W.S.; writing—original draft preparation, W.S.; writing—review and editing, Y.Z.; visualization, W.S.; supervision, Y.Z. and Z.H.; project administration, Y.Z.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Ahmad, W.U.; Chang, K.; Wang, H. (2018). Multi-Task Learning for Document Ranking and Query Suggestion, *Proceedings of 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, OpenReview.net, 1, 2018.

[2] Ahmad, W.U.; Chang, K.; Wang, H. (2019). Context Attentive Document Ranking and Query Suggestion, *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, ACM, 385–394, 2019.

[3] Bennett, P.N.; White, R.W.; Chu, W.; Dumais, S.T.; Bailey, P.; Borisyuk, F.; Cui, X. (2012). Modeling the impact of short- and long-term behavior on search personalization, *Proceedings*

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

[4] Bennett, P.N.; Svore, K.M.; Dumais, S.T. (2010). Classification-enhanced ranking, *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*, ACM, 111–120, 2010.

[5] Blei, D.M.; Ng, A.Y.; Jordan, M.I. (2003). Latent Dirichlet Allocation, *J. Mach. Learn. Res.*, ACM / IW3C2, 3, 993–1022, 2003.

[6] Breese, J.S.; Heckerman, D.; Kadie, C.M. (1998). Empirical Analysis of Predictive Algorithms for Collaborative Filtering, *Proceedings of UAI '98: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, University of Wisconsin Business School, Madison, Wisconsin, USA, July 24-26, 1998*, Morgan Kaufmann, 43–52, 1998.

[7] Burges, C.J.C.; Shaked, T.; Renshaw, E.; Lazier, A.; Deeds, M.; Hamilton, N.; Hullender, G.N. (2005). Learning to rank using gradient descent, *Proceedings of Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, ACM, 89–96, 2005.

[8] Cai, F.; Liang, S.; Rijke, M.d. (2014). Personalized document re-ranking based on Bayesian probabilistic matrix factorization, *Proceedings of The 37th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '14, Gold Coast , QLD, Australia - July 06 - 11, 2014*, ACM, 835–838, 2014.

[9] Carman, M.J.; Crestani, F.; Harvey, M.; Baillie, M. (2010). Towards query log based personalization using topic models, *Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010*, ACM, 1849–1852, 2010.

[10] Collins-Thompson, K.; Bennett, P.N.; White, R.W.; Chica, S.d.l.; Sontag, D.A. (2011). Personalizing web search results by reading level, *Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM 2011, Glasgow, United Kingdom, October 24-28, 2011*, ACM, 403–412, 2011.

[11] Dai, Z.; Xiong, C.; Callan, J.; Liu, Z. (2018). Convolutional Neural Networks for Soft-Matching N-Grams in Ad-hoc Search, *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, ACM, 126–134, 2018.

[12] Deng, C.; Zhou, Y.; Dou, Z. (2022). Improving Personalized Search with Dual-Feedback Network, *Proceedings of WSDM '22: The Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event / Tempe, AZ, USA, February 21 - 25, 2022*, ACM, 210–218, 2022.

[13] Dou, Z.; Song, R.; Wen, J. (2007). A large-scale evaluation and analysis of personalized search strategies, *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, ACM, 581–590, 2007.

[14] Ge, S.; Dou, Z.; Jiang, Z.; Nie, J.; Wen, J. (2018). Personalizing Search Results Using Hierarchical RNN with Query-aware Attention, *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, ACM, 347–356, 2018.

[15] Harvey, M.; Crestani, F.; Carman, M.J. (2013). Building user profiles from topic models for personalised search, *Proceedings of 22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, ACM, 2309–2314, 2013.

[16] Li, X.; Guo, C.; Chu, W.; Wang, Y.; Shavlik, J. (2014). Deep learning powered in-session contextual ranking using clickthrough data, *Proceedings of In Proc. of NIPS*, 605–616, 2014.

[17] Liu, F.; Yu, C.T.; Meng, W. (2002). Personalized web search by mapping user queries to categories, *Proceedings of the 2002 ACM CIKM International Conference on Information and Knowledge Management, McLean, VA, USA, November 4-9, 2002*, ACM, 558–565, 2002.

[18] Lu, S.; Dou, Z.; Jun, X.; Nie, J.; Wen, J. (2019). PSGAN: A Minimax Game for Personalized Search with Limited and Noisy Click Data, *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, ACM, 555–564, 2019.

[19] Lu, S.; Dou, Z.; Xiong, C.; Wang, X.; Wen, J. (2020). Knowledge Enhanced Personalized Search, *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, ACM, 709–718, 2020.

[20] Matthijs, N.; Radlinski, F. (2011). Personalizing web search using long term browsing history, *Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM 2011, Hong Kong, China, February 9-12, 2011*, ACM, 25–34, 2011.

[21] Ma, Z.; Dou, Z.; Bian, G.; Wen, J. (2020). PSTIE: Time Information Enhanced Personalized Search, *Proceedings of CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, ACM, 1075–1084, 2020.

[22] Mitra, B.; Craswell, N. (2018). An Introduction to Neural Information Retrieval, *Found. Trends Inf. Retr.*, ACM, 13(1), 1–126, 2018.

[23] Pass, G.; Chowdhury, A.; Torgeson, C. (2006). A picture of search, *Proceedings of the 1st International Conference on Scalable Information Systems, Infoscale 2006, Hong Kong, May 30-June 1, 2006*, ACM, 1, 2006.

[24] Robertson, S.E.; Zaragoza, H. (2009). The Probabilistic Relevance Framework: BM25 and Beyond, *Found. Trends Inf. Retr.*, ACM, 3(4), 333–389, 2009.

[25] Severyn, A.; Aless, ; Moschitti, r. (2015). Learning to Rank Short Text Pairs with Convolutional Deep Neural Networks, *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, ACM, 373–382, 2015.

[26] Sieg, A.; Mobasher, B.; Burke, R.D. (2007). Web search personalization with ontological user profiles, *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, CIKM 2007, Lisbon, Portugal, November 6-10, 2007*, ACM, 525–534, 2007.

[27] Song, Y.; Wang, H.; He, X. (2014). Adapting deep RankNet for personalized search, *Proceedings of Seventh ACM International Conference on Web Search and Data Mining, WSDM 2014, New York, NY, USA, February 24-28, 2014*, ACM, 83–92, 2014.

[28] Sun, J.; Zeng, H.; Liu, H.; Lu, Y.; Chen, Z. (2005). CubeSVD: a novel approach to personalized Web search, *Proceedings of the 14th international conference on World Wide Web, WWW 2005, Chiba, Japan, May 10-14, 2005*, ACM, 382–390, 2005.

[29] Teevan, J.; Dumais, S.T.; Liebling, D.J. (2008). To personalize or not to personalize: modeling queries with variation in user intent, *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008*, ACM, 163–170, 2008.

[30] Teevan, J.; Liebling, D.J.; Ravich, G.; Geetha, r. (2011). Understanding and predicting personal navigation, *Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM 2011, Hong Kong, China, February 9-12, 2011*, ACM, 85–94, 2011.

[31] Teevan, J.; Morris, M.R.; Bush, S. (2009). Discovering and using groups to improve personalized search, *Proceedings of the Second International Conference on Web Search and Web Data Mining, WSDM 2009, Barcelona, Spain, February 9-11, 2009*, ACM, 15–24, 2009.

[32] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. (2017). Attention is All you Need, *Proceedings of Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 5998–6008, 2017.

[33] Vu, T.; Nguyen, D.Q.; Johnson, M.; Song, D.; Willis, A. (2017). Search Personalization with Embeddings, *Proceedings of Advances in Information Retrieval - 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, April 8-13, 2017, Proceedings*, 598–604, 2017.

[34] Vu, T.T.; Willis, A.; Tran, S.N.; Song, D. (2015). Temporal Latent Topic User Profiles for Search Personalisation, *Proceedings of Advances in Information Retrieval - 37th European Conference on IR Research, Vienna, Austria, March 29 - April 2, 2015. Proceedings*, 605–616, 2015.

[35] Vu, T.T.; Song, D.; Willis, A.; Tran, S.N.; Li, J. (2014). Improving search personalisation with dynamic group formation, *Proceedings of The 37th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '14, Gold Coast , QLD, Australia - July 06 - 11, 2014*, ACM, 951–954, 2014.

[36] Wang, H.; He, X.; Chang, M.; Song, Y.; White, R.W.; Chu, W. (2013). Personalized ranking model adaptation for web search, *Proceedings of The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13, Dublin, Ireland - July 28 - August 01, 2013*, ACM, 323–332, 2013.

[37] Wang, S.; Zhang, H.; Yuan, Z. (2024). Enhancing Sequence Representation for Personalized Search, *Proceedings of CCL*, Lecture Notes in Computer Science, Springer, 14232, 434–448, 2024.

[38] White, R.W.; Chu, W.; Awadallah, A.H.; He, X.; Song, Y.; Wang, H. (2013). Enhancing personalized search by mining and modeling task behavior, *Proceedings of 22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, International World Wide Web Conferences Steering Committee / ACM, 1411–1420, 2013.

[39] Xiong, C.; Dai, Z.; Callan, J.; Liu, Z.; Power, R. (2017). End-to-End Neural Ad-hoc Ranking with Kernel Pooling, *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, ACM, 55–64, 2017.

[40] Yao, J.; Dou, Z.; Wen, J. (2020). Employing Personal Word Embeddings for Personalized Search, *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, Virtual Event, China, July 25-30, 2020*, ACM, 1359–1368, 2020.

[41] Yao, J.; Dou, Z.; Xu, J.; Wen, J. (2021). RLPS: A Reinforcement Learning-Based Framework for Personalized Search, *ACM Transactions on Information Systems*, 39(3), 27:1–27:29, 2021.

[42] Yuan, Z. (2023) Self-supervised End-to-end Graph Local Clustering, *World Wide Web*, Springer 26(3), 1157–1179, 2023.

[43] Yuan, Z.; Wen, J.; Wei, Z.; Liu, J.; Yao, B.; Zheng, K. (2019). Real-time Interactive Analysis on Big Data, *Journal of Software*, 13(1), 162–182, 2020.

[44] Yuan, Z.; Wei, Z.; Lv, F.; Wen, J. (2024). Index-free Triangle-based Graph Local Clustering, *Frontiers of Computer Science*, Springer 18(3), 183404, 2024.

[45] Zhang, Y.; Rahman, M.M.; Braylan, A.; Br, ; Dang, o.; Chang, H.; Kim, H.; McNamara, Q.; Angert, A.; Banner, E.; Khetan, V.; McDonnell, T.; Nguyen, A.T.; Xu, D.; Wallace, B.C.; Lease, M. (2016). Neural Information Retrieval: A Literature Review, *CoRR*, ACM, abs/1611.06792, 373–382, 2016.

[46] Zhou, Y.; Dou, Z.; Wen, J. (2020). Encoding History with Context-aware Representation Learning for Personalized Search, *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, ACM, 1111–1120, 2020.

[47] Zhou, Y.; Dou, Z.; Wen, J. (2020). Enhancing Re-finding Behavior with External Memories for Personalized Search, *Proceedings of WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, ACM, 789–797, 2020.

[48] Zhou, Y.; Dou, Z.; Wen, J. (2023). Enhancing Potential Re-Finding in Personalized Search With Hierarchical Memory Networks, *IEEE Transactions on Knowledge and Data Engineering*, 35(4), 3846–3857, 2023.

Cite this paper as:

Wang, S.; Yuan, Z.; Zhang, H.; Yang, B. (2025). Group Users Enhanced Representation-based Neural Personalized Search, *International Journal of Computers Communications & Control*, 20(4), 6698, 2025.

https://doi.org/10.15837/ijccc.2025.4.6698