Cross-Domain Product Search with Knowledge Graph

Rui Zhu
ruizhu@whu.edu.cn
School of Cyber Science and Engineering, Wuhan University
China

Yiming Zhao
desert.zym@antgroup.com
Ant Group
China

Wei Qu
qingze.qw@antgroup.com
Ant Group
China

Zhongyi Liu
zhongyi.lzy@antgroup.com
Ant Group
China

Chenliang Li[†]
cllee@whu.edu.cn
Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University
China

ABSTRACT

The notion *personalization* lies on the core of a real-world product search system, whose aim is to understand the user's search intent in a fine-grained level. The existing solutions mainly achieve this purpose through a coarse-grained semantic matching in terms of the query and item's description or the collective click correlations. Besides the issued query, the historical search behaviors of a user would cover lots of her personalized interests, which is a promising avenue to alleviate the semantic gap between users, items and queries. However, as to a specific domain, a user's search behaviors are generally sparse or even unavailable (*i.e.,* cold-start users). How to exploit the search behaviors from the other relevant domain and enable effective fine-grained intent understanding remains largely unexplored for product search. Moreover, the semantic gap could be further aggravated since the properties of an item could evolve over time (*e.g.,* the price adjustment for a mobile phone or the business plan update for a financial item), which is also mainly overlooked by the existing solutions.

To this end, we are interested in bridging the semantic gap via a marriage between cross-domain transfer learning and knowledge graph. Specifically, we propose a simple yet effective knowledge graph based information propagation framework for cross-domain product search (named KIPS). In KIPS, we firstly utilize a shared knowledge graph relevant to both source and target domains as a semantic backbone to facilitate the information propagation across domains. Then, we build individual collaborative knowledge graphs to model both long-term interests/characteristics and short-term interests/characteristics of a user/item respectively. In order to harness cross-domain interest correlations, two unsupervised strategies to guide the interest learning and alignment are introduced: *maximum mean discrepancy* (MMD) and *kg-aware contrastive learning*. In detail, the MMD is utilized to support a coarse-grained domain alignment over the user's long-term interests across two domains. Then, the kg-aware contrastive learning process conducts a fine-grained interest alignment based on the shared knowledge graph. Experiments over two real-world large-scale datasets demonstrate the effectiveness of KIPS over a series of strong baselines. Our online A/B test also shows substantial performance gain on multiple metrics. Currently, KIPS has been deployed in AliPay for financial product search. Both the code implementation and the two datasets used for evaluation will be released online publicly[1].

[†]Chenliang Li is the corresponding author. Work done when Rui Zhu was an intern at Ant Group.

CIKM '22, October 17–21, 2022, Atlanta, GA, USA

ACM ISBN 978-1-4503-9236-5/22/10…$15.00
https://doi.org/10.1145/3511808.3557116

CCS CONCEPTS

• **Information systems → Retrieval models and ranking**.

KEYWORDS

Cross-Domain Search, Product Search, Search and Ranking

ACM Reference Format:
Rui Zhu, Yiming Zhao, Wei Qu, Zhongyi Liu, Chenliang Li. 2022. Cross-Domain Product Search with Knowledge Graph. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM '22), October 17–21, 2022, Atlanta, GA, USA*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3511808.3557116

1 INTRODUCTION

Product search has become an indispensable part of many online businesses. The purpose is to extract the desired product from a tremendous amount of (homogeneous) compatible candidates. Compared with traditional web search that identifies the relevant information regarding the user's information need, we need to precisely approximate the personalized user interest for product search. In other words, performing only the semantic matching based on the user query and item's textual description is not effective to distinguish the fine-grained intent for better user satisfaction.

[1]THe code and datasets are available athttps://github.com/WHUIR/KIPS

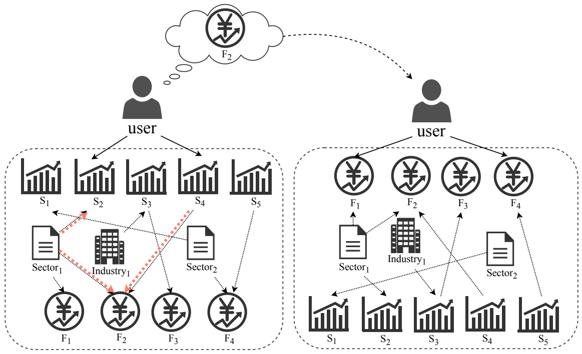


Figure 1: The search intents of the same user across two domains share some correlations. The clicked items in the source domain (i.e., S_2 and S_4) are relevant to the item F_2. Although F_2 does not exist in the user's interactions in the target domain, we can infer that the user is interested in F_2 based on her interactions in the source domain.

Inspired by the recent advance of recommendation systems [9, 11, 14, 21, 33], the historical search behaviors of a user can be considered as a reflection of her personalized interests. Given the sufficient user search behaviors for the relevant items (*e.g.,* in the same domain), it is promising to alleviate the semantic gap between the users and the items by leveraging the interacted items. However, we still face two critical challenges for real-world scenarios. Firstly, the search behaviors of a user could be very sparse and imbalanced. As to a specific domain, we may not have any historical search behavior for a user. In this case, it is hard for us to refine the user's search intent correctly. Secondly, the properties of an item could evolve over time. For example, in our financial product search system, the price of an item could change constantly and the business profile of this item could also be updated.

Actually, the items clicked previously by the same user on different domains may have some correlations. We can infer the user's fine-grained interest on the target domain by transferring her interest from some relevant domain (*i.e.,* source domain). In the past cross-domain methods, the correlated features are usually transferred based on the overlapping users[35]. Specifically, the existing works usually learn the representations of overlapping users on two domains respectively, and then exploit representation correlation across domains. These works can be summarized into two categories. The first ones are to directly fuse the user embedding learnt from the source domain into the target domain [10, 33, 34]. The other category is to use a distance constraint loss to minimise the distance between the two user embeddings [3, 15]. However, both the two strategies aggregate information across domains in a coarse-grained manner, which probably introduce unnecessary noise. Besides, these methods ignore the relatedness of the items in different domains, or only establish indirect relationships through co-click patterns. Furthermore, it is unknown how to accommodate the temporal dynamics of the item properties, which could be a critical factor in many product search systems.

To this end, in this paper, we propose to bridge the semantic gap via a marriage between cross-domain transfer learning and knowledge graph for personalized product search. As shown in Figure 1, the same user could search financial items in two relevant domains. We can utilize a shared knowledge graph relevant to both domains to explicitly establish the underlying interest correlation. It is clear to see that the clicked items (*i.e.,* S_2 and S_4) in the source domain are relevant to financial item F_2 in the target domain. Therefore, we can refine the search results following these semantic clues for better search efficacy.

Many existing knowledge graph based models learn entity and relation representations by knowledge graph embedding. This resultant structure information is validated to be beneficial for many downstream tasks. Some methods such as KGAT [27], proposes to transfer structure proximity in terms of graph convolutional network for item recommendation. They inject the user-item interactions into the underlying item knowledge graph as a *collaborative knowledge graph.* Following this idea, we devise a novel knowledge graph based information propagation framework for cross-domain product search (named KIPS). As aforementioned, a shared knowledge graph covering both the source domain and target domain is utilized to facilitate the information propagation. To further accommodate with the dynamic nature of both user interests and item properties, we first divide the interaction history of both users and items into two parts: long-term and short-term. From a long-term perspective, we consider that the user interactions occurring on the relevant domains are correlated and similar. We adopt a maximum mean discrepancy (MMD) to align the interest embeddings across domains.

On the other hand, the short-term behaviors of the user on source domain may contain more fine-grained interest information that is not reflected on target domain. Hence, we choose to combine the short-term representation of user on source domain into the user embedding of target domain, achieving the information transfer from source domain to target domain. Additionally, we further introduce a kg-aware contrastive learning to enhance the fine-grained interest alignment based on the shared knowledge graph. In detail, for each ground-truth item in either domain, we sample positive and negative items from the opposite domain based on their relatedness inside the knowledge graph, and training model to successfully distinguish the positive and negatives items. In this way, we can not only enhance the bidirectional transferring between source domain and target domain, but also enrich the users' interest information of the target domain.

Our contribution can be summarized as follows:

- To the best of our knowledge, there is no existing work that performs product search through a cross-domain learning paradigm by knowledge graph. Our work can be considered as the first attempt by exploiting the historical search behaviors from the other relevant domain to enhance the search intent understanding in a fine-grained manner.
- The proposed KIPS performs interest alignment across domains by explicitly modeling the long-term and short-term interactions between users and items, which can capture the dynamics of item properties and user interests. Moreover, we introduce a kg-aware contrastive learning to enhance the fine-grained interest alignment based on the shared knowledge graph.

- We perform extensive offline experiments on two pairs of real-world datasets. Both offline and online experiments demonstrate the superiority of our proposed model over the state-of-art methods.

2 RELATED WORK

Since our work is highly related to product search, transfer learning and cross-domain learning, as well as graph neural networks, therefore we briefly summarize the existing advances on these areas.

2.1 Product Search

The object of product search is to estimate the click or purchase likelihood of an item in terms of the query issued by the user and her personalized information. Recently, this task has drawn increasing attention both in academic and industrial. Earlier work learns query representation from structured product entities [4]. DRL [12] proposes to obtain an optimal ranking policy based on reinforcement learning. GEPS [31] incorporates graph embedding techniques into neural retrieval models. They construct a query-item click graph and enrich the semantic information for both queries and items, which is validated to be effective in alleviating the semantic gap to some extent.

Note that for product search, a general query could match well towards thousands of homogeneous yet comparable items. Therefore, we need to incorporate user preference to refine the search results. Specifically, some works focus on the personalized product search. HEM [2] proposes a hierarchical embedding model to jointly learn semantic representations of different entities for personalized product search. Guo *et al.* [7] combine visual preference of users into a multi-modal personalized product search method. Then they further propose to combine the long and short term user preferences with the current query to capture users' search intentions [6]. Ai *et al.* [1] introduce a zero attention model for product search which can automatically determine to personalize user-query pairs. DBML [28] proposes a dynamic learning model to learn latent representations through a probabilistic metric learning framework which can capture uncertainty of entities. GraphSRRL [16] models the structural relationship into graph representation learning to effectively learn a better personalized product search.

2.2 Transfer learning and Cross-domain learning

Under the paradigm of deep learning, fine-tuning strategy is a simple but widely used solution for transfer learning. It first initializes the model parameters for the target domain by adopting the parameters of the well-trained models on the source domain, Then fine-tuning is performed based on labeled data on target domain [18, 30]. A more effective method is to share the hidden feature representations and model parameters between the source tasks and target tasks [29]. Note that there is only few works that perform transfer learning for product search task. The relevant efforts are mainly devoted to enhance recommendation performance. Specifically, CMF [22] transfers knowledge by factorizing the joint rating matrix across domains. They utilize the shared user factors to enable information transfer. Similarly, CoNet [10] achieves dual knowledge transfer between source domain and target domain. CATN [33] extracts multiple aspects from reviews and learns the aspect correlations across domains. WE-CAN [20] proposes a new Wasserstein regularizers which can push the shared features closer and pull domain-specific features more apart. Recent years, more efforts choose to learn the item and user representations by propagating information across domains based on graphs [26, 32, 35]. CD-GNN [17] introduces an additional domain classifier loss to promote the information transfer between domains.

2.3 GNN-based Model

Graph Neural Networks (GNN) can capture inherently high-order dependencies between nodes. Hence, many variants of GNN, such as GCN [13, 32], GraphSage [8], GAT [25] have been widely applied in various scenarios. KGAT [27] employs GAT into the knowledge graph. It proposes to construct a collaborative knowledge graph and apply attentive embedding propagation on this collaborative knowledge graph. KCAN [24] refines the knowledge graph to obtain target-specific node representation with a conditional attention aggregation. To summarize, no existing work aims at performing fine-grained intent understanding in the paradigm of cross-domain learning for product search. Another novelty of our proposed KIPS is to facilitate the information propagation by utilizing a shared knowledge graph as the underlying semantic backbone.

3 METHODS

We start with the preliminaries including the notation description and the construction of collaborative knowledge graph. Then, each component of the proposed KIPS is described in detail, followed by the model optimization process.

3.1 Preliminary

Given there are two domains, let $\mathcal{S} =< \mathcal{U}, Q^{\mathcal{S}}, \mathcal{I}^{\mathcal{S}} >$ and $\mathcal{T} =< \mathcal{U}, Q^{\mathcal{T}}, \mathcal{I}^{\mathcal{T}} >$ denote the source and target domains respectively. Here, $\mathcal{U}$ denotes the set of m overlapped users in both domains, $Q^{\mathcal{S}}$ and $Q^{\mathcal{T}}$ denote the set of queries in the source domain and target domain respectively, $\mathcal{I}^{\mathcal{S}}$ and $\mathcal{I}^{\mathcal{T}}$ denote the set of items in $\mathcal{S}, \mathcal{T}$ respectively.

For a given user u in $\mathcal{U}$, we can organize her clicked items during the previous search activities in chronological order as $\mathcal{N}^{\mathcal{S}} = \{i_1^{\mathcal{S}}, i_2^{\mathcal{S}}, \cdots, i_m^{\mathcal{S}}\}$, $\mathcal{N}^{\mathcal{T}} = \{i_1^{\mathcal{T}}, i_2^{\mathcal{T}}, ..., i_n^{\mathcal{T}}\}$ for the two domains, where $i_j^{\mathcal{S}}$ and $i_j^{\mathcal{T}}$ denote the j-th item clicked by the user in the source and target domains respectively. Based on the historical user-item interactions in $\mathcal{N}^{\mathcal{S}}$ and $\mathcal{N}^{\mathcal{T}}$ for all users in $\mathcal{U}$, we can form the corresponding user-item bipartite graph $\mathcal{G}_b^{\mathcal{S}}$ and $\mathcal{G}_b^{\mathcal{T}}$ for the two domains respectively, where an edge is established between user u and item i when item i was clicked by user u in that domain.

In addition to that, we utilize an external knowledge graph $\mathcal{G}_{kg}$ represented as $\{(h, r, t)|h, t \in \mathcal{E}, r \in \mathcal{R}_{kg}\}$, where a triplet (h, r, t) indicates a relation r exists between head entity h and tail entity t, $\mathcal{E}$ and $\mathcal{R}_{kg}$ is the entity set and the relation set of the knowledge graph respectively. It needs to be emphasized that both items on source domain and target domain are included as entities in this knowledge graph. In other words, we can exploit this knowledge

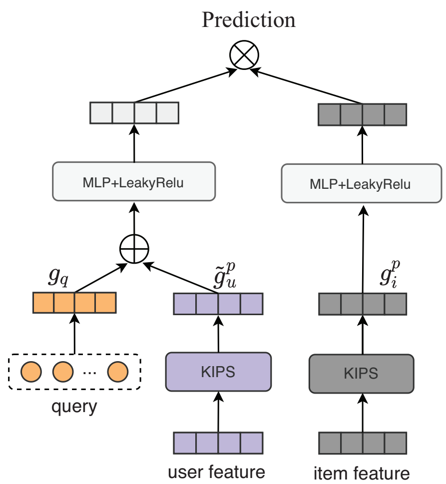


Figure 2: The two-tower structure of our proposed KIPS.

graph to establish high-order relations between the items across the two domains.

Similar to KGAT, we construct two collaborative knowledge graphs $\mathcal{G}^{\mathcal{S}}$, $\mathcal{G}^{\mathcal{T}}$ for the source and target domains respectively by merging both the corresponding user-item bipartite and knowledge graph together, *i.e.,* $\mathcal{G}^{\mathcal{S}} = \{(h,r,t)|h,t \in \bar{\mathcal{E}}, r \in \bar{\mathcal{R}}^{\mathcal{S}}\}$, $\mathcal{G}^{\mathcal{T}} = \{(h,r,t)|h,t \in \bar{\mathcal{E}}, r \in \bar{\mathcal{R}}^{\mathcal{T}}\}$, where $\bar{\mathcal{E}} = \mathcal{E} \cup \mathcal{U}$, $\bar{R}^{\mathcal{S}}$, $\bar{R}^{\mathcal{T}}$ are further augmented on the basis of $\mathcal{R}_{kg}$ by including the user-item interaction (*i.e.,* click) relations. Besides, we utilize $\mathcal{N}_h = \{(r,t)|(h,r,t) \in \mathcal{G}\}$ to denote the neighborhood of entity h on collaborative knowledge graph $\mathcal{G}$. It is worthwhile to highlight that users' search intents in the two domains are quite different, and we cannot simply regard the items in the two domains as from the same homogeneous space. Consequently, instead of building the user-item interactions of the two domains into a whole collaborative knowledge graph and sharing the embeddings of knowledge graph for different domains, we construct an individual graph for each domain. We conduct empirical analysis to verify the rationality in Section 4.5.1.

3.2 Overview

As illustrated in Figure 2, our proposed KIPS is a two-tower structure where both the query and user's personalized interest are considered in the user side. For the user embedding, we first divide the users' behavioural sequences into long-term and short-term based on the timestamp information of each interaction. For long-term and short-term interactions, we further construct different collaborative knowledge graphs. From the long-term perspective, user search intents could be similar across domains. Hence, we adopt a MMD loss to concretize this assumptions. On contrast, from the short-term perspective, users may have more historical search behaviors on the source domain. Therefore, it is possible that the source domain could convey the recent interest information that is not well expressed by the user on the target domain. After performing the information propagation over these collaborative knowledge graphs, we combine the short-term user's embedding on source domain into the target domain user's embedding to form a better user interest representation. Moreover, we introduce a kg-aware contrastive learning to enhance the correlation learning across domains. We aim to push the embeddings of relevant items based on the knowledge graph more similar, and pull the embeddings of irrelevant items more apart. At last, the user-query representations and item representations are passed through a network to get the final prediction score.

3.3 Knowledge-Aware Graph Attention Network

Inspired by KGAT and KCAN, we use a knowledge-aware attention mechanism to propagate information on the collaborative knowledge graph to capture the high-order relationships between entities. Specifically, we aggregate the l-ego network information of entity h within l-th layer by the following formula:

$$e^l_{\mathcal{N}_h} = \sum_{(r,t)\in\mathcal{N}_h} \alpha(h,r,t) e^{l-1}_t \tag{1}$$

e^{l-1}_t is the representation of entity t from the previous layer. $\alpha(h,r,t)$ is a relation-specific attention which measures correlation of entity t to entity h under the relationship r and determines how much information is propagated from entity t to entity h, we formulate the attention coefficient as followed:

$$\alpha(h,r,t) = softmax(f(W_r e^{l-1}_h + e^{l-1}_r, W_r e^{l-1}_t)) \tag{2}$$

where w_r is trainable parameter, $f(\cdot)$ is a cos similarity function, which calculates the similarity between $W_r e_h + e_r$ and $W_r e_t$.

Then the current node representation is combined with the neighbors representation to form a new representation:

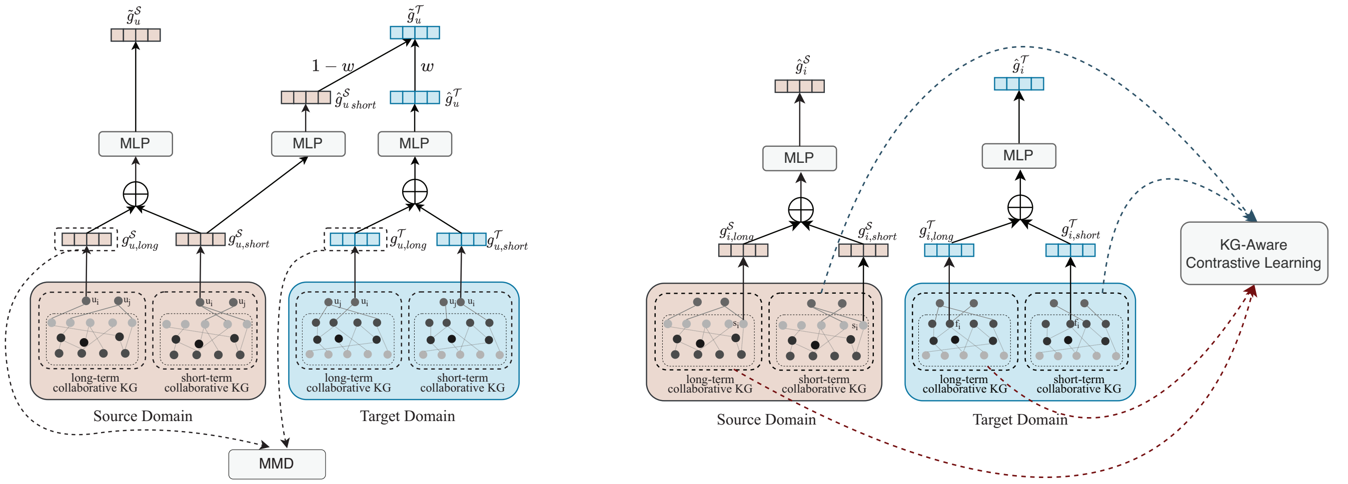
$$e^l_h = LeakyReLU(W^l(e^{l-1}_h \| e^l_{\mathcal{N}_h}) + b^l) \tag{3}$$

Where $\|$ means concatenation, LeakyReLU$(\cdot)$ is the activation function, and W^l, b^l are trainable parameters. After performing L layers, we then concatenate all representations of each layers as the final output of knowledge-aware graph attention network, i.e., $g_h = e^0_h\|...\|e^L_h$, where e^0_h is the initial embedding of entity h and L is the total number of layers. Through the knowledge-aware graph attention network, we can obtain the graph embedding of each node on the collaborative knowledge graph.

3.4 Long-Term and Short-Term Information Propagation

We take the items clicked by the user during her previous searches in the latest n days as short-term and those before n days as long-term ones. Here, parameter n is to control the granularity level of short-term interest learning. Then we divide the user interactions $\mathcal{N}^{\mathcal{S}}$, $\mathcal{N}^{\mathcal{T}}$ on each domain into two parts accordingly, *i.e.,* long-term and short-term, or written as $\mathcal{N}^{\mathcal{S}}_{short}$, $\mathcal{N}^{\mathcal{S}}_{long}$, $\mathcal{N}^{\mathcal{T}}_{short}$, $\mathcal{N}^{\mathcal{T}}_{long}$, Correspondingly, we can construct four collaborative knowledge graphs for this four sets of user-item interactions: $\mathcal{G}^{\mathcal{S}}_{short}$, $\mathcal{G}^{\mathcal{S}}_{long}$,$\mathcal{G}^{\mathcal{T}}_{short}$, and $\mathcal{G}^{\mathcal{T}}_{long}$.

Following the above procedure, We can obtain long-term and short-term graph embeddings of user and item in the both domains through knowledge-aware graph attention network $g^p_{o,long}$ and $g^p_{o,short}$, where $p \in \{\mathcal{S},\mathcal{T}\}$ means different domain, $o \in \{u,i\}$



(a) User Embedding (b) Item Embedding

Figure 3: The procedure of generating User Embedding and Item Embedding on two domains

means user or item. Afterwards, we adopt a MLP layer to transform the long-term and short-term graph embedding into a unified whole:

$$\hat{g}_o^p = MLP(g_{o,long}^p \| g_{o,short}^p) \tag{4}$$

where $\|$ is the concatenation operation.

We assume that the users' behaviors are highly relevant across domains in the long-term, but the source domain contains more differentiated information compared to the target domain in the short-term. Here, we conduct an analysis to verify the reasonability of this assumption on our stock-fund dataset, and detailed data statistics are shown in Table 1. We measure the similarity of items from different domains in terms of the distance on the knowledge graph, i.e., the closer distance means the higher similarity. According to the above divided long (short) interactions sequence, we calculate the shortest distance between the stock (source domain item) and fund (target domain item) that are clicked by a same user. Then the mean value of all stock-fund pairs' distance is considered as the distance of a user in the long (short) term. At last, we consider the average of all users' distance as the overall distance between source and target domains. The resultant distances are 3.2792 and 4.5434 for the long-term and short-term respectively, which means a more similar interaction behavior across domains in the long-term. Also, the sufficient interactions on the source domain could contain user's recent interest that are not timely well reflected on the target domain .

3.4.1 Long-term. From a long-term perspective, users' interaction behaviors are similar across domains, that is, the distribution of users' preferences in the two domains should tend to be consistent. Maximum Mean Discrepancy(MMD) has been widely used to reduce the distribution mismatch between source domain and target domain in domain adaptation tasks. It is an effective function without additional parameters[5]. The MMD between two input vectors can be calculated by followed formula:

$$\mathrm{MMD}\left(x^s, x^t\right) = \left(\frac{1}{n^{s2}} \sum_{i,j=1}^{n^s} k\left(x_i^s, x_j^s\right) - \frac{2}{n^s n^t} \sum_{i,j=1}^{n^s, n^t} k\left(x_i^s, x_j^t\right) + \frac{1}{n^{t2}} \sum_{i,j=1}^{n^t} k\left(x_i^t, x_j^t\right)\right)^{\frac{1}{2}} \tag{5}$$

where $k(\cdot,\cdot)$ is the gram-matrix of all possible kernels in the data space, and we adopt Gaussian kernel, i.e., $k(x_i, x_j) = \exp(-\frac{\|x_i - x_j\|^2}{2s^2})$, which is a universal kernel function[23].

We adopt MMD as an auxiliary loss to push user long-term graph embedding $g_{u\ long}^{\mathcal{S}}$, $g_{u\ long}^{\mathcal{T}}$ on this two domains closer, and facilitate the bidirectional propagation of information between the two domains:

$$\mathcal{L}_1 = MMD(g_{u,long}^{\mathcal{S}}, g_{u,long}^{\mathcal{T}}) \tag{6}$$

3.4.2 Short-term. In the short-term, it is possible that user preferences on the source domain are not yet reflected on the target domain, which can help to explore users' preference on target domain. Therefore, we fuse the users' short-term embedding on source domain into the users' embedding on target domain.

As Figure 3 shows, after performing a MLP transformation for $g_{u,short}^{\mathcal{S}}$: $\hat{g}_{u,short}^{\mathcal{S}} = MLP(g_{u,short}^{\mathcal{S}})$, We apply an attention mechanism to generate the new user embedding on target domain:

$$\tilde{g}_u^{\mathcal{T}} = w\hat{g}_u^{\mathcal{T}} + (1-w)\hat{g}_{u,short}^{\mathcal{S}} \tag{7}$$

w is an automatically trainable parameter that determines how much information is transferred from the source domain to the target domain.

Note that we consider that the user's long-term embeddings of both domains are similar to each other. Therefore, we no longer additionally pass the long-term information on the source domain

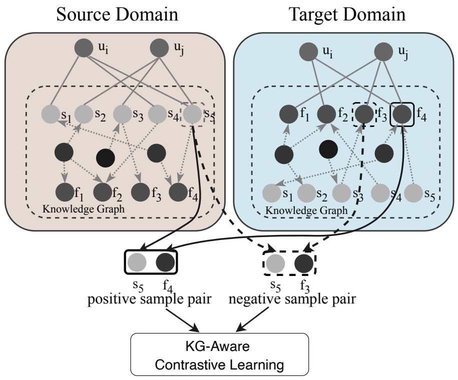


Figure 4: The network structure of KG-aware contrastive learning.

to the target domain. Besides, the short-term information propagating is happened only from source domain to target domain, so the overall structure is asymmetrical. Since we assume that the information in the short-term of target domain is not of great help to source domain. As a result, we do not make any further process for the user embedding of source domain.

3.5 KG-aware Contrastive Learning

As emphasized in the previous section, our knowledge graph contains both entities of source domain and target domain. We aim to push the embeddings of related items from different domains more similar and pull the embedding of unrelated items more different. For example, a stock is key invest of a fund, i.e., there is an edge in the knowledge graph that links these two entities. Though these two entities come from different domains, the graph embeddings of these two entities should be closer. In order to enhance the correlation between domains, we introduce a knowledge graph-based contrastive learning loss:

$$\mathcal{L}_2 = \sum_{p\in\{\mathcal{S},\mathcal{T}\}} -log(\sigma({g_i^p}^T g_j^p)) - \sum_{k=1}^{N_{neg}} log(\sigma(-{g_i^p}^T g_k^p)) \quad (8)$$

where g_i^p is the graph embedding of current item in p domain. g_j^p and g_k^p denotes the positive sample and negative sample graph embedding of current item i respectively.

Both positive and negative samples are from the opposite domain. We sample one entity directly linked with current item on the knowledge graph as the positive sampled nodes with a certain probability P. The probability P is determined by the properties of the knowledge graph, such as the percentage of a fund's key invest stocks. For the negative samples, to guarantee the sampled nodes are independent of the current item, we sample N_{neg} negative samples nodes except the current item's 5-hop neighbours on knowledge graph. For example, as shown in Figure 4, the current item is fund f_5, we sample the stock s_4 as the positive sample, because s_4 is directly related to f_5 on knowledge graph. On the other hand, (f_5, s_5) is a negative sample pair for their complete irrelevance.

3.6 Objective and Training

A query is represented as a sequence of words: $q = [w_1, w_2, ..., w_{n_q}]$. A word can be mapped into a dense embedding vector x_i via a word embedding table. Then we simply take the mean of the word embeddings as the query embedding: $e_q = \frac{1}{n_q}\sum x_i$.

As shown in Figure 2, our model is a two-tower structure. We concatenate the user graph embedding $\tilde{e}_u^p$ and query embedding e_q, where $p \in \mathcal{S}, \mathcal{T}$ and feed it into a neural network to get a unified embedding of user and query as the left tower. While the right tower is the item embedding processed by a neural network. The probability of user u click item i through query q can be calculated by:

$$\hat{y}_{uiq}^p = f(\tilde{g}_u^p \| e_q)\hat{g}_i^p \quad (9)$$

Then we adopt the binary cross entropy as the loss function for the CTR prediction task:

$$\mathcal{L}_{pre} = -\sum_{p\in\{\mathcal{S},\mathcal{T}\}} y_{uiq}^p log(\hat{y}_{uiq}^p) + (1 - y_{uiq}^p) log(1 - \hat{y}_{uiq}^p) \quad (10)$$

The final loss function is the combination of the three parts, i.e., the prediction loss $\mathcal{L}_{pre}$, the MMD between two user long-term graph embeddings $\mathcal{L}_1$ and the knowledge graph based information enhancement loss $\mathcal{L}_2$:

$$\mathcal{L} = \mathcal{L}_{pre} + \mathcal{L}_1 + \mathcal{L}_2 + \lambda\|\Theta\|_2^2 \quad (11)$$

where λ is a hyper-parameter, $\lambda\|\Theta\|_2^2$ is the L2-regularizer of parameters and embedding. We adopt Adam optimizer to minimize the loss.

4 EXPERIMENTS

In this section, we conduct extensive experiments over two real-world product search datasets for performance evaluation.

4.1 Dataset

We conduct offline evaluations on a large-scale real world dataset collected from Alipay digital finance search platform. This dataset contains consecutive exposure and click logs in December 2021. We have extracted multiple domains in the dataset based on the categorization of these financial items: funds, stocks and fund managers. We then divide them into two groups: the first group treats stocks as the source domain while the target domain is funds; the second group takes funds as the source domain and fund managers as the target domain. Note that there are overlapping users between source domain and target domain, but each item may only exist in one single domain. We take 21 consecutive days of data as the training set, and the following two days as the validation set and test set respectively. To ensure the denseness of the source domain, we filtered the datasets to retain users with at least 10 click records on the source domain. In addition, we filter a test subset that contains users that do not exist in the training set of target domain, and named as cold-start test set. Table 1 shows the basic statistics of our offline data. Besides, there is a shared knowledge graph covers the entities of all domains and the relations between entities. The detailed knowledge graph data statistics are shown in Table 2.

Table 1: The statistics of the two datasets. #records: the number of user search behaviors; #clk: the number of clicks made by the users; Sparsity: the ratio of #clk to #records.

Dataset	Stock-Fund				Fund-Fund Manager			
	Source Domain	Target Domain	Test	Cold-Start Test	Source Domain	Target Domain	Test	Cold-Start Test
#users	66,905	64,123	37,950	11,165	81,361	80,063	37,066	26,167
#items	7,244	15,316	11,998	5,412	15,293	3,669	2,940	2,701
#records	20,225,847	17,938,856	955,934	16,8254	43,223,141	5,670,891	235,108	161,310
#clk	3,549,022	1,448,444	68,813	925	9,891,115	68,516	3,074	722
sparsity	17.547%	8.074%	7.199%	0.550%	22.884%	1.208%	1.307%	0.448%

Table 2: Knowledge Graph Statistics

Entity Type	Count	Edge Type	Count
Sector	392	Fund_Sector	21,290
Fund	15,784	FundManager_Fund	20,997
FundManager	3,159	Fund_Index	1,901
Index	488	Fund_Industry	956
Industry	185	Stock_Fund	128,981
Stock	6,249	Stock_Sector	42,376
		Stock_Industry	5,713

4.2 Experimental Setting

4.2.1 Baseline Methods. We compare KIPS with two categories of baseline: three single-domain methods, KGAT[27], GEPS[31], DHGAT[19] and three cross-domain methods, CoNet[10], WE-CAN[20], CD-GNN[17].

- **KGAT** combines knowledge graph with collaborative information, while introducing graph attention mechanisms into collaborative knowledge graphs and training in an end-to-end manner.
- **GEPS** integrates click-graph features into a unified neural ranking model and combines heterogeneous external information with meta-paths into this method to improve search results.
- **DHGAT** attentively adopts heterogeneous and homogeneous neighbors of heterogeneous graph nodes for search, and relieves the long-tail phenomenon.
- **CoNet** is the collaborative cross networks, which establishes cross connections to transfer knowledge across domains in simple multi-layer feedforward networks.
- **WE-CAN** introduces a new multi-task regularizer based on Wasserstein distance to help extract both domain-shared and domain-private features for multiple domains.
- **CD-GNN** designes a domain invariant layer to indiscriminate the source and target domains, and optimizes the learning task of both domains simultaneously based on graph embedding.

KGAT, CoNet and CD-GNN are designed for recommendation tasks and have no query information in the original model. For a fair comparison, we combine the same query processing methods as KIPS into these method, i.e., the mean value of the word embeddings is regarded as a query embedding and incorporated as an additional feature for the final prediction. Note that the input graph of KGAT only combines the source domain interactions and knowledge graph.

4.2.2 Evaluation Protocol. We adopt three performance metrics: Area Under the receiver operating characteristic Curve (AUC), Group AUC (GAUC) and NDCG@10, which are widely used for offline evaluation of CTR prediction tasks. In our experiment, GAUC can calculate the AUC between predictions and ground truths under each user, and we combine the results of all users by a linear combination with the search frequencies as the weights.

4.3 Overall Performance

The overall performance comparison results are shown in Table 3. Note that for the Single-Domain methods, we only train them on the target domain and report the performance on target domain; for the Cross-Domain methods, we train them on both domains but also report the result on target domain. We make the following observations from the results:

Firstly, KIPS consistently outperforms all methods on all test sets. In particular, compared to the strongest baseline, the AUC, GAUC, NDCG@10 of KIPS are improved by 0.97%, 1.63%, 0.72%, 0.55%, 0.63%, 2.22% on the Stock-Fund test set and Fund-Fund Manager test set, respectively. With distinct long-term and short-term user information propagation patterns and kg-aware contrastive learning, KIPS can efficiently combine information from the source domain to the target domain. Moreover, compared with KGAT, KIPS integrates the collaborative signals on both domains and alleviates the data sparsity problem on the target domain. KIPS differs from the traditional cross-domain methods such as WE-CAN in that we not only connect the two domains through common users, but also enable the transfer of information between the domains through the knowledge graph. Furthermore, we find that KIPS also has a better performance in all metrics on the cold-start test set, especially AUC.

Secondly, KGAT is the best performed model in the single-domain methods, although there is no complicated process of query information. However, KGAT is the unique method that incorporates knowledge graph, which means that it actually integrates information from the other domains through the knowledge graph in our setting, and this verifies the significance of collaborative signals for transferring knowledge. It is worth mentioned that KGAT even outperforms some cross domain methods. The global performance

Table 3: Overall performance of baselines and KIPS. The best and second best results are highlighted in boldface and underline respectively. All reported improvements over baseline methods are statistically significant at a 0.05 level.

Methods		Stock-Fund						Fund-Fund Manager					
		Test			Cold-Start Test			Test			Cold-Start Test		
		AUC	GAUC	NDCG	AUC	GAUC	NDCG	AUC	GAUC	NDCG	AUC	GAUC	NDCG
Single Domain	GEPS	0.8702	0.6602	0.7903	0.8034	0.6458	0.7240	0.8435	0.7036	0.8072	0.6571	0.6839	0.8115
	DHGAT	0.8949	0.7464	0.8407	0.8281	0.6568	0.7492	0.8452	0.6316	0.7721	0.7350	0.5962	0.7711
	KGAT	0.9103	0.7913	0.8734	0.8573	0.7017	0.8192	0.8629	0.7012	0.8252	0.8065	0.7010	0.8225
Cross Domain	CoNet	0.8728	0.7200	0.8069	0.8081	0.6565	0.7662	0.8036	0.6325	0.8093	0.6566	0.6274	0.7988
	CD-GNN	0.8948	0.7642	0.8338	0.8049	0.6630	0.7524	0.8492	0.6697	0.7925	0.7483	0.6573	0.7723
	WE-CAN	0.9234	0.8085	0.8900	0.8433	0.7068	0.8288	0.8782	0.7104	0.8182	0.8136	0.6688	0.7962
	KIPS	**0.9324**	**0.8216**	**0.8964**	**0.8805**	**0.7125**	**0.8323**	**0.8830**	**0.7148**	**0.8435**	**0.8221**	**0.7092**	**0.8301**

of DHGAT is better than GEPS, which indicates that the reasonable use of heterogeneous graph structure and the successfully aggregation of information from homogeneous and heterogeneous neighbor nodes on knowledge graph can effectively improve the prediction accuracy.

At last, the structure of WE-CAN is designed to be similar to a multi-task model, enabling the full use of information in a specific domain while effectively combining shared information. WE-CAN achieves sub-optimal performance results over Cross-Domain methods, indicating that such a structure facilitates information propagation across domains. The poor performance of CoNet inspires us that combining cross-domain information is not always effective, as the unreasonable introduction of information from other domains would add noise that the model cannot handle. Besides, the mediocre performance of CD-GNN shows that the obfuscated domain strategy is not entirely suitable for our scenario.

4.4 Ablation Study

In this part, we evaluate the effect of different components of KIPS on the Stock-Fund dataset. In particular, we compare the original model with five variants:(1) KIPS$_{w/o\ MMD}$. It removes the long-term MMD loss of KIPS. (2) KIPS$_{w/o\ short_trans}$. It removes the short-term information transfer from source domain to target domain. (3) KIPS$_{w/o\ kgc}$. It removes the kg-aware contrastive learning between the two domains. (4) KIPS$_{w/o\ cross}$. It removes all information propagation strategies across domains. (5) KIPS$_{w/o\ query}$. It removes the query embedding of the final prediction.

We summarize the results in Table 4 and have the following observations. Firstly, it demonstrates that the MMD, short-term information transfer and kg-aware contrastive learning have different contributions for the cross-domain information propagation. Besides, KIPS$_{w/o\ cross}$ degenerates into KGAT trained in the source and target domains simultaneously, but performs worse than KGAT, which indicates that inappropriate introduction of information from other domains is equivalent to noise. Finally, although we only have a simple processing of query, it is effective of the introduction of query information for the final prediction.

4.5 Effectiveness Analysis

4.5.1 Effectiveness of different transformation manners and the separate collaborative knowledge graphs. In our work, we expect the long-term information is transferred in dual directions, but the short-term information is propagated from source domain to target domain. Thus, we can make full and effective use of information across domains. To validate the effectiveness of this module, we design two variant of KIPS: KIPS$_{only_trans}$ and KIPS$_{only_MMD}$.

The KIPS$_{only_trans}$ transfers the both short-term and long-term information from source domain to target domain, i.e., the Eq.(7) is changed into $\tilde{g}_u^{\mathcal{T}} = w\hat{g}_u^{\mathcal{T}} + (1-w)\hat{g}_u^{\mathcal{S}}$. Moreover, we remove the MMD auxiliary loss $\mathcal{L}_1$ in KIPS$_{only_trans}$. Correspondingly, the KIPS$_{only_MMD}$ discards the short-term transformation, i.e. $\tilde{g}_u^{\mathcal{T}} = \hat{g}_u^{\mathcal{T}}$, and the Eq.(6) is modified to $\mathcal{L}_1 = MMD(\hat{g}_u^{\mathcal{S}}, \hat{g}_u^{\mathcal{T}})$. The results of these two variants are shown in Table 5. The performance gains of the original model indicates that it is effective to transfer information in the long-term and short-term in different ways, which are accommodated with the dynamic nature of user interests and item properties.

Besides, we conduct an additional experiment that builds the users' interactions from both source and target domains into a whole collaborative knowledge graph, and result is also shown in Table 5, i.e., KIPS$_{whole_graph}$. This variant means that we share the graph embeddings of item entities for the two domains. Note that KIPS$_{whole_graph}$ includes the source and target domains interactions and performs information transfer same as KIPS, which is different from KGAT. We can conclude that the performance of KIPS$_{whole_graph}$ is worse than the original KIPS. In other words, because of the heterogeneity of items from different domains, a whole collaborative knowledge graph cannot provide excellent knowledge transformation despite the connection of knowledge graph.

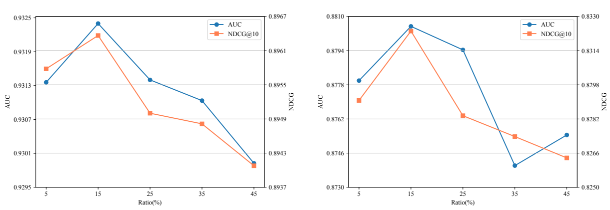
4.5.2 Parameter Analysis. Figure 5 plots the performance of different ratios of short-term data to all training data. The blue line indicates the AUC, while the orange line represents the NDCG@10. It is shown that the different proportions of long-term and short-term data affect the final prediction results. We observe that the best performance is obtained when the short-term data holds 15% either on test set or cold-start test set.

Table 4: Performance comparison for KIPS and its five variants.

Model	Test			Cold-Start Test		
	AUC	GAUC	NDCG@10	AUC	GAUC	NDCG@10
KIPS	**0.9324**	**0.8216**	**0.8964**	**0.8805**	**0.7125**	**0.8323**
$\text{KIPS}_{w/o\ MMD}$	0.9303	0.8140	0.8910	0.8700	0.7024	0.8260
$\text{KIPS}_{w/o\ short\_trans}$	0.9270	0.8113	0.8885	0.8573	0.7118	0.8274
$\text{KIPS}_{w/o\ kgc}$	0.9165	0.7985	0.8738	0.8350	0.7052	0.8213
$\text{KIPS}_{w/o\ cross}$	0.9058	0.7881	0.8653	0.8616	0.6979	0.8087
$\text{KIPS}_{w/o\ query}$	0.9017	0.7609	0.8405	0.8583	0.6553	0.7447

Table 5: Effective of different transformation manners for long and short term interactions, and the rationality of building collaborative knowledge graph for different domains

Model	Test			Cold-Start Test		
	AUC	GAUC	NDCG@10	AUC	GAUC	NDCG@10
KIPS	**0.9324**	**0.8216**	**0.8964**	**0.8805**	**0.7125**	**0.8323**
KIPS_{only_trans}	0.9289	0.8127	0.8917	0.8760	0.7111	0.8219
KIPS_{only_MMD}	0.9276	0.8140	0.8927	0.8723	0.7105	0.8266
$\text{KIPS}_{whole\_graph}$	0.9157	0.8026	0.8789	0.8485	0.7095	0.8232



(a) Test set (b) Cold-Start Test set

Figure 5: The AUC and NDCG@10 with different ratios of short-term data to all training data.

4.5.3 Effectiveness of KG-aware Contrastive Learning. In this subsection, we show the effectiveness of the kg-aware contrastive learning in a visual way. This module is designed for pushing the related item nodes more closer on the knowledge graph. As a result, we random sample $1,000$ pairs of items, one from the source domain and the other from the target domain on the test set. Note that each pair of items is a pair of first-order neighbour on the knowledge graph and have been clicked by the same user. According to our assumption, although the two items are from different domains, they can be jointly trained through the kg-aware contrastive learning. Therefore, the representations of a item pair are supposed to be closer than without this module.

We trained on the models containing the kg-aware contrastive learning loss, and not containing this loss, respectively, to obtain embeddings of these $1,000$ pairs of items. Then we calculate the cosine similarity between the $1,000$ pairs of item embeddings and draw the frequency distribution histogram showed as Figure 6. Here, $\text{KIPS}_{w/o\ kgc}$ means not containing the kg-aware contrastive learning, KIPS means the complete model. The frequency distribution histogram shows intuitively that the complete KIPS has a

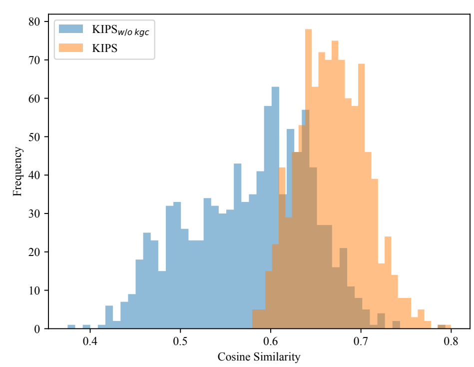


Figure 6: The frequency distribution histogram of cosine similarity between 1,000 pairs of item embeddings.

higher cosine similarity in the whole. Note that the average value of cosine similarity of $\text{KIPS}_{w/o\ kgc}$ is 0.5772 while KIPS is 0.6683, but the minimal values are 0.3750, 0.5795 respectively. This demonstrates that the distribution of KIPS is more concentrated. Hence, the addition of kg-aware contrastive learning promotes a higher similarity of the representations of related items from different domains, which conforms to our expectation.

4.6 Online A/B Test

We deploy our proposed model on the Alipay search platform providing cross-domain search service of Fund and Stock, and conduct online A/B test the Fund search scenario(i.e., target domain). One bucket is KIPS, and another is the latest model deployed online. The collaborative knowledge graphs cover interactions of three weeks and can be updated everyday. KIPS achieves performance gains of 0.45% and 1.2% improvement on UV-CTR and UV-CTCVR

Table 6: The relative improvements on Alipay on the Fund search scenario on the target crowd

Metric	UV-CTR	UV-CTCVR
Relative Improvement	+3.15%	+10.52%

(i.e., the number of clicked/conversion users divided by the number of viewed users). Furthermore, we evaluate KIPS on the target crowd who have interacted on the Stock search scenario(i.e., source domain) in the last three weeks. The results of relative improvements compared to baseline are shown in Table 6, indicating the effectiveness of information transfer across domains.

5 CONCLUSION

A large-scale product search system calls more for personlization since the potential candidate items that are relevant to the user's query are numerous. However, one can not make brick without straw. The same goes for personlized product search. In this paper, we aim to exploit the historical search behaviors of a user for better personalized search intent understanding. Facing the possible semantic gap and data sparsity problem, we resort to a marriage between cross-domain transfer learning and knowledge graph. The proposed KIPS have demonstrated significant superiority over two real-world large-scale product search datasets as well as the online A/B testing. In the future, we plan to extend KIPS with more than two domains for better search accuracy.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (No. 61872278); and Young Top-notch Talent Cultivation Program of Hubei Province. Chenliang Li is the corresponding author.

REFERENCES

[1] Qingyao Ai, Daniel N. Hill, S. V. N. Vishwanathan, and W. Bruce Croft. 2019. A Zero Attention Model for Personalized Product Search. In *CIKM*. 379–388.

[2] Qingyao Ai, Yongfeng Zhang, Keping Bi, Xu Chen, and W. Bruce Croft. 2017. Learning a Hierarchical Embedding Model for Personalized Product Search. In *SIGIR*. 645–654.

[3] Ye Bi, Liqiang Song, Mengqiu Yao, Zhenyu Wu, Jianming Wang, and Jing Xiao. 2020. DCDIR: A Deep Cross-Domain Recommendation System for Cold Start Users in Insurance Domain. In *SIGIR*. 1661–1664.

[4] Huizhong Duan and ChengXiang Zhai. 2015. Mining Coordinated Intent Representation for Entity Search and Recommendation. In *CIKM*. 333–342.

[5] Muhammad Ghifary, W. Bastiaan Kleijn, and Mengjie Zhang. 2014. Domain Adaptive Neural Networks for Object Recognition. In *PRICAI (Lecture Notes in Computer Science, Vol. 8862)*. Springer, 898–904.

[6] Yangyang Guo, Zhiyong Cheng, Liqiang Nie, Yinglong Wang, Jun Ma, and Mohan S. Kankanhalli. 2019. Attentive Long Short-Term Preference Modeling for Personalized Product Search. *ACM Trans. Inf. Syst.* 37, 2 (2019), 19:1–19:27.

[7] Yangyang Guo, Zhiyong Cheng, Liqiang Nie, Xin-Shun Xu, and Mohan S. Kankanhalli. 2018. Multi-modal Preference Modeling for Product Search. In *2018 ACM Multimedia Conference on Multimedia Conference, MM 2018, Seoul, Republic of Korea, October 22-26, 2018*. 1865–1873.

[8] William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *NIPS*. 1024–1034.

[9] Ming He, Jiuling Zhang, Peng Yang, and Kaisheng Yao. 2018. Robust Transfer Learning for Cross-domain Collaborative Filtering Using Multiple Rating Patterns Approximation. In *WSDM*. 225–233.

[10] Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. CoNet: Collaborative Cross Networks for Cross-Domain Recommendation. In *CIKM*. 667–676.

[11] Guangneng Hu, Yu Zhang, and Qiang Yang. 2019. Transfer Meets Hybrid: A Synthetic Approach for Cross-Domain Collaborative Filtering with Text. In *WWW*. 2822–2829.

[12] Yujing Hu, Qing Da, Anxiang Zeng, Yang Yu, and Yinghui Xu. 2018. Reinforcement Learning to Rank in E-Commerce Search Engine: Formalization, Analysis, and Application. In *KDD*. 368–377.

[13] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*. OpenReview.net.

[14] Bin Li, Qiang Yang, and Xiangyang Xue. 2009. Transfer learning for collaborative filtering via a rating-matrix generative model. In *ICML*, Vol. 382. 617–624.

[15] Pan Li, Zhichao Jiang, Maofei Que, Yao Hu, and Alexander Tuzhilin. 2021. Dual Attentive Sequential Learning for Cross-Domain Click-Through Rate Prediction. In *KDD*. 3172–3180.

[16] Shang Liu, Wanli Gu, Gao Cong, and Fuzheng Zhang. 2020. Structural Relationship Representation Learning with Graph Embedding for Personalized Product Search. In *CIKM*. 915–924.

[17] Ziqi Liu, Yue Shen, Xiaocheng Cheng, Qiang Li, Jianping Wei, Zhiqiang Zhang, Dong Wang, Xiaodong Zeng, Jinjie Gu, and Jun Zhou. 2021. Learning Representations of Inactive Users: A Cross Domain Approach with Graph Neural Networks. In *CIKM*. 3278–3282.

[18] Lili Mou, Zhao Meng, Rui Yan, Ge Li, Yan Xu, Lu Zhang, and Zhi Jin. 2016. How Transferable are Neural Networks in NLP Applications?. In *EMNLP*. The Association for Computational Linguistics, 479–489.

[19] Xichuan Niu, Bofang Li, Chenliang Li, Rong Xiao, Haochuan Sun, Hongbo Deng, and Zhenzhong Chen. 2020. A Dual Heterogeneous Graph Attention Network to Improve Long-Tail Performance for Shop Search in E-Commerce. In *KDD*. 3405–3415.

[20] Minghui Qiu, Bo Wang, Cen Chen, Xiaoyi Zeng, Jun Huang, Deng Cai, Jingren Zhou, and Forrest Sheng Bao. 2019. Cross-domain Attention Network with Wasserstein Regularizers for E-commerce Search. In *CIKM*. 2509–2515.

[21] Dimitrios Rafailidis and Fabio Crestani. 2017. A Collaborative Ranking Model for Cross-Domain Recommendations. In *CIKM*. 2263–2266.

[22] Ajit Paul Singh and Geoffrey J. Gordon. 2008. Relational learning via collective matrix factorization. In *KDD*. 650–658.

[23] Ingo Steinwart. 2001. On the Influence of the Kernel on the Consistency of Support Vector Machines. *J. Mach. Learn. Res.* 2 (2001), 67–93.

[24] Ke Tu, Peng Cui, Daixin Wang, Zhiqiang Zhang, Jun Zhou, Yuan Qi, and Wenwu Zhu. 2021. Conditional Graph Attention Networks for Distilling and Refining Knowledge Graphs in Recommendation. In *CIKM*. 1834–1843.

[25] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph Attention Networks. *CoRR* abs/1710.10903 (2017). arXiv:1710.10903

[26] Shoujin Wang, Liang Hu, Yan Wang, Xiangnan He, Quan Z. Sheng, Mehmet A. Orgun, Longbing Cao, Francesco Ricci, and Philip S. Yu. 2021. Graph Learning based Recommender Systems: A Review. In *IJCAI*. ijcai.org, 4644–4652.

[27] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. In *KDD*. 950–958.

[28] Teng Xiao, Jiaxin Ren, Zaiqiao Meng, Huan Sun, and Shangsong Liang. 2019. Dynamic Bayesian Metric Learning for Personalized Product Search. In *CIKM*. 1693–1702.

[29] Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2017. Transfer Learning for Sequence Tagging with Hierarchical Recurrent Networks. In *ICLR*. OpenReview.net.

[30] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks. In *NIPS*. 3320–3328.

[31] Yuan Zhang, Dong Wang, and Yan Zhang. 2019. Neural IR Meets Graph Embedding: A Ranking Model for Product Search. In *WWW*. ACM, 2390–2400.

[32] Ziwei Zhang, Peng Cui, and Wenwu Zhu. 2022. Deep Learning on Graphs: A Survey. *IEEE Trans. Knowl. Data Eng.* 34, 1 (2022), 249–270.

[33] Cheng Zhao, Chenliang Li, Rong Xiao, Hongbo Deng, and Aixin Sun. 2020. CATN: Cross-Domain Recommendation for Cold-Start Users via Aspect Transfer Network. In *SIGIR*. 229–238.

[34] Feng Zhu, Yan Wang, Chaochao Chen, Guanfeng Liu, and Xiaolin Zheng. 2020. A Graphical and Attentional Framework for Dual-Target Cross-Domain Recommendation. In *IJCAI*. ijcai.org, 3001–3008.

[35] Feng Zhu, Yan Wang, Chaochao Chen, Jun Zhou, Longfei Li, and Guanfeng Liu. 2021. Cross-Domain Recommendation: Challenges, Progress, and Prospects. In *IJCAI*. ijcai.org, 4721–4728.