BOMGraph: Boosting Multi-scenario E-commerce Search with a Unified Graph Neural Network

Shuai Fan
School of Informatics,
Xiamen University,
Xiamen, China
fanshuai.fan@alibaba-inc.com

Jinping Gou
Alibaba Group,
Hangzhou, China
bixin.gjp@alibaba-inc.com

Yang Li
Institute of Artificial Intelligence,
Xiamen University,
Xiamen, China
goatxy@stu.xmu.edu.cn

Jiaxing Bai
School of Aerospace Engineering,
Xiamen University,
Xiamen, China
jiaxingbai@stu.xmu.edu.cn

Chen Lin*
School of Informatics,
Xiamen University,
Xiamen, China
chenlin@xmu.edu.cn

Wanxian Guan
Xubin Li†
Hongbo Deng
Jian Xu
Bo Zheng
Alibaba Group
Hangzhou, China
lxb204722@alibaba-inc.com

ABSTRACT

Mobile Taobao Application delivers search services on multiple scenarios that take textual, visual, or product queries. This paper aims to propose a unified graph neural network for these search scenarios to leverage data from multiple scenarios and jointly optimize search performances with less training and maintenance costs. Towards this end, this paper proposes BOMGraph, **BO**osting **M**ulti-scenario E-commerce Search with a unified **Graph** neural network. BOMGraph is embodied with several components to address challenges in multi-scenario search. It captures heterogeneous information flow across scenarios by inter-scenario and intra-scenario metapaths. It learns robust item representations by disentangling specific characteristics for different scenarios and encoding common knowledge across scenarios. It alleviates label scarcity and long-tail problems in scenarios with low traffic by contrastive learning with cross-scenario augmentation. BOMGraph has been deployed in production by Alibaba's E-commerce search advertising platform. Both offline evaluations and online A/B tests demonstrate the effectiveness of BOMGraph.

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

KEYWORDS

Heterogeneous graph neural networks; E-commerce search; Multi-scenario learning

ACM Reference Format:
Shuai Fan, Jinping Gou, Yang Li, Jiaxing Bai, Chen Lin, Wanxian Guan, Xubin Li, Hongbo Deng, Jian Xu, and Bo Zheng. 2023. BOMGraph: Boosting Multi-scenario E-commerce Search with a Unified Graph Neural Network. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3583780.3614794

1 INTRODUCTION

Nowadays, it has been a common practice for E-commerce platforms to provide search services in multiple scenarios. For example, mobile Taobao, one of China's largest E-commerce applications, offers three search scenarios, namely "visual search", "textual search", and "similar product search". They are launched from different portal pages and take different queries. As shown in Figure 1, users can search in the landing page via a picture (i.e., "visual search"), or several keywords (i.e., "textual search"). Users can also click on a trigger item in the "discovery page" and search for similar products (i.e., "similar product search").

The problem of multi-scenario search has attracted an emerging interest from both academia and industry [7, 8, 14, 29, 37]. Recent studies show that, utilizing data from multiple search scenarios can improve the overall performance of different search scenarios, and alleviate the cold start problem for scenarios with low traffic. Multi-scenario search can be achieved either by building *independent models* for each scenario [8, 13], or by building a *unified model* for all scenarios [14, 17]. The advantage of using a unified model is that the maintenance resource is significantly less [9, 24, 37].

Our goal is to design *a unified Graph Neural Network* (GNN) for multi-scenario search in Mobile Taobao. The reason for resorting to GNN is three-fold. Firstly, unlike existing multi-scenario search problems [3, 14, 23, 37], our queries include more complex queries,

*Corresponding author. Supported by the National Key R&D Program of China (No. 2022ZD0160501), the Natural Science Foundation of China (No. 61972328), Alibaba Group through Alibaba Innovative Research program.

†Corresponding author.

CIKM '23, October 21–25, 2023, Birmingham, United Kingdom

ACM ISBN 979-8-4007-0124-5/23/10…$15.00
https://doi.org/10.1145/3583780.3614794

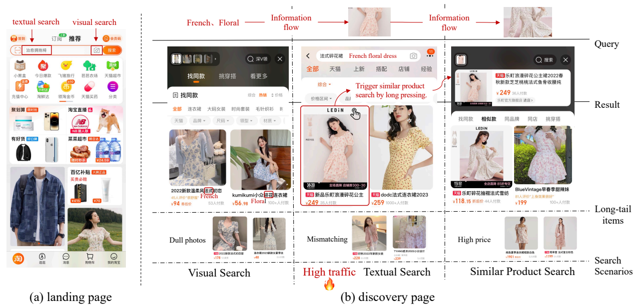

Figure 1: Multi-scenario search with multi-modal queries

e.g., product queries with multi-modal contents. With GNNs, the keyword queries and product queries can be naturally modeled as nodes in a unified graph, and thus different scenarios can conveniently share the same model. Otherwise, independent encoding modules are required to learn query representations, and these modules are not utilized across scenarios. Secondly, GNNs are powerful in capturing relationships [25, 30], and modeling the relationships between queries and products is crucial for search performance. Thirdly, a unified graph can reduce the time required to build and train individual graphs for each scenario.

While GNN-based E-commerce search systems have shown promising results [2, 10, 15], existing industrial solutions are restricted on a single scenario. To design a unified GNN for multi-scenario search, three key challenges must be addressed.

C1: heterogeneous information flow across scenarios An important property of mobile Taobao is the information flow across scenarios. Many users will alternate among search scenarios. As shown in Figure 1, a user wants to buy a dress, and she switches from "visual search", to "textual search", and to "similar product search". Since each search scenario focuses on a different modality, she is able to clarify her preferences (i.e., "pink, french floral, slip dress, luxurious fabric") on various aspects during this switching process. We can see that the information flow across search scenarios provides both research opportunities and challenges. On the one hand, it allows us to comprehend query intents and enrich training data for all scenarios. On the other hand, modeling heterogeneous information indistinguishably on the graph cannot capture the unique characteristics of each search scenario.

C2: learning scenario-robust item representations Although multiple scenarios share the same item universe, they emphasize different item features. For example, if a user wants to find a dress in the "visual search" scenario, the silhouette and style are most important in matching a query image. In the "similar product search" scenario, the texture of the fabric is most important in matching a trigger item. Therefore, to provide robust performance, the unified graph neural network should be able to learn item representations that encode the commonalities across scenarios and specific characteristics for different scenarios.

C3: insufficient click signals On the one hand, search performance suffers from insufficient click signals. This is more severe for low-traffic scenarios and long-tail items. On the other hand, the number of clicks received for items may vary in different scenarios. For example, in "visual search", items with dull photos are rarely clicked and more likely to be long-tail items, while in "similar product search", long-tail items are more likely to have high prices. This means that the unified graph neural network must deal with data sparsity and diverse long-tail items in each scenario, by properly transferring knowledge from other scenarios.

We propose BOMGraph, **BO**osting **M**ulti-scenario E-commerce search with a unified **Graph** neural network. To address **C1**, BOMGraph leverages heterogeneous information flow across scenarios by inter-scenario and intra-scenario metapaths. To address **C2**, on the representations fused from multiple scenarios by metapath propagation, BOMGraph identifies common knowledge and refines item representations by disentangled learning. To address **C3**, BOMGraph alleviates label scarcity and long-tail problems by cross-scenario data augmentation and contrastive learning.

Our main contributions are summarized below. (1) We study a novel multi-scenario learning problem that involves multi-modal E-commerce search. To the best of our knowledge, this problem has never been explored in the literature. (2) We present BOMGraph that models multi-scenario E-commerce search in a unified graph neural network, and it achieves superior performances on multiple scenarios at a limited resource cost. (3) Offline experiments on billion-scale real production data demonstrate that BOMGraph outperforms state-of-the-art competitors. BOMGraph has been fully deployed in production by Alibaba's E-commerce search advertising platform. Our online A/B tests for seven days show that BOMGraph produces a 2.55% RPM improvement over the existing solutions.

2 RELATED WORK

Multi-Task learning (MTL) [27, 34] and Cross-Domain Transfer Learning (CDTL) [32] have been widely studied in the literature, Multi-Scenario learning (MSL) [37], which can be seen as a special case of MTL, has shown strong performance in practical applications [3, 8, 14, 23, 24, 29, 37]. Despite the different purposes of CDTL, MTL and MSL, they all involve information sharing across domains/tasks/scenarios. Information sharing can be captured at model level and representation level. We briefly review related work based on their model architectures and representation learning.

Independent models Some pioneering works apply independent base models on each scenario and fuse the learned representations through various mapping modules, such as MLP [20], Transfomer [8] or GNN [16, 27]. The fusion can be learned from shared users [20] or non-shared users [13]. However, maintaining multiple different models would consume significant maintenance resources [24]. To solve this problem, recent studies resort to building a unified model for all scenarios.

Unified model Methods that build a unified model for multiple scenarios fall into two categories, modeling scenario relationships implicitly and explicitly. The former category implements separate prediction sub-modules on top of a shared bottom structure [14, 17, 36]. The latter category utilizes auxiliary networks [3], star topology networks [24], or additional transformation layers [23] to capture relationships between scenarios explicitly.

Another line of related studies is on representation learning for MSL and MTL. Recently, disentangled representation learning and contrastive learning have received considerable research interest.

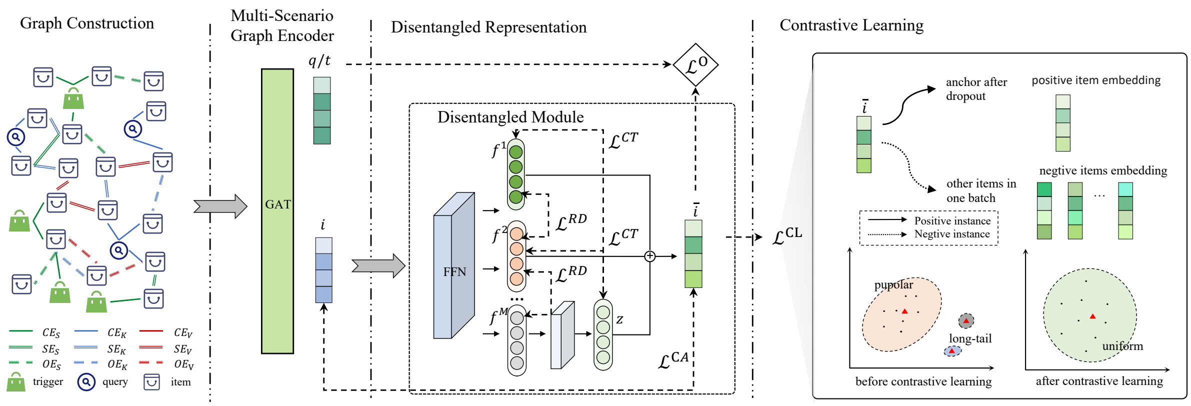

Figure 2: Overall framework of BOMGraph

Disentangled representation learning Disentanglement representation has been proven to be effective in the recommendation by disentangling user intentions through relationships between user/item [28] or given behavior sequences [18]. Disentangled representation learning is beginning to focus on cross-domain recommendation tasks. For example, DisenCDR [1] uses two mutual-information-based regularizers to disentangle scenario-shared information and scenario-specific information.

Contrastive representation learning Contrastive representation learning has shown extraordinary performance in single-scenario applications [12, 31, 35]. Unlike single-scenario recommendations, CCDR [33] designs three inter-scenario contrastive learning tasks, which can alleviate the long-tail problem. However, CCDR assumes that items should be represented similarly across scenarios, but this assumption does not hold for different or multi-modal scenarios.

Remarks Previous studies adopt extra modules to explicitly model scenario relationships, which increases the cost of computational resources. Instead, BOMGraph explicitly captures the information flow through different scenarios via metapath in a unified graph neural network, which is more resource efficient. Furthermore, there is much room to improve representation learning in a unified model for multi-scenario learning. In particular, in existing studies [1, 33], the scenario differences or even noise remains largely overshadowed by scenario relatedness.

3 METHODOLOGY

The overall framework of BOMGraph is depicted in Figure 2. First, a graph of queries, triggers and items is constructed from the multiple scenarios, and metapaths are defined (Section 3.1). The graph is fed into a multi-scenario graph encoder that obtains node embeddings by meta-path guided information propagation (Section 3.2). The item embeddings are refined by a disentangled representation module (Section 3.3). Finally, cross-scenario data augmentation and contrastive learning are incorporated (Section 3.4) in the training.

3.1 Graph Construction

We first construct a heterogeneous graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. **Nodes and node types**. $\mathcal{V}$ is a set of nodes, each node is associated with a D-dimensional embedding vector $\mathbf{v} \in \mathbf{R}^D$. There are three types of nodes in $\mathcal{G}$. When the node type is constrained, we will use t to represent a trigger node (i.e., a trigger item that is used to search products)[1], q to represent a keyword node (i.e., a query keyword), and i to represent a product node. **Edges and edge types**. $\mathcal{E} = \{e_{v,v\prime} | v \in \mathcal{V}, v\prime \in \mathcal{V}\}$ is a set of edges. Since we consider clicks and co-occurrences in three search scenarios (i.e., S for "similar product search", K for "textual search" and V for "visual search"), there are in total seven types of edges, $\{CE, \{OE_o\}, \{SE_o\} | o \in \{S, K, V\}\}$. CE represents click edges, which connect a trigger/keyword and an item that has been clicked at least once for the trigger/keyword query. $OE_o, o \in \{S, K, V\}$ represents co-occurrence edge under each search scenario S, K, V. The OE edge connects a pair of items that have been clicked by the same user under the same query, and their clicks happened in a search session of subsequent queries in 30 minutes. $SE_o, o \in \{S, K, V\}$ represents the similarity edge that connects two items that expose similar contents in each scenario S, K, V[2]. In the implementation, we add an SE edge if the cosine similarity between displayed contents exceeds 0.85.

Next, we define intra-scenario metapaths and inter-scenario metapaths to propagate information on $\mathcal{G}$.

Intra-Scenario metapath Intra-Scenario metapaths are defined inside each search scenario. The intuition is to collect scenario-specific collaborative feedback along the intra-scenario metapath, and allow information to transmit to relevant queries and items.

Driven by this intuition, in the "similar product search" scenario S and "textual search" scenario K, we define metapaths connecting queries (e.g., triggers t and keywords q) and items (i.e., i), along the click edges CE:

$$M_t^S = t \xrightarrow{CE} i \xrightarrow{CE} t', M_i^S = i \xrightarrow{CE} t \xrightarrow{CE}, i',$$
$$M_q^K = q \xrightarrow{CE} i \xrightarrow{CE} q', M_i^K = i \xrightarrow{CE} q \xrightarrow{CE} i'. \tag{1}$$

[1]Note that trigger items are always selected from the products in the display page. This means that for every trigger item in the item universe, it has a trigger node and a product node.

[2]In mobile Taobao application, different search scenarios may expose different item contents, e.g., different fragments of product descriptions, different images, and so on. The construction of similarity edges is based on the exposed contents. Thus, the similarity edges are different in each scenario.

For example, M_t^S starts from a trigger node t, passes an item node i that has been clicked (i.e. a click edge CE) for this trigger t, to another trigger node $t\prime$ that has clicked i . Thus, this metapath M_t^S collects the collaborative feedback in scenario S and links trigger items $t, t\prime$ with similar preferences (i.e., the same product is preferred for $t, t\prime$). M_i^S, M_q^K, M_i^K are defined in a similar manner.

In the "visual search" scenario V, due to the fact that users generally take photos that do not exist in the item universe, it is infeasible to construct query nodes in this scenario. Thus we can not define metapaths to combine query and item nodes. Instead, the metapath collects collaborative feedback through co-occurence and similarity edges.

$$M_i^V = i \xrightarrow{OE_V} i' \xrightarrow{SE_V} i'', \tag{2}$$

where the metapath M_i^V passes from an item i, through a relevant item i' that has been clicked with i at least once (i.e., a co-occurence edge OE), to another visually and textually similar item i'' (i.e., a similarity edge SE).

Inter-Scenario metapath Different from existing retrieval models based on heterogeneous network [6], in addition to intra-scenario metapaths, we define inter-scenario metapaths to capture cross-scenario information flow. The intuition is to connect items in different scenarios with similar contents through similarity edge, so that information can flow across scenarios and inter-scenario commonalities can be exploited. In particular, we define three metapaths across the three scenarios in different orders.

$$\begin{aligned} M_i^{S-K-V} &= i \xrightarrow{SE_K} i' \xrightarrow{SE_V} i'', \\ M_i^{K-V-S} &= i \xrightarrow{SE_V} i' \xrightarrow{SE_S} i'', \\ M_i^{V-S-K} &= i \xrightarrow{SE_S} i' \xrightarrow{SE_K} i''. \end{aligned} \tag{3}$$

For example, M_i^{S-K-V} describes information flow from an item i in the similar product search scenario S, via $i\prime$ which is similar in textual search K, to another similar item i'' in the visual search scenario V. Similarly, M_i^{K-V-S} travels from the textual scenario, through the visual search scenario, to the similar product search scenario. M_i^{V-S-K} starts from the visual search scenario and ends in textual search scenario. Note that the three inter-scenario metapaths utilize content similarity (i.e., SE edges) to aggregate information from different scenarios. It is unnecessary to cover all possible orderings and all node types. Instead, we only connect product nodes i, as the query node embeddings and trigger node embeddings will be updated by linking intra-scenario and inter-scenario metapaths.

3.2 Multi-Scenario Graph Encoder

Metapath-based sub-graph sampling Scalable GNNs learn node embeddings by aggregating from neighbors in the sampled subgraph [10]. To ensure that cross-scenario information is aggregated, we design metapath-based sub-graph sampling. The motivation is to distinguish information flow within and across scenarios. For each node $v \in \mathcal{V}$, where v can be a trigger/query/item node, we sample along each scenario's intra-scenario metapath that starts from v to form $G_v^{intra,o}|o \in \{S, K, V\}$. We also form $G_v^{inter,o}|o \in \{S, K, V\}$ by selecting nodes along each inter-scenario metapath that starts from scenario o. Specifically, we randomly sample three nodes in the one-hop neighbor and three nodes in the two-hop neighbor, respectively. To prevent over-exploiting data bias, if the sampled

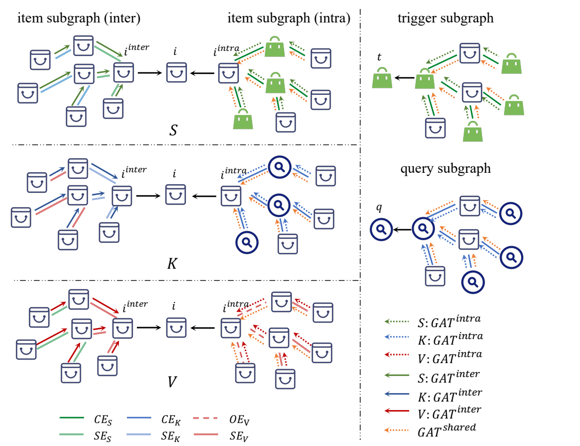

Figure 3: Multi-Scenario Graph Encoder. The item representations combine information from intra-scenario, inter-scenario, and scenario-shared information flow. The query and trigger representations are learned on their particular scenario.

neighbor constitutes a pair of positive pairs in the mini-batch, we discard it and re-sample until the sample size is achieved.

Graph node encoder Previous work[25] pointed out that the aggregation effect of GAT (Graph Attention Network) [25] is better than that of GCN [30], so we use GAT as the graph encoder backbone. Let's define $\mathbf{v} = GAT(\mathbf{v}^0, G_v)$ as the embedding of node v on the subgraph G_v by GAT, where $\mathbf{v}^0$ is the initialized embedding vector.

To initialize item node embedding $\mathbf{i}^0$ and trigger node embedding $\mathbf{t}^0$, we concatenate the one-hot ID embeddings, image feature embedding, and text feature embedding from product titles and pass them through a fully connected layer. The initial query node embedding $\mathbf{q}^0$ is obtained by concatenating ID embeddings and text feature embeddings.

For a query q, or a trigger node t, since their representations are only utilized on the particular scenario (i.e., q for "textual search" K and t for "similar product search" S), we should focus on intra-scenario information flow. Thus their embeddings are obtained in a scenario-specific manner, on intra-scenario metapath-guided subgraphs, as shown below:

$$\begin{aligned} \mathbf{q} &= W * GAT^{intra}(\mathbf{q}^0, G_q^{intra,K}), \\ \mathbf{t} &= W * GAT^{intra}(\mathbf{t}^0, G_t^{intra,S}). \end{aligned} \tag{4}$$

Since three scenarios share the same item universe, in deriving the item representation $\mathbf{i}$, we take into account intra-scenario and inter-scenario information flow. As shown in Figure 3, we first derive the item representation on each scenario. We utilize three GAT encoders, to combine information from intra-scenario, inter-scenario, and scenario-shared information flow.

$$\begin{aligned} \mathbf{i}^{intra,o} &= GAT^{intra,o}(\mathbf{i}^0, G_i^{intra,o}) + GAT^{shared}(\mathbf{i}^0, G_i^{intra,o}), \\ \mathbf{i}^{inter,o} &= GAT^{inter,o}(\mathbf{i}^0, G_i^{inter,o}), \\ \mathbf{i} &= W[\mathbf{i}^{inter,o} || \mathbf{i}^{intra,o}], o \in \{S, K, V\}. \end{aligned} \tag{5}$$

where $GAT^{intra,o}$ are GAT networks with scenario-specific parameters to learn item representations on intra-scenario metapath-guided subgraphs. GAT^{shared} is another GAT network on intra-scenario metapath-guided subgraphs. The parameters of GAT^{shared} are shared across scenarios. The role of GAT^{shared} is to exploit the correlations among scenarios by shared parameters. $GAT^{inter,o}$ is a GAT network that learns on inter-scenario metapath-guided subgraphs.

3.3 Disentangled Representation

We keep the query embedding and trigger embedding obtained by Equation 4 unchanged, and feed the item embedding obtained by Equation 5 to the disentangled representation module, which consists of three steps.

Representation Dissociation(RD) We argue that a more robust item representation must reflect the commonalities and subtle differences between scenarios, instead of simply fusing information across scenarios.

We thus fine-tune the item representation in Equation 5 by decoupling it into M independent counterparts. These counterparts can correspond to specific characteristics that different scenarios concentrate on. Note that we do not enforce M to be the number of scenarios, because it is possible that two scenarios favor the same set of item features (i.e., the same counterpart).

Formally, given $\mathbf{i}$ output by Equation 5, we project it into M feature spaces, as shown below:

$$\mathbf{f}^m = \frac{\sigma(W^m \mathbf{i} + b^m)}{\|\sigma(W^m \mathbf{i} + b^m)\|_2}, m = 1, 2, ..., M, \tag{6}$$

where W^m, b^m are the trainable parameters of space m.

As shown in Figure 2, to preserve that the $\mathbf{f}^m$ are independent counterparts, we introduce $\mathcal{L}^{RD}$, as shown below:

$$\mathcal{L}^{RD} = \sum_{\mathbf{i}\in\mathcal{B}} \sum_{m=1}^{M} -\log \frac{\exp(\mathbf{f}^m \cdot \mathbf{f}^m/\tau)}{\sum_{m'=1}^{M} \exp(\mathbf{f}^{m'} \cdot \mathbf{f}^m/\tau)}, \tag{7}$$

where τ is a temperature parameter.

Common knowledge Transfer(CT) We then use the counterparts to derive commonalities among scenarios. This can be achieved by transforming one counterpart and regulating it to reflect common knowledge. Without loss of generality, we can apply the transformation on the last orthogonal feature space $\mathbf{f}^M$. That is, we feed $\mathbf{f}^M$ to a feed-forward layer, as shown below:

$$\mathbf{z} = W\mathbf{f}^M + b, \tag{8}$$

where $\mathbf{z}$ is the transferred feature from $\mathbf{f}^M$, $W \in \mathbb{R}^{F\times F}$ and $b \in \mathbb{R}^F$ are the trainable parameters.

Since $\mathbf{z}$ reflects common knowledge among the counterparts, it should capture common patterns in $\mathbf{f}^m$. Inspired by the domain alignment loss [4], we regulate $\mathbf{z}$ to capture the element-wise variance of other counterparts:

$$\mathcal{L}^{CT} = \frac{1}{M-1}\frac{1}{F^2} \sum_{\mathbf{i}\in\mathcal{B}} \sum_{m=1}^{M-1} \sum_{(a,b)} (\mathbf{f}_a^m \cdot \mathbf{f}_b^m - \mathbf{z}_a \cdot \mathbf{z}_b), \tag{9}$$

where F is the dim of the disentangled feature, $\mathcal{B}$ is the mini-batch of samples, a, b are the column indexes, and $\mathbf{z}_a \cdot \mathbf{z}_b$ is the product of the a-th and b-th columns of vector $\mathbf{z}$.

Centroid Alignment(CA) We fine-tune item embeddings by merging the disentangled counterparts $\mathbf{f}^m$ and the common knowledge $\mathbf{z}$:

$$\bar{\mathbf{i}} = \sum_{m=1}^{M-1} \mathbf{f}^m + \mathbf{z}. \tag{10}$$

For a more stable performance, we can encourage the centroid of refined item embeddings $\bar{\mathbf{i}}$ to align with the centroid of original item embeddings $\mathbf{i}$ and avoid $\bar{\mathbf{i}}$ drifting to a distant region.

$$\mathcal{L}^{CA} = \sum_{\mathbf{i}\in\mathcal{B}} \| \frac{1}{|\mathcal{B}|} \sum_{\mathbf{i}\in\mathcal{B}} \mathbf{i} - \frac{1}{|\mathcal{B}|} \sum_{\mathbf{i}\in\mathcal{B}} \bar{\mathbf{i}} \|_2. \tag{11}$$

The overall objective in Disentangled Representation is:

$$\mathcal{L}^D = \beta_1 \mathcal{L}^{RD} + \beta_2 \mathcal{L}^{CT} + \beta_3 \mathcal{L}^{CA}, \tag{12}$$

where $\beta_1, \beta_2, \beta_3$ are the hyper-parameters.

3.4 Cross-scenario Augmentation and Contrastive Learning

The long-tail problem has always been a tough problem in E-commerce. We present two strategies to address this problem.

Cross-scenario data Augmentation As mentioned in Section 1, long-tail items differ in various search scenarios. The training signals of long-tail items in one scenario can be enriched by exploring user feedback in other scenarios, where they can receive more clicks. Therefore, we implement cross-scenario augmentation at the data level, using items with high similarity between scenarios to supplement the current scenario's training samples. The process proceeds as follows. On the current scenario, if a query-item pair (q, i) connects to an item i which receives less than three clicks, then we use similarity edge SE to find items i' that are highly similar to i in other scenarios. We select top-5 items i' based on their similarity. These top-5 items are constructed as the new query-item pairs(q, i').

We optimize the pair-wise supervised loss. For a query q or a trigger t, an item i^+ that has been clicked under the query (or is constructed by cross-scenario data augmentation as above) is treated as the positive sample. We randomly sample M other items i^- under the same category as negative samples. The InfoNCE loss[22] in supervised-learning is defined as follows:

$$\mathcal{L}^O = \sum_{\mathbf{x},\bar{\mathbf{i}}^+\in\mathcal{B}} -\log \frac{\exp(\mathrm{sim}(\mathbf{x}, \bar{\mathbf{i}}^+)/\tau)}{\sum_{\bar{\mathbf{i}}^-\in\mathcal{N}^-} \exp(\mathrm{sim}(\mathbf{x}, \bar{\mathbf{i}}^-)/\tau) + \exp(\mathrm{sim}(\mathbf{x}, \bar{\mathbf{i}}^+)/\tau)}, \tag{13}$$

where $\mathcal{N}^-$ represents the negative sample set. $\mathbf{x}$ is either the query embedding or the trigger embedding $\mathbf{q}, \mathbf{t}$ defined in Equation 4.

Contrastive Learning Recent study[26] suggests that GNNs have a poor performance in uniformity and make it difficult to retrieve long-tail items because popular items may cluster together. Inspired by this, we additionally utilize contrastive learning to address the long-tail problem. The motivation of contrastive learning is to shorten the distance between the anchor item and positive items while increasing the distance between the anchor item and negative items. By using contrastive learning, the distribution of samples in the feature space is more uniform. Formally, we define the contrastive loss:

$$\mathcal{L}^{CL} = \sum_{\bar{\mathbf{i}}\in\mathcal{B}} -\log \frac{\exp(\mathrm{sim}(\bar{\mathbf{i}}, \bar{\mathbf{i}}^+)/\tau)}{\sum_{\bar{\mathbf{i}}^-\in\mathcal{B}/\{\bar{\mathbf{i}}\}} \exp(\mathrm{sim}(\bar{\mathbf{i}}, \bar{\mathbf{i}}^-)/\tau) + \exp(\mathrm{sim}(\bar{\mathbf{i}}, \bar{\mathbf{i}}^+)/\tau)}, \tag{14}$$

where $\bar{\mathbf{i}}^+$ is the positive sample obtained by passing the anchor item embedding to a dropout layer. We use the rest items in a batch

as the negative samples. The keeping probability of the dropout layer is $\mathbf{p}_d$. sim $(\cdot, \cdot)$ denotes cosine similarity between two vectors.
Joint training Finally, the overall loss includes the objective in supervised-learning $\mathcal{L}^O$, the contrastive loss $\mathcal{L}^{CL}$, the regulation loss in disentangled representation $\mathcal{L}^D$:

$$\mathcal{L} = \sum_{i=1}^{N} (\mathcal{L}^O + \lambda_1 \mathcal{L}^D + \lambda_2 \mathcal{L}^{CL}), \tag{15}$$

where λ_1, λ_2 are the hyper-parameters.

4 OFFLINE EVALUATION

In this section, we conduct extensive experiments on offline datasets to study the following research questions:

*RQ*1 Did BOMGraph improve search performance on multiple scenarios?
*RQ*2 How well did each component in BOMGraph perform?
*RQ*3 Can BOMGraph alleviate the long-tail problem?
*RQ*4 How was BOMGraph affected by its hyper-parameter setting?

4.1 EXPERIMENTAL SETUP

Datasets For training the models, we collect search logs in mobile Taobao within a seven-day period on the visual search and textual search scenarios. Since traffic of the similar product search scenario is relatively lower, we gather search logs in 90 days on the similar product search scenario. Each record in the search log includes various attributes of the item, such as its ID, category, price, sales data, images, and title. During preprocessing, we eliminate query-item interactions that occur less than five times in a seven-day period for visual and textual search scenarios, and less than three times in a 90-day period for similar product search scenario. We select the actual queries on the next day of the training period as a testing set. For each query, we use the actual clicked items as the ground truth to evaluate the model performance. We report the key statistics of the three scenarios in Table 1, including the number of nodes and edges in the training set and the number of queries and actual clicks in the testing set. A similar product search scenario contains the largest amount of items, but it is the most sparse dataset.

Implementation The input product title and textual query embeddings are 50-dimensional vectors extracted from a word2vec model pretrained on a large scale E-commerce corpus. The input product image and Visual query embeddings are 512-dimensional vectors extracted from a metric learning model trained on an e-commerce platform visual search dataset. The node embedding size after GAT is 128. The FC layers size in the disentangled representation is 128. The number of independent counterparts M in Equation 6 is 3. In order to balance the magnitude of three loss terms in the disentangled representation module, we set the hyperparameters $\beta_1, \beta_2, \beta_3$ are 0.5,0.7,0.2 respectively. Except in Section 4.5, the dropout value to derive similar item embedding in contrastive learning is $\mathbf{p}_d = 0.7$ since it produces the best performance in hyper-parameter tuning. We set the optimization coefficients $\lambda_1 = 0.02, \lambda_2 = 0.0001$ to balance the value of each loss term. BOMGraph is trained using Adagrad optimizer. Batchsize is 256 for each scenario, and the epoch is 3. The shape of query and item embedding learned by BOMGraph is 128. Finally, we use ANN search [19] to recall the top-K relevant items in the same category with the query and evaluate the metrics. The purpose is to filter out items that belong to significantly different categories from the query and enhance performance. For product queries, the query category is already available. For keyword and visual queries, we use pretrained BERT[5] and ResNet[11] embeddings to predict the query's category.

Evaluation Metrics We adopt commonly used evaluation metrics [3, 9, 21, 37], such as Normalized Discounted Cumulative Gain (NDCG)@100/200, Hit Ratio (HR)@100/200, and Mean Reciprocal Rank (MRR)@100/200. The higher the metric value is, the more accurate the returned results are.

4.2 Comparative study

Baselines We compare BOMGraph to several industry-scale GNNs for *single scenario graph*. (1) GraphSage [10]generates embeddings by sampling and aggregating features from a node's local neighborhood instead of training individual embeddings for each node. (2) AdaptiveGCN [15] proposes a novel spectral graph convolution network that feeds on diverse graph structures and customizes spectral filter that combines neighborhood topological features. (3) LasGNN [3] is a metapath-based graph neural network, which samples the neighbors layer-wise along the intra-scenario metapath and then aggregates messages on the constructed subgraph. We also compare BOMGraph to GNNs that can be implemented on large-scale heterogeneous graph of *multiple scenarios*. (4) MSGraph[1] is a multi-scenario graph neural network that samples neighbors layer-wise only along the intra-scenario metapath, and then aggregates messages within the constructed intra-subgraph from the large multi-scenario graph.

We report the results of different methods on each search scenario in Table 2. The GraphSage, AdaptiveGCN, LasGNN baselines are trained and tested on the same scenario because they are incapable of modeling multi-scenario problems. The MSGraph is trained with all three scenarios to obtain maximal performance, and it is tested on different single scenarios. BOMGraph is trained with different combinations of scenarios to evaluate the impact of adding different scenarios. For example, BOMGraph-SK suggests that BOMGraph is trained with a similar product search dataset and textual search. Note that since the similar product search scenario is the largest dataset, we always incorporate it in the joint training.

From Table 2, we have the following observations. (1) *BOMGraph-SKV consistently achieves best results on all search scenarios*, which demonstrates the superiority of joint learning multi-scenario search. The proposed BOMGraph-SKV boosts the HR@100, HR@200, MRR@100, MRR@200, NDCG@100, NDCG@200 by 11.6%, 9.3%, 14.4%, 14.3%, 13.4%, 12.4% respectively, in similar product search scenario, compared with the best baseline LasGNN. In the textual search scenario, BOMGraph-SKV boosts the HR@100, MRR@100, NDCG@100 by 5.8%, 16.75%, 12.1%, respectively. In the visual search scenario, BOMGraph-SKV boosts the HR@100, MRR@100, NDCG@100 by 23.2%, 27.7%, 24.8% respectively. (2) We observe that *combining more search scenarios improves ranking performance of BOMGraph*. For example, BOMGraph-SKV, which combines three scenarios, outperforms model variants that combine two scenarios

[3]https://github.com/alibaba/euler

Table 1: Dataset statistics

	Training					Testing	
Scenario	#Queries	#Items	#Click Edges	#Co-Occurrence Edges	#Similarity Edges	#Queries	#Clicks
Similar product search	41,091,137	152,947,641	183,516,724	62,617,406	82,390,475	618,371	646,935
Textual search	142,715,949	46,562,213	1,015,194,641	28,569,906	42,882,795	6,040,132	190,004,120
Visual search	199,804,270	74,925,225	N/A	44,392,019	65,311,323	4,954,787	14,970,884

Table 2: Search performance of different models on each search scenario. The best performance is shown in bold font.

Scenario	Model	HR@100	HR@200	MRR@100	MRR@200	NDCG@100	NDCG@200
Similar product search (S)	GraphSage	0.4359	0.5134	0.0777	0.0782	0.1465	0.1573
	AdaptiveGCN	0.4434	0.5138	0.825	0.0830	0.1531	0.1630
	LasGNN	0.4805	0.5466	0.1103	0.1107	0.1831	0.1924
	MSGraph-SKV	0.4713	0.5382	0.1003	0.1005	0.1715	0.1809
	BOMGraph-SK	0.5314	0.5938	0.1231	0.1235	0.2042	0.2129
	BOMGraph-SV	0.5234	0.5872	0.1207	0.1212	0.2005	0.2094
	BOMGraph-SKV	**0.5363**	**0.5976**	**0.1262**	**0.1266**	**0.2077**	**0.2163**
Textual search (K)	GraphSage	0.3967	0.4887	0.0739	0.0747	0.1279	0.1401
	AdaptiveGCN	0.4027	0.4907	0.0742	0.0749	0.1357	0.1481
	LasGNN	0.4141	0.5019	0.0746	0.0791	0.1359	0.1479
	MSGraph-SKV	0.4089	0.4979	0.0721	0.0776	0.1332	0.1457
	BOMGraph-SK	0.4344	0.5212	0.0851	0.0858	0.1513	0.1634
	BOMGraph-SKV	**0.4382**	**0.5237**	**0.0871**	**0.0877**	**0.1538**	**0.1657**
Visual search (V)	GraphSage	0.2633	0.3181	0.0511	0.0517	0.0897	0.0913
	AdaptiveGCN	0.2721	0.3275	0.0521	0.0528	0.0942	0.0964
	LasGNN	0.2906	0.3434	0.0574	0.0577	0.1021	0.1095
	MSGraph-SKV	0.2887	0.3409	0.0553	0.0561	0.1001	0.1079
	BOMGraph-SV	0.3526	0.4066	0.0691	0.0695	0.1246	0.1331
	BOMGraph-SKV	**0.3581**	**0.4189**	**0.0733**	**0.0737**	**0.1275**	**0.1351**

Table 3: Ablation study of different components, CL represents Contrastive Learning with cross-scenario augmentation, DR represents Disentangled Representation, and CS represents Cross-Scenario graph encoder.

Scenario	Model	HR@100	HR@200	MRR@100	MRR@200	NDCG@100	NDCG@200
Similar product search	**BOMGraph**-SK	**0.5314**	**0.5938**	**0.1231**	**0.1235**	**0.2042**	**0.2129**
	w/o CL	0.5110	0.5742	0.1184	0.1189	0.1963	0.2052
	w/o CL,DR	0.4878	0.5532	0.1117	0.1121	0.1857	0.1948
	w/o CL,DR,CS	0.4738	0.5399	0.1010	0.1015	0.1743	0.1835
Keyword search	**BOMGraph**-SK	**0.4344**	**0.5212**	**0.0851**	**0.0858**	**0.1513**	**0.1634**
	w/o CL	0.4314	0.5197	0.0824	0.0828	0.1481	0.1614
	w/o CL,DR	0.4265	0.5185	0.0797	0.0813	0.1446	0.1579
	w/o CL,DR,CS	0.4123	0.5017	0.0743	0.0790	0.1356	0.1472
Visual search	**BOMGraph**-SV	**0.3526**	**0.4066**	**0.0691**	**0.0695**	**0.1246**	**0.1331**
	w/o CL	0.3310	0.3879	0.0651	0.0665	0.1157	0.1247
	w/o CL,DR	0.3101	0.3681	0.0596	0.0601	0.1062	0.1134
	w/o CL,DR,CS	0.2906	0.3434	0.0574	0.0577	0.1021	0.1095

on each dataset. However, even with two scenarios, the proposed BOMGraph significantly outperforms LasGNN, the best competitor. This observation verifies our assumption that performance on each scenario can be boosted by jointly learning multi-scenario in a unified framework. (3) In addition, we observe that LasGNN with single scenario outperforms MSGraph-SKV which combines three scenarios. This shows that *simply fusing multiple scenarios is*

Table 4: Ablation results of Disentangled Representation, base: Multi-scenario graph encoder, RD: Representation Dissociation, CA: Centroid Alignment, CT: Common knowledge Transfer.

Model	Similar product search					
	HR@100	HR@200	MRR@100	MRR@200	NDCG@100	NDCG@200
base+RD+CT+CA	**0.5110**	**0.5742**	**0.1184**	**0.1189**	**0.1963**	**0.2052**
base+RD+CT	0.5092	0.5726	0.1184	0.1188	0.1958	0.2046
base+RD	0.5010	0.5651	0.1164	0.1169	0.1924	0.2014
base	0.4878	0.5532	0.1117	0.1121	0.1857	0.1948

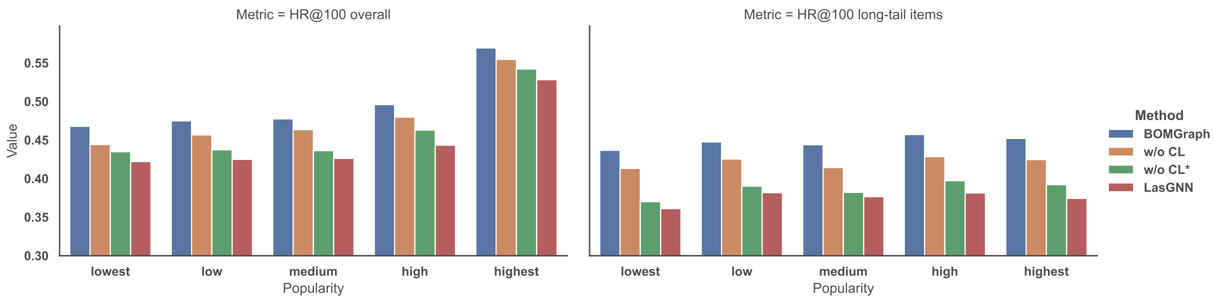

Figure 4: Performance of different models on all items and long-tail items, in a similar product scenario, with respect to different query bins. CL: Contrastive Learning, CL*: Contrastive Learning with cross-scenario augmentation.

Table 5: Performance obtained by additionally keeping probability p_d of dropout layer in Contrastive Learning module.

Model	p_d	**Similar product search**		
		HR@100	MRR@100	NDCG@100
BOMGraph	0.9	0.5329	0.1242	0.2054
BOMGraph	0.8	0.5339	0.1253	0.2065
BOMGraph	0.7	**0.5363**	**0.1262**	**0.2077**
BOMGraph	0.6	0.5351	0.1257	0.2071
BOMGraph	0.5	0.5313	0.1232	0.2042
BOMGraph	0.4	0.5310	0.1236	0.2046
BOMGraph	0.3	0.5300	0.1234	0.2041

not enough and does not necessarily yield better results than single-scenario learning. Meanwhile, the performance of LasGNN is better than GraphSage. The underlying reason is that subgraphs sampled based on metapath can aggregate node information more efficiently than randomly sampled subgraphs in E-commerce search.

4.3 Ablation Study

4.3.1 Impacts of different modules in BOMGraph. To further investigate the effectiveness of the modules we proposed, we conduct an ablation study with model variants that remove different components in BOMGraph, i.e., Contrastive Learning with cross-scenario Augmentation (CL), Disentangled representation learning (DR), and Cross-Scenario graph encoder (CS). We observe the effectiveness of each component from the results in Table 3. For example, in "similar product search", removing the Contrastive Learning with cross-scenario Augmentation (i.e., "w/o CL"), the performance is degraded by 0.0204, 0.0116, in terms of HR@100, NDCG@100, respectively. Further removing the Disentangled representation learning (i.e., "w/o CL,DR"), the performance is degraded by 0.0232, 0.0069. Further removing the Cross-Scenario graph encoder (i.e., "w/o CL,DR,CS"), BOMGraph degrades to LasGNN, and the performance is degraded by 0.0014, 0.0114. This proves the *necessity of each component in capturing the cross-scenario information.*

Comparing the results in Table 3 over scenarios, we observe that removing each component causes a more significant performance decline in the visual search scenario and the similar product search scenario. These two scenarios match products to more complicated queries (i.e., multi-modal queries), and the datasets are significantly more sparse than the textual search scenario. The results imply that *the proposed components are potentially more beneficial for scenarios with low traffic and complex queries.*

4.3.2 Impacts of Disentangled Representation Steps. To study in detail the proposed Disentangled Representation learning, we conducted model variants on the similar product search scenario. The baseline is BOMGraph-SK with only a multi-scenario graph encoder without contrastive learning. We incrementally add steps of Disentangled Representation learning to the baseline, i.e., RD: Representation Dissociation, CA: Centroid Alignment, and CT: Common knowledge Transfer. From the results in Table 4, we have observed that best performance is achieved by adding three steps of the disentangle representation learning. This demonstrates the effectiveness of disentangled representation. In general, adding each step leads to positive impacts. Comparing over the steps, Representation Dissociation yielded a large performance boost on all evaluation metrics, indicating that we were able to learn more nuanced features for representations in different scenarios. Centroid

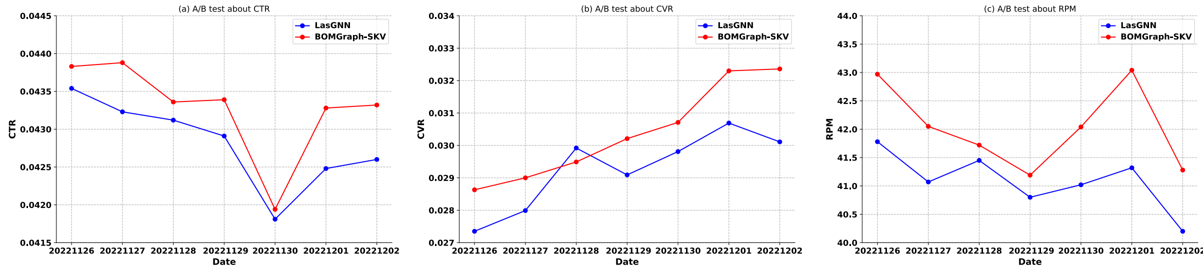

Figure 5: Online A/B test results about CTR(a), CVR(b) and RPM(c)

Table 6: The performance of A/B tests conducted online over a seven-day period.

	CTR	CVR	RPM
LasGNN	0.0433	0.0304	40.43
BOMGraph-SKV	0.0437	0.0313	41.46
	+0.92%	+2.89%	+2.55%

Alignment did not significantly improve the MRR metric, but it did lead to a notable increase in the Hit Ratio. This is a welcomed improvement, because instead of ranking one ground-truth click higher (i.e., MRR), E-commerce search aims to rank all ground-truth clicks higher (i.e., Hit Ratio).

4.4 Long-tail performance

To investigate the effect of cross-scenario Augmentation and contrastive learning on long-tail items, we compute the average performance of different models on all items, and the performance on long-tail items. By long-tail items, we refer to items which have less than three clicks. The experiment is conducted on similar product search scenario. We also divide queries (i.e., trigger items) into five bins based on their popularity, i.e., from lowest to highest popularity, and all the bins have equal width.

According to Figure 4, (1) search performance is significantly affected by the popularity of items and queries. In general, performance is worse on long-tail items and low-popularity queries. (2) BOMGraph achieves satisfying $HR@100$ (> 0.4) on all items and long-tail items for queries of different popularity, which proves that BOMGraph is more robust regardless of the property of queries and items. (3) By comparing the performance of BOMGraph w/o CL* (CL*: Contrastive Learning with cross-scenario augmentation) and BOMGraph w/o CL (CL: Contrastive Learning), we can see that cross-scenario augmentation is important for long-tail items, i.e., w/o CL* causes larger performance drop on long-tail items than on overall items. This verifies our assumption in Section 1 that long-tail items are diverse in different scenarios, and the proposed cross-scenario augmentation can combine the information from multiple scenarios to better handle long-tail items.

4.5 Impacts of of Hyper-parameters

Finally, we explore the impact of dropout probability in the Contrastive Learning module defined in Section 3.4. We use the anchor item after dropout as a positive example of the anchor item. From the results (Table 5), we find that the lower the keep probability $\mathbf{p}_d$ is, the worse the model performs. An obvious reason is that the item embedding that loses too much information is less effective as a positive example. Besides, we find that the keeping probability of dropout layer $\mathbf{p}_d = 0.7$ yields the best result. We argue that the reason is, discarding 30% information of the item embedding can give the model a challenging, positive sample to learn meaningful features while keeping 70% of the original item embedding provides useful clues for the model to learn more efficiently.

5 ONLINE PERFORMANCE

We conducted online A/B tests to compare the performance of our proposed model, BOMGraph, to that of the previous industrial solution, i.e., single-scenario model LasGNN. During the seven-day experiment, we only modified the candidate generation step by using different models to recall the top 200 items for each product query on the similar product search scenario, while keeping all other ranking steps unchanged. We compared the online real click-through rate (CTR), conversion rate (CVR) and Revenue Per Mille (RPM) to determine whether the items recalled by BOMGraph were more likely to be clicked on and purchased by users. As shown in Figure 5, in similar product search scenarios, our proposed BOMGraph-SKV consistently outperforms LasGNN in terms of CTR and RPM. In terms of CVR, it only falls behind LasGNN for one day. This may be because while our model is able to increase the click-through rate every day, it doesn't necessarily always lead to an increase in conversion rate. However, overall, our CVR still shows improvement. As shown in Table 6, in the seven-day period, the proposed BOMGraph-SKV increased the CTR, CVR and RPM by 0.92%, 2.89% and 2.55%, respectively, compared with LasGNN.

6 CONCLUSION

In this paper, we propose BOMGraph, which is a novel graph neural network that jointly optimize large-scale multi-scenario E-commerce search. BOMGraph has been deployed in production by Alibaba's E-commerce search advertising platform. It consists of novel techniques to capture heterogeneous information flow across scenarios, learn scenario-robust item representations, and address the long-tail problems. We conduct extensive offline experiments on billion-scale real production data and online A/B test to demonstrate that BOMGraph outperforms state-of-the-art competitors.

REFERENCES

[1] Jiangxia Cao, Xixun Lin, Xin Cong, Jing Ya, Tingwen Liu, and Bin Wang. 2022. DisenCDR: Learning Disentangled Representations for Cross-Domain Recommendation. In *SIGIR*. 267–277.

[2] Jie Chen, Tengfei Ma, and Cao Xiao. 2018. FastGCN: Fast Learning with Graph Convolutional Networks via Importance Sampling. In *ICLR (Poster)*.

[3] Yuting Chen, Yanshi Wang, Yabo Ni, Anxiang Zeng, and Lanfen Lin. 2020. Scenario-aware and Mutual-based approach for Multi-scenario Recommendation in E-Commerce. In *ICDM (Workshops)*. 127–135.

[4] Zhihong Chen, Rong Xiao, Chenliang Li, Gangfeng Ye, Haochuan Sun, and Hongbo Deng. 2020. Esam: Discriminative domain adaptation with non-displayed items to improve long-tail performance. In *SIGIR*. 579–588.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). 4171–4186.

[6] Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, and Yongliang Li. 2019. Metapath-Guided Heterogeneous Graph Neural Network for Intent Recommendation. 2478–2486.

[7] Jun Feng, Heng Li, Minlie Huang, Shichen Liu, Wenwu Ou, Zhirong Wang, and Xiaoyan Zhu. 2018. Learning to Collaborate: Multi-Scenario Ranking via Multi-Agent Reinforcement Learning. In *Proceedings of the 2018 World Wide Web Conference*. 1939–1948.

[8] Yulong Gu, Wentian Bao, Dan Ou, Xiang Li, Baoliang Cui, Biyu Ma, Haikuan Huang, Qingwen Liu, and Xiaoyi Zeng. 2021. Self-Supervised Learning on Users' Spontaneous Behaviors for Multi-Scenario Ranking in E-commerce. In *CIKM*. 3828–3837.

[9] Yangyang Guo, Zhiyong Cheng, Liqiang Nie, Xin-Shun Xu, and Mohan Kankanhalli. 2018. Multi-modal preference modeling for product search. In *Proceedings of the 26th ACM international conference on Multimedia*. 1865–1873.

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

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[12] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. 2018. Adversarial Personalized Ranking for Recommendation. In *SIGIR*. 355–364.

[13] SeongKu Kang, Junyoung Hwang, Dongha Lee, and Hwanjo Yu. 2019. Semi-Supervised Learning for Cross-Domain Recommendation to Cold-Start Users. In *CIKM*. 1563–1572.

[14] Pengcheng Li, Runze Li, Qing Da, Anxiang Zeng, and Lijun Zhang. 2020. Improving Multi-Scenario Learning to Rank in E-commerce by Exploiting Task Relationships in the Label Space. In *CIKM*. 2605–2612.

[15] Ruoyu Li, Sheng Wang, Feiyun Zhu, and Junzhou Huang. 2018. Adaptive Graph Convolutional Neural Networks. In *AAAI*. 3546–3553.

[16] Meng Liu, Jianjun Li, Guohui Li, and Peng Pan. 2020. Cross Domain Recommendation via Bi-directional Transfer Graph Collaborative Filtering Networks. In *CIKM*. 885–894.

[17] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. In *KDD*. 1930–1939.

[18] Jianxin Ma, Chang Zhou, Hongxia Yang, Peng Cui, Xin Wang, and Wenwu Zhu. 2020. Disentangled Self-Supervision in Sequential Recommenders. In *KDD*. 483–491.

[19] Yu A Malkov and Dmitry A Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence* 42, 4 (2018), 824–836.

[20] Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. 2017. Cross-Domain Recommendation: An Embedding and Mapping Approach. In *IJCAI*. 2464–2470.

[21] Xichuan Niu, Bofang Li, Chenliang Li, Jun Tan, Rong Xiao, and Hongbo Deng. 2021. Heterogeneous Graph Augmented Multi-Scenario Sharing Recommendation with Tree-Guided Expert Networks. In *WSDM*. 1038–1046.

[22] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint* (2018). https://arxiv.org/abs/1807.03748

[23] Qijie Shen, Wanjie Tao, Jing Zhang, Hong Wen, Zulong Chen, and Quan Lu. 2021. SAR-Net: A Scenario-Aware Ranking Network for Personalized Fair Recommendation in Hundreds of Travel Scenarios. In *CIKM*. 4094–4103.

[24] Xiang-Rong Sheng, Liqin Zhao, Guorui Zhou, Xinyao Ding, Binding Dai, Qiang Luo, Siran Yang, Jingshan Lv, Chi Zhang, Hongbo Deng, and Xiaoqiang Zhu. 2021. One Model to Serve All: Star Topology Adaptive Recommender for Multi-Domain CTR Prediction. In *CIKM*. 4104–4113.

[25] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *stat* 1050 (2017), 20.

[26] Chenyang Wang, Yuanqing Yu, Weizhi Ma, Min Zhang, Chong Chen, Yiqun Liu, and Shaoping Ma. 2022. Towards Representation Alignment and Uniformity in Collaborative Filtering. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1816–1825.

[27] Menghan Wang, Yujie Lin, Guli Lin, Keping Yang, and Xiao-Ming Wu. 2020. M2GRL: A Multi-task Multi-view Graph Representation Learning Framework for Web-scale Recommender Systems. In *KDD*. 2349–2358.

[28] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. 2020. Disentangled Graph Collaborative Filtering. In *SIGIR*. 1001–1010.

[29] Yichao Wang, Huifeng Guo, Bo Chen, Weiwen Liu, Zhirong Liu, Qi Zhang, Zhicheng He, Hongkun Zheng, Weiwei Yao, Muyu Zhang, Zhenhua Dong, and Ruiming Tang. 2022. CausalInt: Causal Inspired Intervention for Multi-Scenario Recommendation. In *KDD*. 4090–4099.

[30] Max Welling and Thomas N Kipf. 2016. Semi-supervised classification with graph convolutional networks. In *J. International Conference on Learning Representations (ICLR 2017)*.

[31] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised Graph Learning for Recommendation. In *SIGIR*. 726–735.

[32] Tao Wu, Ellie Ka In Chio, Heng-Tze Cheng, Yu Du, Steffen Rendle, Dima Kuzmin, Ritesh Agarwal, Li Zhang, John R. Anderson, Sarvjeet Singh, Tushar Chandra, Ed H. Chi, Wen Li, Ankit Kumar, Xiang Ma, Alex Soares, Nitin Jindal, and Pei Cao. 2020. Zero-Shot Heterogeneous Transfer Learning from Recommender Systems to Cold-Start Search Retrieval. In *CIKM*. 2821–2828.

[33] Ruobing Xie, Qi Liu, Liangdong Wang, Shukai Liu, Bo Zhang, and Leyu Lin. 2022. Contrastive Cross-domain Recommendation in Matching. In *KDD*. 4226–4236.

[34] Chenxiao Yang, Junwei Pan, Xiaofeng Gao, Tingyu Jiang, Dapeng Liu, and Guihai Chen. 2022. Cross-Task Knowledge Distillation in Multi-Task Recommendation. In *AAAI*. 4318–4326.

[35] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. 2022. Are Graph Augmentations Necessary?: Simple Graph Contrastive Learning for Recommendation. In *SIGIR*. 1294–1303.

[36] Jiejie Zhao, Bowen Du, Leilei Sun, Fuzhen Zhuang, Weifeng Lv, and Hui Xiong. 2019. Multiple Relational Attention Network for Multi-task Learning. In *KDD*. 1123–1131.

[37] Xinyu Zou, Zhi Hu, Yiming Zhao, Xuchu Ding, Zhongyi Liu, Chenliang Li, and Aixin Sun. 2022. Automatic Expert Selection for Multi-Scenario and Multi-Task Search. In *SIGIR*. 1535–1544.