Disentangled Interest importance aware Knowledge Graph Neural Network for Fund Recommendation

Ke Tu*
Ant Group
tuke.tk@antgroup.com

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

Zhengwei Wu
Ant Group
zejun.wzw@antfin.com

Zhiqiang Zhang
Ant Group
lingyao.zzq@antfin.com

Zhongyi Liu
Ant Group
zhongyi.lzy@alibaba-inc.com

Yiming Zhao
Ant Group
desert.zym@antgroup.com

Le Wu
Hefei University of Technology
lewu.ustc@gmail.com

Jun Zhou†
Ant Group
jun.zhoujun@antgroup.com

Guannan Zhang
Ant Group
zgn138592@antgroup.com

ABSTRACT

At present, people are gradually becoming aware of financial management and thus fund recommendation attracts more and more attention to help them find suitable funds quickly. As a user usually takes many factors (e.g., fund theme, fund manager) into account when investing a fund and the fund usually consists of a substantial collection of investments, effectively modeling multi-interest representations is more crucial for personalized fund recommendation than the traditional goods recommendation. However, existing multi-interest methods are largely sub-optimal for fund recommendation, since they ignore financial domain knowledge and diverse fund investment intentions. In this work, we propose a Disentangled Interest importance aware Knowledge Graph Neural Network (DIKGNN) for personalized fund recommendation on FinTech platforms. In particular, we restrict the multiple intent spaces by introducing the attribute nodes from the fund knowledge graph as the minimum intent modeling unit to utilize financial domain knowledge and provide interpretability. In the intent space, we define disentangled intent representations, equipped with intent importance distributions to describe the diverse fund investment intentions. Then we design a new neighbor aggregation mechanism with the learned intent importance distribution upon the interaction graph and knowledge graph to collect multi-intent information. Furthermore, we leverage micro independence and macro balance constraints on the representations and distributions respectively to encourage intent independence and diversity. The extensive experiments on public recommendation benchmarks demonstrate that DIKGNN can achieve substantial improvement over state-of-the-art methods. Our proposed model is also evaluated over one real-world industrial fund dataset from a FinTech platform and has been deployed online.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

fund recommendation, disentangled, graph neural network, knowledge graph

ACM Reference Format:

Ke Tu, Wei Qu[1], Zhengwei Wu, Zhiqiang Zhang, Zhongyi Liu, Yiming Zhao, Le Wu, Jun Zhou, and Guannan Zhang. 2023. Disentangled Interest importance aware Knowledge Graph Neural Network for Fund Recommendation. 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/10.1145/3583780.3614846

1 INTRODUCTION

In the past decade, financial technology (FinTech) has already greatly increased the convenience of financial services and given everyone easy access to financial markets, especially in the asset management industry. Empowered by technological advances such as artificial intelligence and mobile computing, FinTech platforms (e.g., Alipay[1] operated by Ant Financial, Tencent LiCaiTong[2] operated by Tencent, Robinhood[3], etc.) which sell funds via mobile apps, have emerged as a new sales mode to reach a wide variety of customers. Due to its quick and flexible service, this new mobile channel for fund distribution is fast increasing its market share compared to traditional channels. By end of 2018, about one-third of the sales of funds has taken place on FinTech platforms [10].

Compared with general recommendations for e-commerce, designing a personalized recommendation strategy for funds on FinTech platforms faces unique challenges due to complex finance domain knowledge and diverse fund investment intentions. First, without the assistance of finance domain knowledge, it is challenging to fully comprehend the relationship between funds and even

*These authors contributed equally to this work.
†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/10.1145/3583780.3614846

[1]https://www.alipay.com
[2]https://www.tenpay.com
[3]https://robinhood.com

more challenging to make suitable recommendations. For example, as shown in Figure 1, although "GF Growth Selective Hybrid C" and "Fullgoal Precision Healthcare Flexible Allocation Hybrid" appear to be unrelated at first glance, they actually have the same top holding stock and both belong to the healthcare sector. Then the purchase of the first fund implies a potential interest in the second one. The second challenge is the diverse fund investment intentions. Unlike straightforward purchasing intention of daily life goods, a user usually takes many factors into account when investing a fund, including seeking high expected return, balancing potential risk, keeping optimistic about a specific industry, or simply trusting a fund manager, etc. Therefore effectively modeling multi-interest representations is more crucial for personalized recommendations than traditional goods recommendations. For instance, as illustrated in Figure 1, the funds existing in Alice's historical behaviors cover a wide variety of investment sectors, fund types, asset management firms, etc. After analyzing Alice's investment intentions through her historical behaviors with fund domain knowledge, we find that Alice has two main interests including hybrid healthcare sector related funds and ETF link funds in the semiconductor sector.

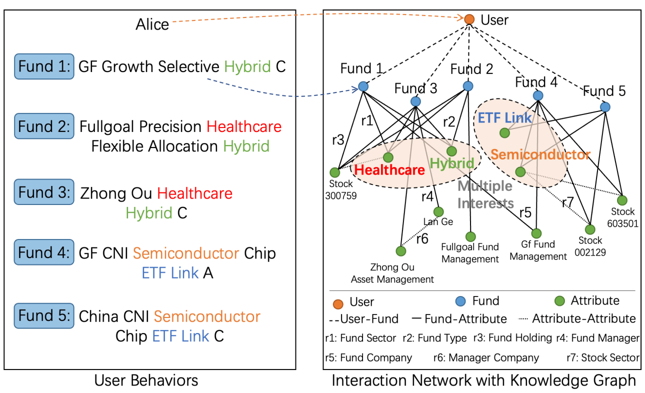

Figure 1: An illustration of multi-faceted interests behind historical interactions with funds on the FinTech platform.

Some recent works [32, 35, 38] have attempted to leverage knowledge graph techniques to translate complex domain knowledge into a structured representation that is accessible to both humans and machines, while combining multi-interest learning for the recommendation. Inspired by disentangled representation learning [19] in the field of image, some GNN-based methods, such as DisenGCN [20] and DisenKGAT [44], design a graph disentangling module equipped with neighbor routing and embedding propagation mechanisms to characterize users' diverse interests. The KG-based methods, i.e., KGIN [40] and KTUP [2], seek to leverage the auxiliary semantic information of knowledge graph to model multi-interests.

Despite their effectiveness, we argue that these methods neglect the following considerations: (1) **Fine-grained Knowledge-aware Representation**. In these studies, they fall short of capturing fine-grained multi-interest representations using the knowledge graph to its full potential. GNN-based models [20, 41] leverage dynamic routing mechanisms to learn implicitly the representation of interests which are opaque to deeper understanding. Although KG-based multi-interest models [2, 40] leverage the auxiliary knowledge of items, they lack the ability of modeling complex fund interaction behaviors (for example, a user may have multiple intents to interact with a fund) because they use the relation as the minimum interest modeling unit. (2) **Multi-interest Importance Distribution**. The multi-interest based methods [20, 40, 41] mainly focus on extracting non-overlapping interests while ignoring differences in interest importance. Therefore, they fail to capture the diverse fund investment intentions accurately.

In this paper, we propose a new model, *Disentangled Interest Importance Aware Knowledge Graph Neural Network* (DIKGNN) to solve the foregoing issues. (1) We design **Knowledge-aware Disentangled Intent Modeling** to learn disentangled intent representation in semantic intent spaces constrained by domain knowledge. Specifically, we utilize the fund knowledge graph to introduce finance domain knowledge and regard the attribute nodes (the nodes apart from fund nodes) from the fund knowledge graph as the minimum intent modeling unit to limit the distribution of attribute nodes to one-hot vectors. In other words, each intent can be regarded as a hidden cluster of attribute nodes, thereby effectively shaping the intent space while enhancing learning performance and model explainability. Upon these intent spaces, we disentangle the users and funds into multiple channels of independent representations to represent the diverse fund investment intentions. Then an intent importance distribution is generated by the disentangled representations to describe the preference for different intents. (2) We carefully design **Intent Importance-specific Aggregation** to aggregate neighbor information from separate intent spaces while taking intent importance distribution into account. Considering the heterogeneity of the interaction network and fund knowledge graph, we correspondingly conduct the users' behavior neighbor aggregation and the funds' knowledge neighbor aggregation to collect the multi-interest representation and use the generated intent importance distribution to discern aggregation weights. (3) We propose an **Independence and Balance Constraint**. Specifically, A micro independence constraint on disentangled representations is designed to encourage intent independence and a macro balance constraint on the intent importance distribution is used to increase intent diversity. The final similarity score is obtained by fusing all channel representations with the intent importance.

To summarize, the contributions of this paper are as follows,

- To the best of our knowledge, this is the first work tailored to address the multi-interest problem in fund recommendation, where traditional approaches are not generalizable due to the unique characteristics including complex finance domain knowledge and diverse investment intentions in fund field.
- We propose a novel Disentangled Interest Importance Aware Knowledge Graph Neural Network (DIKGNN). It contains a Knowledge-aware Disentangled Intent module, an Intent Importance-specific Aggregation module, and an Independence and Balance Constraint module, which provides better model interpretability and more granular multi-interest representation.
- We conduct extensive experiments on three public datasets and one large-scale dataset collected from real-world FinTech platforms. The experimental results show significant performance improvements compared with the state-of-the-art methods. Further analysis is provided to demonstrate

the robustness and effectiveness of DIKGNN. The model has been deployed online to serve fund recommendation in real-world FinTech platforms.

2 RELATED WORK

2.1 General Recommendation

The general recommendation models could be roughly divided into three types: shallow models, neural network-based models, and GNN-based models. The typical examples of *shallow models* are collaborative filtering [26], matrix factorization [14], and factorization machines [23]. However, these methods encounter critical challenges such as complex user behaviors or data input. To address it, *neural network-based models* are proposed inspired by the success of deep learning in computer vision and natural language processing. For example, Neural Collaborative Filtering [8], NFM [8] and Deep Matrix Factorization Models [45] respectively integrate their corresponding shallow models with multi-layer perceptron to represent the interaction between users and items. Another line of work tries to apply *GNN-based models* [9, 39, 46] for recommendation. Specifically, graph neural networks recursively utilize embedding propagation and neighbor aggregation mechanisms to access high-order neighbors' information, rather than only the first-order neighbors' as the traditional methods do.

2.2 Knowledge Graph Based Recommendation

Existing KG-based recommendation approaches can be roughly grouped into three categories: embedding-based, path-based, and propagation-based methods. For *embedding-based methods* [2, 34, 48], they usually first embed entities and relations in KG via knowledge graph embedding methods (e.g., TransR [18]), and then add them as auxiliary information of items into the recommender model. For example, CKE [48] utilizes TransR and feeds the learned item embeddings of the knowledge graph into matrix factorization. However, the embedding-based methods focus more on item representation learning in KG while ignoring the high-order user-item interactions for recommendations. Thus, they cannot capture the complex dependencies of user-item relations for user preference learning. *The path-based methods* [12, 29, 42, 47] aim to find different semantic paths to connect users and items via entities in KG to guide the recommendation process. For instance, KPRN [42] generates path representations by composing the semantics of both entities and relations and integrating them into the recommendation. Nevertheless, they highly depend on handcraft meta-path construction, which relies on domain knowledge and human efforts, and aggregating information along different meta-paths is very time-consuming. *The propagation-based methods* [35–37] have attracted increasing attention in recent years, where the propagation process is performed iteratively to extract auxiliary information from KG for user preference learning. KGAT [38] first applies knowledge graph embedding methods to obtain entity representations and then propagate the representations over the user-item interaction network and knowledge graph to collect the high-order information. KCAN [32] proposes conditional attention on the knowledge graph to capture the target-specific preference of users. Despite successful applications, these methods ignore discrimination of various interests by aggregating multifaceted preferences into a single vector.

2.3 Multi-interest Recommendation

Some recent works have explored fund recommendation scenarios, such as [11] and [4]. However, these studies do not consider the multi-interests of users when purchasing funds. To identify the diverse interests of users, recently many efforts have been made on multi-interest recommendation [6, 17, 31] which can be mainly divided into two lines. The first line of work learns multi-interest representations from behavior collaborative signals. For example, MIND [17] takes the first attempt to leverage the dynamic routing mechanism [25] to represent users with multiple interests. Similarly, DisenGCN [20] applies a dynamic routing mechanism on the neighbor aggregation to model multiple interests. ComiRec [3] designs multi-interest module to capture multiple interests from user behavior sequences and an aggregation module with a controllable factor to balance the recommendation accuracy and diversity. All these works capture intents implicitly from the users' multiple behaviors, it is hard to identify the semantics of each intent explicitly. Another line of work focuses on constructing multi-interest representations from the knowledge graphs. DisenKGAT [44] extends DisenGCN into a KG-based setting and introduces knowledge attention into the disentangled GNN. In order to enhance interpretability, KGIN [40] models each intent as an attentive combination of KG relations to refine high-level concepts of user intents. We argue that the existing approaches lack the ability to model complex fund interaction behaviors because they use the relation as the minimum interest modeling unit. Our method designs a Knowledge-aware Disentangled Intent Module to capture entity-level intentions.

3 PRELIMINARY

We first introduce the notations and give a formal definition of our task. In the fund recommendation system, users have implicit feedback, e.g. click or purchase. Let $\mathcal{G} = (\mathcal{V} = \{\mathcal{U}, \mathcal{I}\}, O = \{(u,i)|u \in \mathcal{U}, i \in \mathcal{I}\})$ be the interaction graph, where $\mathcal{U}$ and $\mathcal{I}$ are the set of users and funds. And O is a set of the observed interactions in which a sample $(u,i) \in O$ indicates a user u has interacted with a fund i. Compared with the traditional recommendation scenarios, there is an additional knowledge graph to introduce financial domain knowledge in our task. Let $\mathcal{G}_k = \{(h,r,t)|h \in \mathcal{E}, r \in \mathcal{R}, t \in \mathcal{E}\}$ be the knowledge graph, where h, r and t denote the head node, relation and tail node, $\mathcal{E}$ is the node set in the knowledge graph, and $\mathcal{R}$ denotes the relation set in the knowledge graph. The $\mathcal{E} = \{\mathcal{I}, \mathcal{C}\}$ is the node set that contains the set of funds $\mathcal{I}$ and the sets of attribute nodes $\mathcal{C}$. The nodes in the set $\mathcal{C}$, such as fund sector and fund manager, usually are the attribute information of funds.

Fund Recommendation. Given a user-fund interaction graph $\mathcal{G} = (\mathcal{V} = \{\mathcal{U}, \mathcal{I}\}, O = \{(u,i)|u \in \mathcal{U}, i \in \mathcal{I}\})$ and a fund knowledge graph $\mathcal{G}_k = \{(h,r,t)|h \in \mathcal{E}, r \in \mathcal{R}, t \in \mathcal{E}\}$, our task of fund recommendation is to learn a score function $\Phi(u,i) : \mathcal{U} \times \mathcal{I} \rightarrow \mathbb{R}$ that assigns how likely a user will interact with a fund in the future.

4 METHODOLOGY

In this section, we will introduce our proposed model *Disentangled Interest importance aware Knowledge Graph Neural Network*

(DIKGNN). The framework is shown in Figure 2. It consists of three key components: (1) Knowledge-aware Disentangled Intent Modeling. We first apply multiple channels of disentangled representations with an intent importance distribution to describe the diversity of intent. Especially, we introduce the fund knowledge graph to utilize financial domain knowledge to limit the intent space. (2) Intent Importance-specific Aggregation, which aggregates each corresponding channel of representations from neighbors respectively and considers the different preferences or importance of the intents among users' behavior neighbors and funds' knowledge neighbors propagation. (3) Independence and Balance Constraints, which add an independence regularizer on disentangled representations to encourage independence of intent, along with a balance constraint on the intent importance distribution to avoid distribution unbalance and thus increase the diversity of intent.

4.1 Knowledge-aware Disentangled Intent Modeling

To better capture the diverse fund investment intentions and utilize finance domain knowledge from the knowledge graph, we first define the knowledge-aware disentangled intent space. The intent space can be determined by representation, distribution, and volume. (1) **Representation**. To describe the multi-interest of users and the multi-attribute of funds, the representation of each node is defined as multiple channels of disentangled representations. Each channel of representation denotes a distinct intent of the node. (2) **Distribution**. The distribution of intent space for each node is to describe the different impacts of the user for different intents. For example, observed from the behaviors in Figure 1, the user Alice may have two intents, i.e. "Healthcare" sector and "ETF" type when she clicks funds. The two intents may be disentangled into two channels of representations. But Alice may value more on the sector "Healthcare" than the type "ETF", it can not be reflected from the disentangled representations. So we give each channel of representation an intent importance as the knowledge intent distribution. (3) **Volume**. The volume of the intent space determines the range of representation and distribution, which reflects the expressive ability of the intent space. Our analysis reveals that the aforementioned intents can be described by the knowledge in the knowledge graph, such as $(Fund1, fundSector, Healthcare)$ and $(Fund4, fundType, ETF)$. Therefore, attribute nodes like "fundSector" and "ETF" in the knowledge graph can represent some aspect of the latent intent of users. These attribute nodes are unambiguous and do not have multiple aspects of semantics. We aim to cluster each attribute node implicitly as one of the disentangled intent spaces. The direct approach to achieving this goal is to use a single representation for the attribute node. However, this method is incompatible with the multiple representations of the user/fund nodes, and it may harm the performance of neighbor aggregation. To overcome this issue, we restrict the intent distribution by constraining the intent importance distribution of the attribute node as a one-hot vector. In the following subsections, we will provide the mathematical form of the Knowledge-aware Disentangled Intent Modeling.

4.1.1 Initialization of Disentangled Representation. For each node $u \in \mathcal{V}$, we need to obtain multiple representation $\mathbf{H}_u^0 = \{\mathbf{H}_{uk}^0, k = 0, 1, ..., K-1\}$ with K independent components to capture nodes' multiple intents. To achieve this goal, we first project the node feature $\mathbf{X}_u$ into different channels to initialize the disentangled representations in which each channel extracts distinct semantic intent from the features.

$$\mathbf{H}_u^0 = \{\mathbf{H}_{uk}^0 = \sigma(\mathbf{W}_k^0 \mathbf{X}_u), k = 0, 1, ..., K-1\}, \tag{1}$$

where $\mathbf{W}_k^0$ is the trainable parameters, $\mathbf{H}_{uk}^0 \in \mathbb{R}^{d/K}$ is the k-th channel of representations, σ is the activate function, e.g. LeakyReLU [21]. d is the total dimension of representations and the representation dimension of each channel is d/K. It is worth highlighting that we set the same total dimension of representations $d = 64$ with the other non-disentangled methods in the experiment which will not increase the number of trainable parameters.

4.1.2 Knowledge Intent Importance Distribution. Since the importance of different intents is distinct over different nodes. For example, the top heavily invested stock is more important to a fund than the tail invested stock; A user may pay more attention to the risk level of the fund than the fund manager. Therefore, the disentangled representations are not enough to capture the accurate intents, we also need to generate the intent importance distribution:

$$\mathbf{S}_u^0 = \{\mathbf{S}_{uk}^0, k = 0, 1, ..., K-1\}, \tag{2}$$

where $\mathbf{S}_{uk}^0 \in \mathbb{R}^1$ is the importance of the intent representation $\mathbf{H}_{uk}^0$. And the total importance of all intent is set as 1, i.e., $\sum_k \mathbf{S}_{uk}^0 = 1$.

In the fund recommendation, different nodes have different distributions and there are two types of nodes. The first one is user and fund node $\mathcal{V} = \{\mathcal{U}, \mathcal{I}\}$ which contain multiple semantic intents, e.g., users' multi-interest and funds' multi-attribute. The distribution of these nodes is diverse. Another one is the attribute node which contains the nodes in the knowledge graph except funds, $C = \mathcal{E}/\mathcal{I}$. In general, their semantics are univocal, e.g., risk level, fund manager, fund type, and so on. These nodes can be regarded as potential intents for users' behaviors. Therefore, the attribute node contains only one intent, and the intent distribution of the attribute node approximates a one-hot distribution which is a sparse vector with only one non-zero dimension. In such a way, we restrict the volume of each channel of intent space. Motivated by this phenomenon, we leverage an attentive mechanism to guide the learning of the intent importance distribution.

$$\mathbf{S}_{uk}^0 = \text{Softmax}_k(\text{LeakyReLu}(\mathbf{W}_{sk}^0 \mathbf{H}_{uk}^0 / \tau_u)), \tag{3}$$

where $\mathbf{W}_{sk}^0$ is a trainable parameter and LeakyReLu is a non-linear activation. The τ_u is the temperature parameter [16]. By the temperature scaling, we give each node u a parameter τ_u to control the diverse intent distribution. When the τ_u is bigger, the result of Softmax is closer to the uniform distribution. When the τ_u is smaller, the result of Softmax is closer to one-hot distribution. So, we give a smaller τ_u for the attribute node to obtain a sparse intent distribution. We set the temperature parameter of all user/fund nodes as a constant $\tau_v = 1$ and the temperature parameter of all attribute nodes as a constant $\tau_c = 0.5$.

4.2 Intent Importance-specific Aggregation

In fund recommendation, the clicking or buying behavior is much lower than in traditional daily goods recommendation on e-commerce

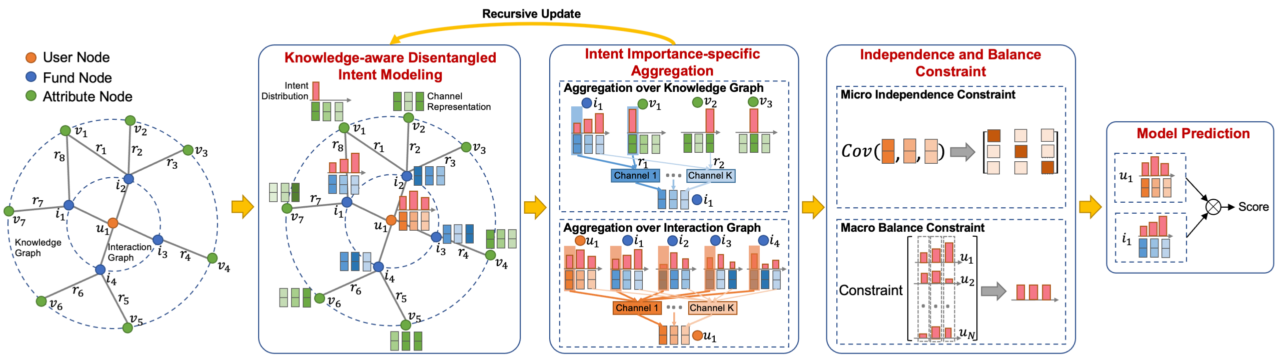

Figure 2: The framework of our proposed DIKGNN.

platforms. Therefore, it is crucial to import GNNs to aggregate neighbors' information in the interaction graph and knowledge graph and enrich multi-interest information. In this subsection, we focus on how to incorporate a neighborhood aggregation scheme into the disentangled knowledge graph learning task. Prior disentangled GNN-based methods typically use dynamic routing on the relations as minimum interest modeling, but they are insufficient for preserving the granular structure information of complex funds. Additionally, previous methods neglect to handle the intent importance distribution, which leads to inaccurate characterization of users' and funds' intents. To address these limitations, we introduce our intent importance-specific aggregation module on the interaction and knowledge graph.

4.2.1 Aggregation over Interaction Graph. First, we define the aggregation process over interaction graph. On the user-fund interaction graph, the users and funds include multiple intents. In this aggregation, we aim to retain more information for each intent, instead of choosing only one channel of intent for each neighbor like previous disentangled GNNs by neighbor routing. Therefore, we aggregate all the channels of intent representations of all neighbors and control the importance by the intent importance distribution:

$$\hat{\mathbf{H}}^{t+1}_{uk} = \sum_{v \in \mathcal{N}_{\mathcal{G}}(u)} \mathbf{S}^t_{uk} \mathbf{S}^t_{vk} A^t_k(u,v) \mathbf{H}^t_{vk}, k = 0, 1, ..., K-1, \tag{4}$$

where $\mathbf{H}^t_{uk}$ is the disentangled representation of node u's k-th channel after t aggregation layers. And $\mathbf{S}^t_{uk}$ is the node u's intent importance distribution which is calculated like the Equation 3 with the $\mathbf{H}^t_{uk}$ as input. $\mathcal{N}_{\mathcal{G}}(u)$ means node u's neighbor set on the interaction graph $\mathcal{G}$. $A^t_k(u,v)$ denotes the similarity of node u and v on the k-th channel. We learn it by a dense layer and normalize it by softmax over all neighbors:

$$A^t_k(u,v) = \text{Softmax}_{v \in \mathcal{N}_{\mathcal{G}}(u)}(\mathbf{W}^t_{ak}[\mathbf{H}^t_{uk}||\mathbf{H}^t_{vk}] + \mathbf{b}^t_{ak}), \tag{5}$$

where $\mathbf{W}^t_{ak}, \mathbf{b}^t_{ak}$ are trainable parameters and $||$ is the concatenation operator. By this equation, the aggregated representations can have a positive correlation with three patterns, i.e., the user's preference to the intent space $\mathbf{S}^t_{uk}$, the fund's importance trend to the intent attribute space $\mathbf{S}^t_{vk}$ and the similarity of user and fund among the certain intent space $A^t_k(u,v)$. However, all these terms are lower than 1 in the aggregation procedure, we add a layer normalization [1] after obtaining the aggregated representations to avoid vanishing gradient:

$$\mathbf{H}^{t+1}_{uk} = \text{LayerNorm}_k(\hat{\mathbf{H}}^{t+1}_{uk}). \tag{6}$$

4.2.2 Aggregation over Knowledge Graph. In the knowledge graph, the knowledge relations have different types and represent different semantics. We must distinguish the influence of different types of relations. Besides, different from the interaction graph, there are two types of aggregation in the knowledge graph based on the type of target node. For the user/fund nodes, the aggregation should retain more information to describe multi-interest. While for the attribute nodes, the aggregation should learn its neighbors' common intent and only keep the corresponding channel of intent representation. For example, the attribute node "Healthcare" in the Figure1 has three neighbors Fund 1, 2, 3, only one channel of the funds is related to the attribute node "Healthcare" and other channels are useless for the aggregation of the attribute node "Healthcare". Therefore, we design aggregation by the intent importance distribution as follows:

$$\hat{\mathbf{H}}^{t+1}_{pk} = \sum_{(r,q) \in \mathcal{N}_{\mathcal{G}_k}(p)} \mathbf{S}^t_{pk} \mathbf{S}^t_{qk} M^t_{rk}(p,q) \mathbf{H}^t_{pk}, k = 0, 1, ..., K-1, \tag{7}$$

where $\mathcal{N}_{\mathcal{G}_k}(p)$ is the neighbors of node p which contains both neighbor nodes and related relations. And $M^t_{rk}(p,q)$ is the importance of the knowledge relationships (p,r,q).

By the design of temperature parameters, for user/fund node, the intent distribution $\mathbf{S}^t_{pk}$ is uniform, so we can obtain all the information. As for the attribute node, the intent distribution $\mathbf{S}^t_{pk}$ is one-hot, so we can only preserve one related intent in the aggregation. So the two types of aggregation can be unified into the above common framework. For the influence of different types of relations, we learn the $M^t_{rk}(p,q)$ by the knowledge-aware attention mechanism:

$$M^t_{rk}(p,q) = \text{Softmax}_{(r,q) \in \mathcal{N}_{\mathcal{G}_k}(p)} \text{kg\_score}(\mathbf{H}^t_{pk}, \mathbf{H}^t_{qk}, r), \tag{8}$$

where kg_score is the similarity of knowledge relations, and it can be measured by any knowledge graph representation learning method [27]. For simplicity, we just use a simple method TransH [43]:

$$\text{kg\_score}(\mathbf{H}^t_{pk}, \mathbf{H}^t_{qk}, r) = (\mathbf{H}^t_{pk\perp} + \mathbf{d}_r)^T \mathbf{H}^t_{pk\perp}, \tag{9}$$

$$\mathbf{H}^t_{pk\perp} = \mathbf{H}^t_{pk} - \mathbf{w}_r^\top \mathbf{H}^t_{pk} \mathbf{w}_r, \ \mathbf{H}^t_{qk\perp} = \mathbf{H}^t_{qk} - \mathbf{w}_r^\top \mathbf{H}^t_{qk} \mathbf{w}_r \tag{10}$$

It first maps the node representation into relation space ($\mathbf{H}^t_{pk\perp}$,$\mathbf{H}^t_{qk\perp}$) and then measures the inner product similarity in the relation space. As a result, we can obtain the disentangled intent representations from the knowledge graph. We also add a layer normalization layer.

Finally, for fund nodes in both the interaction graph and knowledge graph, we just simply concatenate their aggregated representation of each channel respectively and transform them into multiple latent spaces by dense layers. We recursively conduct the aggregation F times to obtain the high-order structure information. For briefness, the number of aggregation layers is set as 2. After these aggregations among the two networks, we can obtain the final disentangled representations $\mathbf{H}^F_{uk}$ and intent importance distribution $\mathbf{S}^F_u$.

4.3 Independence and Balance Constraint

The intent represents a different aspect of information. By the information theory [5], if the representations are not independent, it contains redundant information and is less informative to describe user intents. (1) In our limited representation dimensions, we encourage our disentangled representations to be independent for better capacity and explainability with micro independence constraint. (2) Besides, we also obtain the knowledge intent importance distribution compared with previous disentangled GNNs. Hence, we can also conduct our model in the macro-level by limiting the total average distribution over all nodes to be balanced.

4.3.1 Micro Independence Constraint. To constrain the disentangled representations independently, we add a distance correlation index. This index is equal to zero only if the two inputs are independent. Since the attribute nodes are univocal and only one channel of representation is meaningful, we just apply independence constraint on user and fund nodes. We define the micro independence constraint over any two channels as follows:

$$\mathcal{L}_{cor} = \sum_k \sum_{k!=k'} \mathrm{dCor}\left(\mathbf{H}^F_{\cdot k}.\mathbf{H}^F_{\cdot k'}\right) \tag{11}$$

where $\mathbf{H}^F_{\cdot k} = [\mathbf{H}^F_{u_1 k}; ...; \mathbf{H}^F_{u_M k}; \mathbf{H}^F_{i_1 k}; ...; \mathbf{H}^F_{i_N k}] \in \mathbb{R}^{(M+N)\times \frac{d}{K}}$ is the final user/fund representations with $M = |\mathcal{U}|$ and $N = |\mathcal{I}|$. The distance correlation dCor [30] is defined as:

$$\mathrm{dCor}\left(\mathbf{H}^F_{\cdot k}, \mathbf{H}^F_{\cdot k'}\right) = \frac{\mathrm{dCov}\left(\mathbf{H}^F_{\cdot k}, \mathbf{H}^F_{\cdot k'}\right)}{\sqrt{\mathrm{dVar}\left(\mathbf{H}^F_{\cdot k}\right)\cdot \mathrm{dVar}\left(\mathbf{H}^F_{\cdot k'}\right)}}, \tag{12}$$

where dCov is the covariance matrix of two representations and dVar is the variance matrix of each representation.

4.3.2 Macro Balance Constraint. In the macro level, we limit the total average distribution over all nodes $\mathcal{V}\cup\mathcal{E}$ are uniform to avoid distribution unbalance and thus increase the diversity of intents. We use the information entropy [28] to constrain:

$$\mathcal{L}_{balance} = -\mathrm{Entropy}(\frac{\sum_{u\in\mathcal{V}\cup\mathcal{E}} \mathbf{S}^F_{u0}}{\sum_k \sum_{u\in\mathcal{V}\cup\mathcal{E}} \mathbf{S}^F_{uk}}, ..., \frac{\sum_{u\in\mathcal{V}\cup\mathcal{E}} \mathbf{S}^F_{u,K-1}}{\sum_k \sum_{u\in\mathcal{V}\cup\mathcal{E}} \mathbf{S}^F_{uk}}), \tag{13}$$

where $\mathrm{Entropy}(\mathrm{p}_1, .., \mathrm{p}_{\mathrm{K}-1})$ is defined as $\sum_{k=0,...,K-1} -p_k \log(p_k)$ and the $\sum_{u\in\mathcal{V}\cup\mathcal{E}} \mathbf{S}^F_{uk}$ is the sum of intent importance value of all nodes. The loss gets the minimum value when the sum of the intent importance value is equal to $\sum_{u\in\mathcal{V}\cup\mathcal{E}} \mathbf{S}^F_{u0} = ... = \sum_{u\in\mathcal{V}\cup\mathcal{E}} \mathbf{S}^F_{u,K-1}$.

4.4 Model Prediction

Having obtained the final representations $\mathbf{H}^F_{uk}$ and knowledge intent importance distribution $\mathbf{S}^F_u$, we define the score function Φ to estimate how likely a user will have interacted with a fund:

$$\Phi(u, v) = \sum_k \mathbf{S}^F_{uk}\mathbf{S}^F_{vk}(\mathbf{H}^{FT}_{uk}\mathbf{H}^F_{vk}), \tag{14}$$

which is the sum of the inner product of each intent representation weighted by the intent importance. Then, we use pairwise Bayesian personalized ranking loss [24] to make the scores of positive samples larger than the negative ones:

$$\mathcal{L}_{rec} = \sum_{(u,i)\in O,(u,j)\in O^-} -\ln \sigma(\Phi(u,i) - \Phi(u,j)), \tag{15}$$

where O is a positive sample set, each positive instance is an observed user-item interaction, and O^- is the negative sample set, each instance is randomly sampled from the items that the user does not adopt before to pair the user as a negative one; σ is the sigmoid function.

We alternatively optimize this recommendation loss with the independence loss (Equation 11) and balance loss (Equation 13), the total objective function is shown as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{rec} + \lambda * \mathcal{L}_{cor} + \beta * \mathcal{L}_{balance}, \tag{16}$$

where λ and β are the hyperparameters to control the weight of the constraints.

4.5 Time & Space Complexity

4.5.1 Time Complexity. The time cost of our proposed DIKGNN mainly comes from two parts, knowledge-aware disentangled intent modeling and the intent importance-specific aggregation. The time complexity of knowledge aware disentangled intent modeling is $O((|\mathcal{V}|+|\mathcal{E}|)Kd)$ where $|\mathcal{V}|$ is the number of nodes in interaction graph, $|\mathcal{E}|$ is the number of the nodes in the knowledge graph. K is the number of intent representations and d is the total embedding size. For the intent importance-specific aggregation the time complexity if $O((|\mathcal{G}| + |\mathcal{G}_k|)FKd)$ where $|\mathcal{G}|$ and $|\mathcal{G}_k|$ are the number of edges in the interaction graph and knowledge graph. F is the number of graph layers. It is easy to see that the time complexity of our model is linear to the number of nodes and edges in the graph, which is comparable to the start-of-the-art baselines.

4.5.2 Space Complexity. While we slice the node representations into K channels, we set the same total dimension of all representations (i.e., $d = 64$) with the other non-disentangled methods. The used space of our model is mainly determined by the representations of all nodes and the space complexity of our model is $O((|\mathcal{V}| + |\mathcal{E}|)d)$ which is also the same as the space complexity of the state-of-the-art baselines.

5 EXPERIMENT

In this section, we conduct extensive experiments on several tasks to demonstrate the effectiveness of DIKGNN. We use three public

Table 1: Statistics of the datasets.

		MovieLens	Last-FM	Yelp	Fund
Network	#Users	6,036	23, 566	45,919	266,902
	#Items	2,445	48, 123	45, 538	20,019
	#Interactions	376,886	3, 034, 796	1, 185, 068	861,159
	#Density	0.026	0.027	0.0005	0.00016
Knowledge Graph	#Entities	182,011	58, 266	90, 961	33,571
	#Relations	12	9	42	8
	#Triplets	1,241,995	464, 567	1, 853, 704	40,665

benchmark datasets, i.e., MovieLens, Last-FM, Yelp, and one industrial dataset Fund from a real FinTech platform to answer the following research questions:

- **RQ1:** Do the users have multi-interests and different preferences on the intents in real-world fund recommendation scenarios?
- **RQ2:** How does our proposed DIKGNN perform compared with the state-of-the-art models?
- **RQ3:** How do different parameters (e.g., the learning rate, intent number, the temperature parameter of the attribute node τ_c, knowledge intent importance distribution) affect the results of DIKGNN?
- **RQ4:** How does DIKGNN perform on real-world FinTech platforms?

5.1 Experimental Settings

5.1.1 Dataset Description. We conduct the experiment on the Fund dataset sampled from a real-world FinTech platform. The real-world FinTech platform aims to recommend funds for users. We obtain the dataset from online exposure samples over two weeks. The click samples are used as positive samples and the non-click samples are negative samples. The financial knowledge graph is collected based on public fund information from securities companies and fund institutions. The knowledge graph contains multiple relationships between funds and other attributes, such as (Fund, fundSector, Plate), (Fund, Holding, Stock), (Fund, ManagerIs, Manager), (Fund, track, Index), and so on. Since there are no public fund recommendation datasets, we also add three publicly available benchmark datasets to prove our effectiveness over the state-of-the-art baselines. MovieLens[4] is a widely used movie dataset including movie ratings collected by the GroupLens Research website. Last-FM[5] is a music dataset that contains social network, tag, and music artist information for users who listen to the Last.fm online music system. Yelp[6] is a business dataset that integrates the information of businesses, reviews, and user data collected from the 2018 edition of Yelp challenge. Following the method of KCAN [32], the related knowledge graphs of the three public benchmarks are pre-processed by matching the entities of the open knowledge graph with the items by their title names. The detailed statistics of the datasets are summarized in Table 1. The density in the table is calculated by $\frac{\#Interactions}{\#Users*\#Items}$. We can find that Fund dataset is much more sparse than other datasets.

[4]https://grouplens.org/datasets/movielens/1m/
[5]https://grouplens.org/datasets/hetrec-2011//
[6]https://www.yelp.com/dataset/challenge/

5.1.2 Evaluation Metrics. In the evaluation phase on the public benchmark, we adopt two evaluation tasks in recommendation scenarios for all methods: Top-K recommendation and CTR(Click Through Rate) prediction. (1) To evaluate Top-K recommendation, we use the leave-one-out strategy used in other works [8], and follow the same protocols [32]: Hit Ratio@K(Hit@K) and Normalized Discounted Cumulative Gain@K(NDCG@K) [7], where K is set as 10 by default. (2) In the CTR task, we use interacted items by users as positive items and randomly sampled items as negative items. The area under the curve(AUC) [22] is set to be the evaluation metric. We report the average results *w.r.t.* the metrics for all users in the test set.

5.1.3 Alternative Baselines. We compare DIKGNN with state-of-the-art methods, covering KG-free(NMF, GAT, DisenGCN), GNN-based(CKE, KGAT, KCAN) and disentangled GNN-based(DisenKGAT, KGIN) methods:

- **NMF** [8] is a factorization machine model for prediction under sparse settings. The origin algorithm does not consider a knowledge graph.
- **GAT** [33] introduces multi-head attention mechanism into neighbor aggregation.
- **CKE** [48] combines collaborative filtering with the knowledge representation learning in the embedding space.
- **KGAT** [38] extends GAT and introduces graph attention into the knowledge graph, and the attention relies on the knowledge relation.
- **KCAN** [32] introduces conditional attention in the knowledge recommendation on target-specific subgraphs to capture user preference.
- **DisenGCN** [20] uses neighborhood routing among the neighbors to obtain disentangled node representations.
- **DisenKGAT** [44] extends DisenGCN into a KG-based setting and introduces knowledge attention into the disentangled GNN.
- **KGIN** [40] models each intent as an attentive combination of KG relations to refine high-level concepts of intent and make the intent representation disentangled.

For knowledge graph-free based methods which do not consider knowledge graphs originally, we ignore the type of knowledge relations and simply combine the adjacency matrix of the interaction network and knowledge graph into a single one for introducing knowledge graph.

5.1.4 Parameter Settings. We implement the DIKGNN model in PyTorch. For a fair comparison, we adopt Adam [13] as the optimizer for all models, and fix the number of intent K as 4 and total embedding size d of all intents as 64 for disentangled GNN-based models(DisenGCN, DisenKGAT, KGIN and our DIKGNN), and fix the size of embeddings d as 64 for other models(CKE, NMF, GAT, KGAT, KCAN). The number of GNN layers is set as 2 for all GNN-based methods. Without specification, in our DIKGNN, we fix the temperature parameter of the attribute node τ_c as 0.5 for all datasets. The learning rate is tuned in $\{0.01, 0.025, 0.05, 0.1\}$, and the coefficients of the constraint loss are searched in $\{1e-2, 5e-2, 1e-1, 5e-1\}$. Moreover, in Section 5.4, we study their influence by varying K in

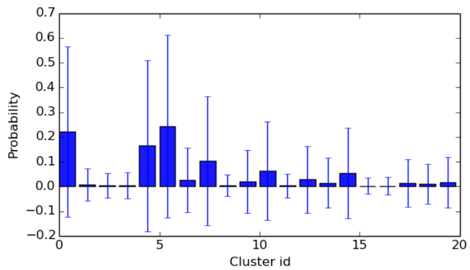

Figure 3: The probability that user clicks the cluster w.r.t. the cluster id.

$\{1, 2, 4, 8\}$, τ_c in $\{0, 0.5, 1\}$ and whether to apply knowledge intent importance distribution.

5.2 Data Analysis (RQ1)

In this subsection, we will analyze whether there are multi-interest and intent importance issues in the real fund recommendation scenario. However, there is no explicit intent label in the dataset. So we use a simple cluster method (K-means [15]) to cluster all the funds by their attributes into 20 classes and each class can implicitly represent the intent of users. And then for each user, we count the clicked funds in the past 14 days and calculate the probability that the user clicks the corresponding cluster of the fund. The mean value and standard deviation of the probability for each cluster are shown in Figure 3. The bar indicates the mean value and the error bar indicates the standard deviation. We can see that there are many non-zero values for all clusters which demonstrates the existence of the multi-intent. Besides, the standard deviation in the error bar is extremely large. It indicates that the different users have very different intents.

5.3 Performance Comparison (RQ2)

We report the empirical results of all methods on the public datasets in Table 2 and Table 3 evaluated with Top-K recommendation and CTR prediction respectively. From the performance comparison, we get the following observations:

- Our proposed DIKGNN consistently outperforms all the baselines of three datasets on both Top-K recommendation and CTR prediction. It demonstrates the effectiveness of our proposed method DIKGNN. Moreover, DIKGNN outperforms other disentangled KG-based methods DisenKGAT and KGIN in all experiments since they neglect to consider the multi-interest importance distribution. It indicates the effectiveness of the intent importance distribution to capture the intent accurately.
- The relative improvements of DIKGNN over the best baselines is larger in the sparsest dataset yelp and Fund than those of other datasets. It demonstrates our design is more stable for sparse datasets and it is important for fund recommendation which is more sparse than traditional goods recommendation.

Table 2: Hit@10 and NDCG@10 in top-K recommendation.

Dataset	MovieLens		Last-FM		Yelp		Fund	
	Hit@10	NDCG@10	Hit@10	NDCG@10	Hit@10	NDCG@10	Hit@10	NDCG@10
NFM	0.384	0.204	0.697	0.421	0.775	0.491	0.588	0.367
GAT	0.482	0.260	0.511	0.293	0.694	0.419	0.737	0.505
CKE	0.293	0.158	0.606	0.383	0.741	0.480	0.476	0.341
KGAT	0.467	0.252	0.699	0.437	0.799	0.502	0.701	0.476
KCAN	0.668	0.365	0.771	0.506	0.810	0.527	0.856	0.642
DisenGCN	0.555	0.292	0.726	0.478	0.777	0.490	0.751	0.507
DisenKGAT	0.645	0.359	0.747	0.489	0.790	0.504	0.809	0.562
KGIN	0.669	0.376	0.785	0.527	0.794	0.513	0.778	0.608
DIKGNN	**0.683**	**0.377**	**0.803**	**0.541**	**0.840**	**0.567**	**0.873**	**0.682**

Table 3: AUC value in CTR prediction.

Dataset	MovieLens	Last-FM	Yelp	Fund
NFM	0.809	0.904	0.908	0.823
GAT	0.849	0.826	0.910	0.861
CKE	0.530	0.850	0.915	0.577
KGAT	0.841	0.905	0.922	0.915
KCAN	0.907	0.923	0.937	0.935
DisenGCN	0.864	0.903	0.922	0.891
DisenKGAT	0.895	0.914	0.935	0.895
KGIN	0.906	0.931	0.942	0.919
DIKGNN	**0.909**	**0.932**	**0.949**	**0.948**

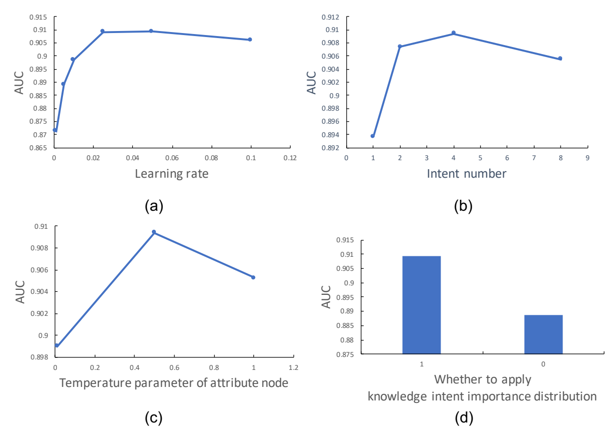

Figure 4: The impact of (a) learning rate, (b) intent number. (c) temperature parameter of attribute node, (d) knowledge intent importance distribution.

- The CKE and NFM achieve the worst results since they do not deal with the domain knowledge in the knowledge graph and they also do not have disentangled representations to capture the multi-interest. And the performance of DisenKGAT over DisenGCN, and KGIN over KGAT also validate the importance of dealing with these two issues respectively.

5.4 Hyper-parameter Sensitivity (RQ3)

In this section, hyper-parameter sensitivity studies on DIKGNN are also conducted to investigate the effectiveness with different hyper-parameters — especially, how the learning rate, the number of the intent K, the temperature parameter of the attribute node τ_c and knowledge intent importance distribution influence our model. For brevity, we only report the AUC results with the smallest MovieLens dataset, and we can observe similar trends on other datasets.

5.4.1 Impact of Learning Rate. To study the influence of learning rate, we vary the learning rate in the range of $\{0.001, 0.005, 0.01, 0.025, 0.05, 0.1\}$ and demonstrate the performance comparison in Figure 4(a). We observe that the AUC metric increases significantly from 0.001 to 0.025 in the learning rate, but decreases slowly from 0.05 to 1, and increases slowly between 0.025 and 0.05. Thus selecting a suitable learning rate is important.

5.4.2 Impact of Intent Number. We then consider varying the number of the intent K in the range of $\{1, 2, 4, 8\}$ and summarize the empirical results in Figure 4(b). We find that the performance can be improved significantly by augmenting the intent number from 1 to 4. Especially, the performance of DIKGNN is the worst when $K = 1$, indicating that the model has the worst performance when the model does not capture the multi-interests. It further proves the effectiveness of disentangling multiple interests of users. And the performance drops when the number of the intent from 4 to 8. The reasons could be that the embedding dimension of each channel become too small to represent the intent when the the number of the intent increases (e.g., $\frac{d}{K} = 8$ when $K = 8$).

5.4.3 Impact of Temperature Parameter of Attribute Node. In the proposed model, the intent importance distribution of user/fund node tends (controlled by τ_v) to be average, but the intent distribution of the attribute node (controlled by τ_c) tends to be sparse. We distinguish user/fund nodes from attribute nodes by setting different temperature parameter of node τ_u. The larger τ_u, the more average the intention distribution, the smaller τ_u, the more sparse the intention distribution. We fix the user/fund node τ_v as 1 and vary the attribute node τ_c in the range of $\{0, 0.5, 1\}$ to conduct parameter sensitivity analysis, the performance comparison is in Figure 4(c). If the τ_c is too small, it is equivalent to the complete average without attention, but if $\tau_c = 1$, it is equivalent to no distinction on the user/fund nodes and attribute nodes. They are not as good as the effect of $\tau_c = 0.5$ which indicates the effectiveness of introducing domain knowledge by limiting the volume of intent space by the attribute node of knowledge graph.

5.4.4 Impact of Knowledge Intent Importance Distribution. We conduct the ablation study on knowledge intent importance distribution to verify its effect, and the performance result is shown in Figure 4(d). The left bar is our DIKGNN and the right bar is one variant of our model to remove the knowledge intent importance distribution. We can find that applying knowledge intent importance distribution in DIKGNN can get obvious benefits. This also demonstrates assigning different importance to different intentions can help to capture the accurate intents.

5.5 Online A/B Test (RQ4)

We deploy our proposed model DIKGNN on the fund recommendation platform and conduct online A/B test(*i.e.*, bucket tests) for two weeks. For the time and business limitations, we can not run all the previous baselines in this online setting and just compare with the latest deployed model KCAN which is the best baseline in our sampled offline Fund dataset. CTR (Click-Through-Rate), a widely used industry metric, is used to measure the performance of methods online. Besides, we also take two purchase metrics, i.e., fund purchase UV(Unique Visitor) and fund purchase volume as our metrics. Detailed performance metrics on fund recommendation are shown in Table 4. We can see that our model achieves significant improvement over the online baseline and thus our model had deployed online to replace the previous model. Besides, we also analyze our model on the hit@K and diversity of the recommendation. We first compare the performance of DIKGNN with the online baseline on Hit@K. In the Hit@K setting, K in Hit@K ranges in $\{5, 10, 15, 20\}$. It can be seen that the effect of DIKGNN is significantly better than the baseline on every K in Figure 5(a).

Table 4: Relative improvement on A/B test.

Metrics	CTR	Fund Purchase UV	Fund Purchase Volume
%Imp.	0.86%	3.17%	4.36%

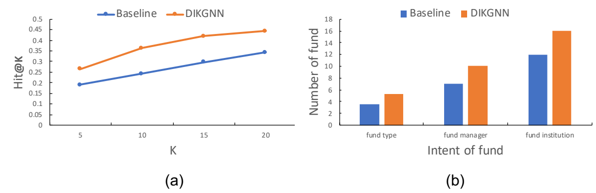

Figure 5: Performance comparison on baseline and DIKGNN.

Then we verify whether our model will harm the diversity. To compare diversity, we select users who have clicked on funds in the last 14 days and analyze diversity from different perspectives of the funds, i.e., the number of fund types, fund institutions and fund managers. We report the average results *w.r.t.* diversity for all users we selected. From Figure 5(b), we find the diversity of our model is also obviously improved compared to the online deployed model, which has a positive impact on the ecology of fund recommendation. This benefits from describing the multi-interests accurately in the fund recommendation by DIKGNN.

6 CONCLUSION

To solve the data sparsity and product complexity issue in the fund recommendation, we propose a novel model named Disentangled Interest Importance Aware Knowledge Graph Neural Network (DIKGNN) to utilize the finance domain knowledge and capture diverse fund investment intentions. We first apply multiple channels of disentangled intent representations, equipped with an intent importance distribution to describe the intent diversity. Next, we restrict the multiple intent spaces by introducing the attribute nodes from the fund knowledge graph to utilize financial domain knowledge. Then we design the users' behavior neighbor aggregation and the funds' knowledge neighbor aggregation to aggregate each corresponding channel of representations from neighbors to obtain multi-intent information. Last, we add an independence regularizer on disentangled representations and a balance cluster constraint on the intent importance distribution to increase the diversity of intent.

REFERENCES

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).

[2] Yixin Cao, Xiang Wang, Xiangnan He, Zikun Hu, and Tat-Seng Chua. 2019. Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. In *The world wide web conference*. 151–161.

[3] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Controllable multi-interest framework for recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2942–2951.

[4] Yi-Ching Chou, Chiao-Ting Chen, and Szu-Hao Huang. 2022. Modeling behavior sequence for personalized fund recommendation with graphical deep collaborative filtering. *Expert Systems with Applications* 192 (2022), 116311.

[5] Imre Csiszár, Paul C Shields, et al. 2004. Information theory and statistics: A tutorial. *Foundations and Trends® in Communications and Information Theory* 1, 4 (2004), 417–528.

[6] Jingwei Guo, Kaizhu Huang, Xinping Yi, and Rui Zhang. 2022. LGD-GCN: Local and Global Disentangled Graph Convolutional Networks. *IEEE Transactions on Neural Networks and Learning Systems* (2022), 1–12. https://doi.org/10.1109/TNNLS.2022.3195336

[7] Xiangnan He, Tao Chen, Min-Yen Kan, and Xiao Chen. 2015. Trirank: Review-aware explainable recommendation by modeling aspects. In *Proceedings of the 24th ACM CIKM*. 1661–1670.

[8] Xiangnan He and Tat-Seng Chua. 2017. Neural factorization machines for sparse predictive analytics. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. 355–364.

[9] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 639–648.

[10] Claire Yurong Hong, Xiaomeng Lu, and Jun Pan. 2019. *FinTech platforms and mutual fund distribution*. Technical Report. National Bureau of Economic Research.

[11] Pei-Ying Hsu, Chiao-Ting Chen, Chin Chou, and Szu-Hao Huang. 2022. Explainable mutual fund recommendation system developed based on knowledge graph embeddings. *Applied Intelligence* (2022), 1–26.

[12] Binbin Hu, Chuan Shi, Wayne Xin Zhao, and Philip S Yu. 2018. Leveraging meta-path based context for top-n recommendation with a neural co-attention model. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1531–1540.

[13] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[14] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[15] K Krishna and M Narasimha Murty. 1999. Genetic K-means algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 29, 3 (1999), 433–439.

[16] Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. 2019. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. *Advances in neural information processing systems* 32 (2019).

[17] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-interest network with dynamic routing for recommendation at Tmall. In *Proceedings of the 28th ACM CIKM*. 2615–2623.

[18] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*.

[19] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. 2019. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*. PMLR, 4114–4124.

[20] Jianxin Ma, Peng Cui, Kun Kuang, Xin Wang, and Wenwu Zhu. 2019. Disentangled graph convolutional networks. In *International conference on machine learning*. PMLR, 4212–4221.

[21] Andrew L Maas, Awni Y Hannun, Andrew Y Ng, et al. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, Vol. 30. 3.

[22] Simon J Mason and Nicholas E Graham. 2002. Areas beneath the relative operating characteristics (ROC) and relative operating levels (ROL) curves: Statistical significance and interpretation. *Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography* 128, 584 (2002), 2145–2166.

[23] Steffen Rendle. 2012. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3, 3 (2012), 1–22.

[24] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. 452–461.

[25] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. *Advances in neural information processing systems* 30 (2017).

[26] J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative filtering recommender systems. In *The adaptive web*. Springer, 291–324.

[27] Xiao Sha, Zhu Sun, and Jie Zhang. 2019. Attentive Knowledge Graph Embedding for Personalized Recommendation. *arXiv preprint arXiv:1910.08288* (2019).

[28] Bonnie J Strait and T Gregory Dewey. 1996. The Shannon information entropy of protein sequences. *Biophysical journal* 71, 1 (1996), 148–155.

[29] Zhu Sun, Jie Yang, Jie Zhang, Alessandro Bozzon, Long-Kai Huang, and Chi Xu. 2018. Recurrent knowledge graph embedding for effective recommendation. In *Proceedings of the 12th ACM conference on recommender systems*. 297–305.

[30] Gábor J. Székely. 2007. Measuring and testing dependence by correlation of distances. *The Annals of Statistics* 35, 6 (Dec. 2007), 2769–2794. https://doi.org/10.1214/009053607000000505 Mathematical Reviews number (MathSciNet): MR2382665; Zentralblatt MATH identifier: 1129.62059.

[31] Qiaoyu Tan, Jianwei Zhang, Jiangchao Yao, Ninghao Liu, Jingren Zhou, Hongxia Yang, and Xia Hu. 2021. Sparse-interest network for sequential recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 598–606.

[32] 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 *Proceedings of the 30th ACM CIKM*. 1834–1843.

[33] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations* (2018).

[34] Chenyang Wang, Min Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. 2020. Make it a chorus: knowledge-and time-aware item modeling for sequential recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 109–118.

[35] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2018. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM international conference on information and knowledge management*. 417–426.

[36] Hongwei Wang, Fuzheng Zhang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2019. Multi-task feature learning for knowledge graph enhanced recommendation. In *The world wide web conference*. 2000–2010.

[37] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge graph convolutional networks for recommender systems. In *The world wide web conference*. 3307–3313.

[38] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD*. 950–958.

[39] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*. 165–174.

[40] Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. 2021. Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the Web Conference 2021*. 878–887.

[41] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. 2020. Disentangled graph collaborative filtering. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 1001–1010.

[42] Xiang Wang, Dingxian Wang, Canran Xu, Xiangnan He, Yixin Cao, and Tat-Seng Chua. 2019. Explainable reasoning over knowledge graphs for recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 5329–5336.

[43] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 28.

[44] Junkang Wu, Wentao Shi, Xuezhi Cao, Jiawei Chen, Wenqiang Lei, Fuzheng Zhang, Wei Wu, and Xiangnan He. 2021. DisenKGAT: knowledge graph embedding with disentangled graph attention network. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2140–2149.

[45] Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. 2017. Deep matrix factorization models for recommender systems.. In *IJCAI*, Vol. 17. Melbourne, Australia, 3203–3209.

[46] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 974–983.

[47] Xiao Yu, Xiang Ren, Yizhou Sun, Quanquan Gu, Bradley Sturt, Urvashi Khandelwal, Brandon Norick, and Jiawei Han. 2014. Personalized entity recommendation: A heterogeneous information network approach. In *Proceedings of the 7th ACM international conference on Web search and data mining*. 283–292.

[48] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD*. 353–362.