

# Dynamic Offset Metric on Heterogeneous Information Networks for Cold-start Recommendation

**Mingshi Liu** LMS@BUPT.EDU.CN and **Xiaoru Wang**<sup>✉</sup> WXR@BUPT.EDU.CN  
*Beijing University of Posts and Telecommunications*

**Zhihong Yu** ZHIHONG.YU@INTEL.COM  
*Intel China research center*

**Fu Li** LIF@PDX.EDU  
*Portland States University*

**Editors:** Berrin Yanıkoğlu and Wray Buntine

## Abstract

The cold-start problem poses a significant challenge in recommendation systems, particularly when interaction data is scarce. While meta-learning has shown promise in few-shot classification, its application to cold-start recommendations has mostly involved simple transplantations of generic approaches. The effectiveness of metric learning, a powerful meta-learning method, is hindered by differences in problem definition when applied to rating prediction.

Heterogeneous information networks (HINs), as high-order graph structures, can capture valuable semantic information even in data-starved conditions. Efficient utilization of HINs can alleviate the cold-start dilemma. However, in the cold-start domain, there is a lack of dynamic node-level and semantic-level feature fusion schemes, resulting in the underutilization of complex information.

This study addresses these issues by combining metric learning and HINs, proposing OMHIN (Dynamic Offset Metric approach to Heterogeneous Information Networks). Our approach transforms a direct similarity metric into an indirect metric to enhance model robustness. By flexibly applying one-dimensional convolution, OMHIN effectively integrates rich information from HINs while minimizing noise introduction.

Experimental results on two datasets demonstrate that OMHIN achieves state-of-the-art performance in various scenarios, particularly in complex and challenging situations. It is especially suitable for sequence cold-start recommendations.

**Keywords:** Meta learning; Metric-learning; Adaptive feature fusion; Prototype extraction; Recommender systems

## 1. Introduction

In the era of Big Data, recommendation systems are being deployed in a considerable number of Internet services due to their ability to effectively alleviate the information overload problem [Huang et al. \(2019\)](#). However, when a new item or user enters the system, such systems typically fail to function properly due to the sparse interaction between the user and the item; this lead to the cold-start issue.

Few-shot learning, on the other hand, aims at solving pattern recognition problems with insufficient sample counts, which is quite similar to the cold-start problem. Naturally, some

studies have abstracted the cold-start problem into a Few-Shot learning problem [Zhu et al. \(2020\)](#). However, most of the current solutions are direct migrations of methods, which have some problems both in terms of model structure and at the data representation level.

In terms of the model structure, meta-learning offers an effective solution for the Few-Shot pattern recognition problem by quickly transferring recommendation experiences from other users or items to the target user or item, regardless of their novelty. While some research addresses the cold-start recommendation problem using meta-learning concepts, most of it simply replicates existing frameworks or focuses on simple binary recommendation problems. This leaves room for improvement in the practical performance of this promising method, as there are no advanced algorithmic alternatives specifically tailored to the recommendation problem.

At the data representation level. Heterogeneous information networks (HINs) connecting different classes of objects through meta-paths [Shi et al. \(2017\)](#), allowing for the construction of a more complex and comprehensive graph from limited data. applying HIN representations directly has limitations such as underutilization of partial data and disregard for inherent differences in heterogeneous information, which is particularly critical in scenarios with scarce data.

Based on the analysis above, this paper investigates two major issues. Firstly, how to adapt state-of-the-art meta-learning models to the cold-start recommendation problem. Existing approaches either apply generic model-independent meta-learning [Finn et al. \(2017a\)](#) or use meta-learning as a supplement to traditional methods like collaborative filtering for simple binary recommendations [Peng et al. \(2016\)](#). Secondly, how to effectively leverage the abundant information present in HINs. HINs offer rich information in scenarios with information scarcity, but they introduce complexity. Elements in different meta-paths may be connected differently, and elements in two meta-paths may have similar edges. Simple aggregation of heterogeneous information in HINs is inappropriate [Lu et al. \(2020a\)](#).

Therefore, we propose a Dynamic Offset Metric approach to Heterogeneous Information Networks for Cold-start Recommendation (OMHIN). OMHIN uses a newly designed metric scheme and data representation to overcome the difficulty that metric learning works well but is only applicable to simple classification problems. It not only successfully leverages the idea of metric learning to solve complex rating prediction problems than binary recommendations under cold-start conditions but also ensures fast model training while leveraging the rich information provided by the HIN.

To summarize, this paper works as follows:

- To the best of our knowledge, this is the first attempt at solving the cold-start recommendation problem using a metric learning approach and HIN. We give metrics practical meaning in terms of offset and attempt to compute metrics dynamically for various recommendation situations.
- An adaptive feature fusion module is used to ensure that the model effectively extracts various auxiliary features of the object.
- The multi-layered meta-path prototype extraction module works from similar meta-paths, avoiding the phenomenon of feature attenuation caused by blind aggregation.

- We conducted an extensive experimental study on two real-world datasets and finally demonstrated that our model achieves state-of-the-art results in majority of the metrics.

## 2. RELATED WORK

### 2.1. Cold-start Recommendations And The Application Of HIN

As previously stated, traditional recommendation algorithms are constrained by a lack of information caused by the limited interaction between new objects and thus cannot accurately recommend or suggest new items to users. This leads to the cold-start issue, which can be addressed by solving the information shortage.

One approach to solving the issue of information shortage involves introducing side information using a content-based approach to make recommendations directly in the cold state [Han et al. \(2020\)](#).

Another one focuses on the case of interaction, which is known as the few-shot problem [Vartak et al. \(2017\)](#). By using highly representative data structures such as hypergraphs [La Gatta et al. \(2022\)](#) or HINs, User-item interactions are captured using relevant characteristics. HIN encodes diverse information by revealing higher-level relationships between objects through meta-paths, expanding available data for limited interactions. HINs were first used to solve the cold-start recommendation problem by MetaHIN [Lu et al. \(2020b\)](#). Nodes and meta-paths can be aggregated using multi-level attention [Wang et al. \(2019\)](#) to consider their importance. HIN-based approaches have also yielded promising results in other recommendation [Liu et al. \(2022\)](#) and classification [Linmei et al. \(2019\)](#) domains.

Our work uses HINs as a representation while drawing on the idea of content-based recommendations and fully exploiting the higher-level information provided by the HIN.

### 2.2. Meta-learning And Metric-based Approaches

Meta-learning is a promising approach in machine learning, often used to deal with few-shot recognition problems and is also referred to as learning to learn [Thrun \(1998\)](#) which is generally classified into optimization-based, metric-based, data augmentation-based, and attention-based approaches [Wang et al. \(2020\)](#).

Optimization-based approaches are dedicated to learning a set of global parameters for optimization, independent of the specific model [Finn et al. \(2017b\)](#), and are therefore more general and have found considerable application in solving cold-start problems. MeLU [Lee et al. \(2019\)](#) applies the MAML [Finn et al. \(2017b\)](#) approach to cold-start recommendations. Meta-Emb [Pan et al. \(2019\)](#) makes use of previously learned ads through gradient-based meta-learning to initialize the embedding of its ID by feeding its contents and attributes. Inheriting the idea of MeLU, MAMO [Dong et al. \(2020\)](#) introduces the feature-specific memories, which are used to guide the model with personalized parameter initialization, while MetaHIN [Lu et al. \(2020b\)](#) attempts semantic-level adaptation for new tasks.

Metric-based approaches, on the other hand, aim to learn the similarity of two samples and were, in most cases, the first to make progress in few-shot identification. [Vartak et al. \(2017\)](#) first attempted to solve the cold-start binary recommendation problem using metric-based meta-learning approaches. Since then, RelationNet [Sung et al. \(2018\)](#)

and PrototypicalNet Snell et al. (2017) have created a proven end-to-end model for metric learning in terms of metric computation and data representation, respectively. ProtoCF 10 (2021) uses metric learning methods to improve collaborative filtering algorithms for cold-start recommendation-like problems. This model also inspired our work, which allows it to deal with relatively complex regression problems rather than the usual classification problems by giving metrics more complex meaning.

### 3. METHODOLOGY

#### 3.1. Problem Formulation

Before formulating the target problem, antecedent definitions must be introduced.

*Definition 1. Heterogeneous Information Network (HIN) Shi et al. (2017).* A HIN can be defined as a graph  $G = \{V, E, O, R\}$ .  $V$  is the ‘node set’,  $E$  is the ‘edge set’,  $O$  is the ‘node object set’ and  $R$  is the ‘relation type set’. Each node and edge satisfy a type mapping function  $\varphi_O : V \rightarrow O$  and  $\varphi_R : E \rightarrow R$  respectively. When at least one of  $|O|$  and  $|R|$  is greater than 2,  $G$  is called a HIN. A path consisting of several relations linked end to end is defined as a meta-path Sun et al. (2011). In graph  $G = \{V, E, O, R\}$ , a meta-path  $P$  with length  $l$  can be  $P = o_1 \xrightarrow{r_1} o_2 \xrightarrow{r_2} o_3 \dots \xrightarrow{r_l} o_{l+1}$ ,  $o_i \in O, r_j \in R$ . Each  $o_i$  need to be different from others. Given that in the case of this paper only one relation is established between a pair of object types, a meta-path  $P$  can be abbreviated as  $P = o_1 o_2 o_3 \dots o_{l+1}$ .

As a higher-order structure, the HIN contains more semantic information than other representations. Meta-paths aid in the extraction of this semantic information.

*Definition 2. Meta-learning.* The meta-learning process typically consists of two stages, meta-training, and meta-testing, with a labeled data-set  $D = \{(x, y)\}$  and a target task paradigm of  $F = \{S, Q\}$ . A specific sub-data-set  $D_T = \{D_{train}, D_{test}\}$  is generated in the data-set  $D$  and then a supervised task  $T$  is generated by  $D_T$  under the paradigm of  $F$ . The goal of each task is to predict the query set  $Q$  given the support set  $S$ . The loss is calculated on a per-task basis during training. We can use the permutation concept to generate a large number of tasks consisting of different query set  $Q$  and support set  $S$  obtained by permuting them in  $D$  for training on a small data set.

Since each task requires only a few or even one labeled sample, few-shot learning is a perfectly adapted testbed for meta-learning ideas.

**Metric-based meta-learning for cold-start recommendation on HIN.** For the recommendation problem, the support set includes a user  $u$  and a number of items  $i_s$ , a set of meta-paths  $P_{i_s}$  starting with  $i_s$ , and the user  $u$ ’s ratings  $r_{u,i_s}$  of the item  $i_s$ . The query set contains a number of items  $i_q$  and a set of meta-paths  $P_{i_q}$  starting with  $i_q$ , and the task goal is to predict user  $u$ ’s rating  $r_{u,i_q}$  of the items in the query set. Formally, the support set and query set can be expressed as follows:

$$S = \{u, (i_s, P_{i_s}, r_{u,i_s})\}_{s=1}^M \quad (1)$$

$$Q = (i_q, P_{i_q}, r_{u,i_q})_{q=1}^N \quad (2)$$

$M$  and  $N$  represent the total number of items in the support set and the query set, respectively. Note that the items in the query set and support set are mutually exclusive. Unlike the general C-way K-shot problem, the number of items in the support set corresponding

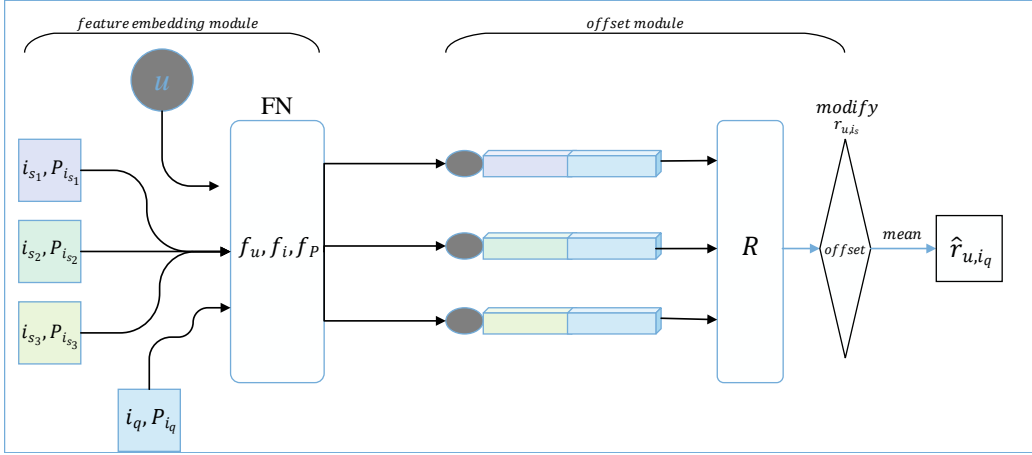


Figure 1: Overview of OMHIN with 1 sample in query set and 3 samples in support set

to each rating is not fixed. According to MeLU, several types of cold-start scenarios can be distinguished by the different organization of the test task: a test task is **user cold-start (UC)**, if it contains a user that has not appeared in meta-training but an item that has appeared in the training phase. Conversely, the task is considered **item cold-start (IC)**. In particular, if both the user and the item are appearing for the first time, then we are faced with a **user and item cold-start (UIC)**.

### 3.2. Model Overview

Our work is inspired by several metric learning-based methods, and uses HIN to obtain more semantic information from fewer samples.

As shown in Fig. 1 OMHIN consists of two parts: the feature embedding module and the offset module. The feature embedding module extracts high-dimensional features of users, items, and meta-paths. The offset module is in charge of computing the metric between different items, which we call the offset. The training goal is to find a suitable feature extraction network and offset calculation function so that the offset function can calculate the offset that accurately reflects the variation in user ratings of items in the embedding space conditional on a specific user generated by the feature extraction network.

The model is input one complete task at a time, and in this case, a task contains three samples of support items and one sample of query items, i.e.,  $M = 3$  and  $N = 1$ . Among the model inputs,  $u$  represents the users in the support set, and for each task, there is only a single user,  $i_s$  represents the items in the support set,  $i_q$  represents the items in the query set, while  $P_i$  represents the set of meta-paths corresponding to each item  $i$ , containing several subsets of meta-paths differentiated by their path types. They are all fed into the feature extraction network ( $FN$ ). For the user, the items, and the set of meta-paths,  $FN$  calculates the feature map  $f_u(u)$ ,  $f_i(i)$ , and  $f_P(P)$  separately. In the offset module, the feature maps are concatenated by operation  $C$  as follows.

$$F_{u,i_s,i_q} = C(f_u(u), f_i(i_s), f_P(P_{i_s}), f_i(i_q), f_P(P_{i_q}))_{s=1}^M \quad (3)$$

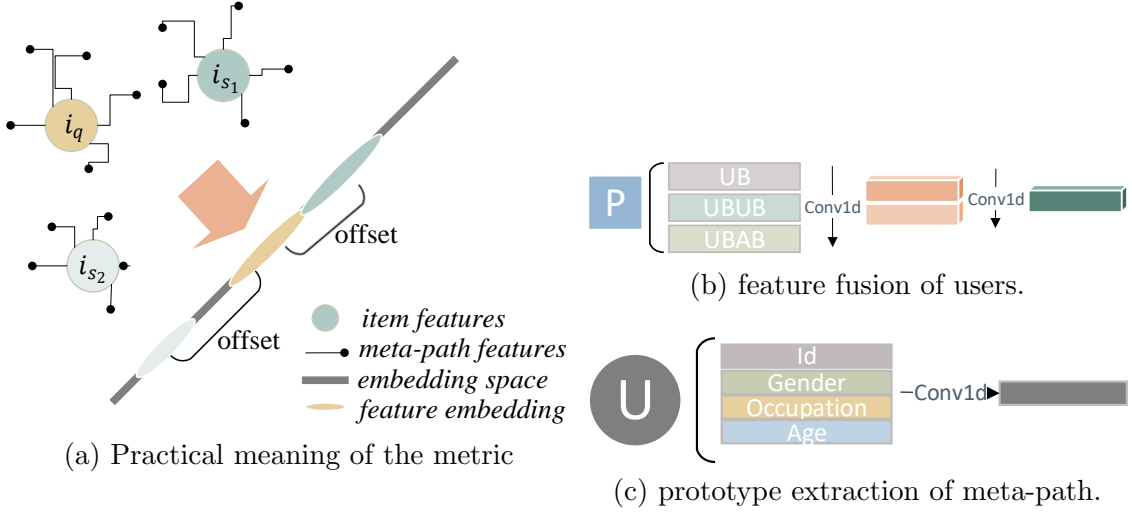


Figure 2: Model details of OM

During the initial stages of model development, specialized models for user, item, and meta-path features led to overfitting. Consequently, streamlining the model and delegating direct feature extraction to the offset module, while preserving vital graph topology information, proves more effective for ensuring information enrichment. The concatenated features are then fed into offset calculation function ( $R$ ), which computes a signed quantity  $b$  representing the user rating offset over the item samples in the support set. For each item sample in the support set,  $R$  generates an offset:

$$b_{u,i_s,i_q} = R(F_{u,i_s,i_q})_{s=1}^M \quad (4)$$

The result of modifying the support item ratings with their corresponding offset  $\hat{r}_{u,i_s,i_q}$  is averaged to obtain the predicted value  $\hat{r}_{u,i_q}$  of the user’s rating for the query item.

$$\hat{r}_{u,i_s,i_q} = \text{modification}(r_{u,i_s}, b_{u,i_s,i_q}) \quad (5)$$

$$\hat{r}_{u,i_q} = \text{mean}\left(\{\hat{r}_{u,i_s,i_q}\}_{s=1}^M\right) \quad (6)$$

### 3.3. More Details

**The practical meaning of metric.** Cold-start recommendations pose unique challenges compared to general few-shot problems. One key difference is the limited coverage of user ratings, especially for new users. This affects the support sets used in metric learning, as query samples may not belong to any of the classes in the support set. The conventional approach, which assumes a default class, is ineffective in addressing the cold-start problem. Moreover, the presence of repeated user ratings and the haphazard nature of support sets significantly impact similarity calculations. To tackle these challenges, we propose a practical modification to the metric. By mapping item and user features to an embedding space using  $FN$ , we calculate the metric  $b_{u,i_s,i_q}$ .

$$b_{u,i_s,i_q} \leftarrow r_{u,i_q} - r_{u,i_s} \quad (7)$$

This metric represents the offset of the user’s ratings for query samples relative to support samples. The offset, resembling a signed vector, indicates whether the user’s rating for the query sample is higher or lower than the support sample. We leverage these offsets to obtain the query sample rating by adjusting the user’s rating of the support sample through a modification operation defined in our paper.

$$modification(r_{u,i_s}, b_{u,i_s,i_q}) = r_{u,i_s} + b_{u,i_s,i_q} \quad (8)$$

By giving the metric a practical meaning, our model indirectly predicts the classification of samples not present in the support set. This approach effectively addresses the challenges of cold-start recommendations. Fig. 2(a) shows item features, meta-paths, and the mapping to the embedding space.

**Adaptive feature fusion.** As shown in Fig. 2(b), a user can have various characteristics, like an ID, age, occupation and gender. Previous approaches have typically taken the average of these features and treated them equally, sometimes using weights for better blending. However, attention mechanisms can be complex. To address this, we use one-dimensional convolution to dynamically assign adaptive weights to each feature. In Fig. 2(b), the user feature map includes four features: ID, gender, occupation, and age. By applying a convolutional kernel of length 1, the user’s embedded features, denoted as  $f_u(u)$ , can be represented as

$$f_u(u) = w_1u_{[Id]} + w_2u_{[Gender]} + w_3u_{[Occupation]} + w_4u_{[Age]} \quad (9)$$

The mythology can be generalized that for several features  $u_{[i]}$ , embedded features can be dynamically calculated as

$$f_u(u) = \Sigma(w_iu_{[i]}). \quad (10)$$

The target of  $f_u$  in the feature extraction network is to learn the convolutional model during training. As a result, we give the model the ability to construct an attention process that allows it to focus adaptively the important features. For a more complex and non-linear process of establishing weights, the number of convolutional layers can be enlarged or by increasing the length of the convolutional kernel.

**Multi-layered meta-path prototype extraction.** Our method extracts multi-layered meta-path prototypes by considering multiple meta-paths starting from a sample. Meta-paths capture complex information, differentiating and relating various types. Previous approaches, such as averaging all meta-paths for parameter updates [Lu et al. \(2020b\)](#), overlook semantic richness and introduce noise. Our aim is to obtain a prototype that effectively expresses the information from multiple meta-paths.

In Fig. 2(c), meta-paths starting from items are divided into sets ( $UB$ ,  $UBUB$ , and  $UBAB$ ) representing User-Book, User-Book-User-Book, and User-Book-Author-Book meta-paths. Averaging the meta-paths within each set yields representative features. The relationships between  $UB$  and  $UBUB$  are stronger than with  $UBAB$  due to their shared relationship. Similarly, due to their equal length,  $UBAB$  is more similar to  $UBUB$  than  $UB$ . By organizing similar meta-paths adjacent to each other and applying one-dimensional convolution in a distinct dimension, we capture similarity, and then perform one-dimensional convolution in a different dimension from the user operation. The convolution kernel must be at least 2 for first-layer prototype extraction. Each layer extracts a single prototype

---

**Algorithm1** Training Procedure of OMHIN

---

**Input:** Training tasks  $T^{tr}$ , meta-path types  $t$   
**for**  $T = (S = \{u, (i_s, P_{i_s}, r_{u,i_s})\}, Q = (i_q, P_{i_q}, r_{u,i_q}))$  in  $T^{tr}$  **do**  
  **for**  $P$  in  $t$  **do**  
     $P_{i_s}[P] = \text{mean}(P_{i_s}\{P\})$   
     $P_{i_q}[P] = \text{mean}(P_{i_q}\{P\})$   
  **end for**  
   $F_s = (f_u(u), f_i(i_s), f_P(P_{i_s})) = FN(S)$   
   $F_q = (f_i(i_q), f_P(P_{i_q})) = FN(Q)$   
  Concatenate  $F_s$  and  $F_q$  by  $C(\bullet, \bullet, \dots)$  to constitute  $F_{u,i_s,i_q}$   
   $b_{u,i_s,i_q} = R(F_{u,i_s,i_q})$   
   $\hat{r}_{u,i_q} = \text{mean}(\{r_{u,i_s} + b_{u,i_s,i_q}\}_s)$   
   $L = \text{mse}(\hat{r}_{u,i_q}, r_{u,i_q})$   
  Update  $FN, R$  using  $L$  by Adam optimization algorithm  
**end for**

---

for similar meta-paths.  $f_p$  performs multiple one-dimensional convolutions to extract all meta-paths as a single prototype, integrating semantic information layer by layer.

### 3.4. Training Procedure

**Algorithm 1** summarizes the training steps of our proposed OMHIN. In a training epoch, a set of training tasks  $T^{tr}$  and a meta-path classification  $t$  organized by similarity is input. As shown in Algorithm 1, each task contains a training set  $S$  and a query set  $Q$ ,  $P_i\{P\}$  denotes the set of all meta-paths with category  $P$  beginning with the item  $i$ , and  $P_i\{P\}$  denotes the representative features of meta-paths with category  $P$  beginning with the item  $i$ . We use the Mean Square Error (MSE) loss training model that

$$\text{mse}(\hat{r}_{u,i}, r_{u,i}) = \frac{1}{m} \sum_i^m (r_{u,i} - \hat{r}_{u,i})^2 \quad (11)$$

## 4. EXPERIMENTS

### 4.1. Environments and Datasets

All experiments below were executed on the PyTorch 1.8.0 GPU deep learning platform, with python version 3.8.0. the server used for experiments was equipped with RTX3090 graphics cards and an Intel Xeon CPU, and the operating system was Linux Ubuntu 18.04LTS. The experiments are also compatible with Windows.

We build cold-start recommendation tasks on two realistic datasets, DBook<sup>1</sup> and MovieLens<sup>2</sup>. Table.1 describes these two datasets and the details related to the experiments in this paper. DBook is a Douban book rating dataset. In this dataset, we consider the case where users and items only have one feature and three types of meta-paths, focusing on

---

1. <https://book.douban.com>

2. <https://grouplens.org/datasets/movielens>



Table 1: Statistics Of The Two Datasets

	<b>DBook</b>	<b>MovieLens</b>
Node type and number of nodes in each type	User(U): 10,592, Author(A): 10,544, Book(B): 20,934	User(U): 6,040, Actor(A): 8,030, Director(D): 2,186, Movie(M): 3,881
Task objectives	Predicting User ratings for Books	Predicting User ratings for Movies
Edge type and number of edges in each type	UB: 649,381, BA: 20,934	UM: 1,000,209, AM: 15,398, DM: 4,210
Meta-paths considered	UB,UBUB,UBAB	UM,UMUM,UMAM,UMDM
User features	Location	ZipCode, Gender, Age, Occupation
Percentage of difficult tasks	17.89%	29.98%

whether OMHIN can be used for cold-start recommendations. MovieLens is a stable benchmark dataset widely used for recommendations that contain user ratings of movies. This dataset, on the other hand, is more complicated. We investigate how well OMHIN works by examining more sophisticated user and item features with four meta-paths and much more difficult tasks.

We build on previous work [Lee et al. \(2019\)](#) by considering users with more than 13 and less than 100 ratings and the items they rated, using 10 of these ratings as the query set and another 3-5 ratings as the support set to build the few-shot learning task, which simulates new users with only 3 to 5 rating records as closely as possible. We chose users and items that were later added to the dataset as new users and new items. The meta-training tasks were all composed of existing users and items, and four meta-testing tasks, user cold-start(UC), item cold-start(IC), user and item cold-start(UIC), non-cold-start(NC), were constructed according to the previous problem definition. In terms of difficulty, non-cold-start < user cold-start and item cold-start < user-item cold-start. A task is considered difficult if it contains a query sample whose rating does not exist in the support set because traditional metric learning approaches are incapable of making correct predictions in any case.

## 4.2. Experimental Setup

Fig. 3 depicts the OMHIN model architecture. The experimental configuration sets the embedding dimension of each attribute to 32, which determines the values of all undefined parameters in Fig. 3.

$FN$  is a convolutional neural network composed of three modules. Where  $f_i$  consists of linear layers that do not modify feature dimensions.  $f_u$  adds two additional convolutional modules that do not change the dimension of features for enhancing non-linearity to the basic one-layer convolutional module with a one-dimensional convolutional kernel of 3.  $f_p$  also introduces an additional layer of convolutional modules, here the number of filters is determined by the number of meta-path feature dimensions. The initial number of meta-path prototypes determines the size of the convolution kernel for the second layer of convolution modules. For it to output one prototype within two layers of convolution, the setting should be (2,2) for DBook, and for MovieLens of four initial prototypes, this setting is (2,3).  $f_i$  and  $f_p$  share a fully connected module, which ensures that item features and meta-path features are mapped to the same embedding space to calculate the offsets.

First,  $R$  splices the embedding features exported by  $FN$  before computing offsets. To ensure a fair comparison with baselines, all fully connected layers adopt the same depth and

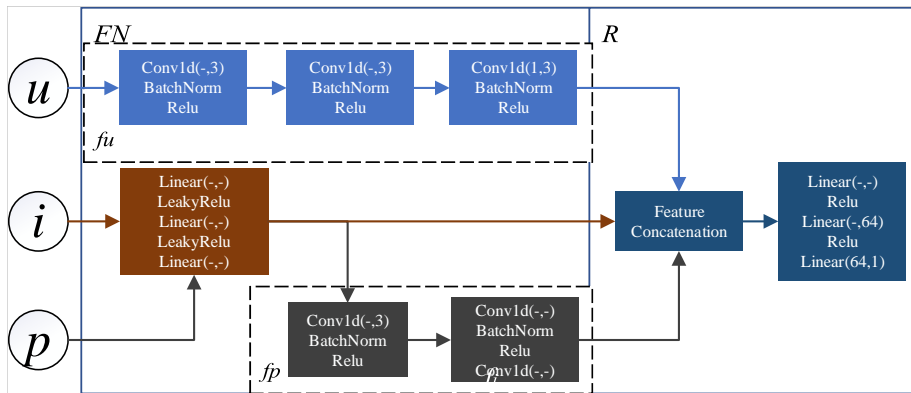


Figure 3: Architecture of OMHIN

parameter settings as other meta-learning-based methods, with the value of the undefined part being equivalent to the dimension of the spliced features.

We compare OMHIN to three categories of our previous work, which are traditional methods (FM Rendle et al. (2011) and NeuMF He et al. (2017)), HIN-based methods (mp2vec Dong et al. (2017) and HERec Shi et al. (2019)) and meta-learning-based methods (MeLU Lee et al. (2019) and MetaHIN Lu et al. (2020b)). MetaHIN, in particular, is an HIN-based approach. For the non-HIN approach, we reduce node objects to item features. All meta-training and meta-testing tasks are combined into a training set and a testing set for non-meta-learning methods.

We adopt two popular metrics. One is the mean absolute error (MAE), which measures the accuracy of the rating prediction; the lower the value, the more accurate the prediction. The other is the normalized discounted cumulative accuracy at rank K (nDCG@K), which represents the prediction performance of the top-K rankings, the higher the value, the better the performance. In this study, the values of K are set to 3 and 5.

Each group of experiments is conducted a minimum of ten times, and the evaluation metrics are then averaged over at least ten rounds after the model has converged to ensure the validity and reliability of the results.

### 4.3. Comparison with State-of-the-arts

We compared the overall performance of OMHIN under different scenarios on both the datasets with the baseline. Table.2 and Table.3 presents the comparison results on DBook and MovieLens datasets for four scenarios. Figure.4 depicts the speeds of fitting for the different model compared to the baseline model.

OMHIN excels in predicting ratings across scenarios, similar to the state-of-the-art baseline. Traditional approaches struggle with complex contexts like MovieLens, while our adaptive fusion of user features and meta-paths propels OMHIN to outperform, enhancing nDCG@5 and nDCG@3 by 3.83-5.46% and 3.90-5.25%. Utilizing heterogeneous information significantly improves problem-solving. Our robust model effectively tackles few-shot problems, surpassing the traditional approach. OMHIN’s prioritization of sample relationships yields higher ranking accuracy. Moreover, our model remains robust even under challenging

Table 2: Performance Comparison in Cold-start Scenario

A) User cold-start(UC)						
Model	DBook			MovieLens		
	MAE	nDCG@5	nDCG@3	MAE	nDCG@5	nDCG@3
FM	0.7027	0.8032	-	1.0421	0.7303	-
Neu-MF	0.6541	0.8225	-	0.8569	0.7708	-
mp2vec	0.6619	0.8144	-	0.8793	0.8233	-
HERec	0.6518	0.8233	-	0.8691	0.8389	-
MeLU	0.6353	0.8793	0.8572	0.8104	0.8415	0.8182
MetaHIN	<b>0.6018</b>	<b>0.9004</b>	0.8738	0.8053	0.8327	0.8105
<b>OMHIN</b>	0.6092	0.8946	<b>0.8791</b>	<b>0.7929</b>	<b>0.8754</b>	<b>0.8552</b>

B) Item cold-start(IC)						
Model	DBook			MovieLens		
	MAE	nDCG@5	nDCG@3	MAE	nDCG@5	nDCG@3
FM	0.7186	0.8342	-	1.3488	0.7218	-
Neu-MF	0.7062	0.7396	-	0.9822	0.6063	-
mp2vec	0.7371	0.8231	-	1.0615	0.6367	-
HERec	0.7481	0.7827	-	0.9959	0.7312	-
MeLU	0.6518	0.8882	0.8697	0.9196	0.8041	0.7824
MetaHIN	0.6262	<b>0.8897</b>	0.8703	<b>0.8696</b>	0.8337	0.8106
<b>OMHIN</b>	<b>0.6244</b>	0.8932	<b>0.8735</b>	0.8710	<b>0.8656</b>	<b>0.8423</b>

C) User and item cold-start(UIC)						
Model	DBook			MovieLens		
	MAE	nDCG@5	nDCG@3	MAE	nDCG@5	nDCG@3
FM	0.8326	0.8201	-	1.3001	0.7015	-
Neu-MF	0.6949	0.8566	-	0.9686	0.8063	-
mp2vec	0.7987	0.8527	-	1.0548	0.6687	-
HERec	0.7859	0.8545	-	0.9974	0.7389	-
MeLU	0.6517	0.8891	0.8740	0.9091	0.8106	0.7911
MetaHIN	0.6325	0.8891	<b>0.8755</b>	<b>0.8584</b>	0.8408	0.8204
<b>OMHIN</b>	<b>0.6238</b>	<b>0.895</b>	0.8739	0.8707	<b>0.8867</b>	<b>0.8635</b>

Table 3: Performance Comparison in Non-cold-start Scenario

warmModel	DBook			MovieLens		
	MAE	nDCG@5	nDCG@3	MAE	nDCG@5	nDCG@3
FM	0.7358	0.8086	-	1.0043	0.6493	-
Neu-MF	0.6904	0.7924	-	0.9249	0.7335	-
mp2vec	0.6897	0.8342	-	0.8788	0.7091	-
HERec	0.6794	0.8411	-	0.8652	0.7182	-
MeLU	0.6523	0.8701	0.8482	0.8132	0.8475	0.8167
MetaHIN	<b>0.6396</b>	<b>0.8864</b>	<b>0.8630</b>	0.8001	0.8354	0.8099
<b>OMHIN</b>	0.6656	0.8830	0.8571	<b>0.7874</b>	<b>0.8753</b>	<b>0.8566</b>

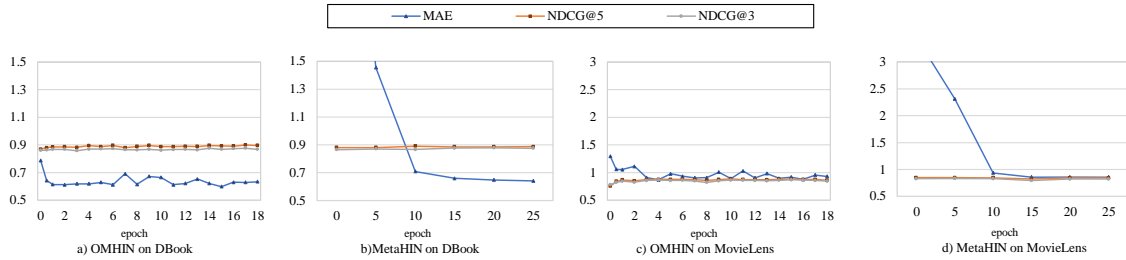


Figure 4: Convergence of OMNIN and MetaHIN in the UIC scenario. The first and second half of the batch in the first epoch are calculated separately for the metrics in order to study the convergence of OMHIN.

conditions. It shows notable improvement in more difficult scenarios, approaching few-shot problems effectively. Comparatively, the meta-learning-based approach surpasses the traditional approach, with OMHIN showing notable improvements in nDCG@K. OMHIN focuses on relationships between samples rather than performing a direct mapping between sample features and ratings, resulting in a higher ranking accuracy.

OMHIN and MetaHIN share complexity and parameter settings, but OMHIN’s convolutional layers slightly increase complexity. However, OMHIN requires fewer epochs to converge to a similar solution to MetaHIN. Fig. 4 illustrates OMHIN’s faster convergence compared to metaHIN (3 epochs vs. 15 epochs). Our model effectively handles data-poor situations.

In conclusion, our model achieves significant improvements in dealing with complex and difficult scenarios and the fastest fitting speed.

#### 4.4. Ablation Experiments

We conducted ablation experiments to analyze the impact of individual improvements on our model. Two sets of experiments investigated metric patterns and feature extraction methods. The recorded performance focused on UIC, the most challenging scenario, which reflects performance in other scenarios.

Table 4: Ablation Study

a) Impact of the practical meaning of the metric						
Model	DBook			MovieLens		
	MAE	nDCG@5	nDCG@3	MAE	nDCG@5	nDCG@3
OMHIN	0.651	0.895	0.8739	0.8707	0.8867	0.8635
OMHIN-A-M	0.6634	0.8877	0.8615	0.9656	0.8425	0.8137
OMHIN-P-A-M	0.7245	0.8888	0.8600	1.2329	0.8509	0.8122
b) Improvements of feature fusion and prototype extraction						
Model	DBook			MovieLens		
	MAE	nDCG@5	nDCG@3	MAE	nDCG@5	nDCG@3
OMHIN-A-M	0.6294	0.8877	0.8615	0.9656	0.8425	0.8137
OMHIN-A	0.6274	0.8878	0.8628	0.9342	0.875	0.8566
OMHIN-M	0.6221	0.8951	0.8665	0.9117	0.8469	0.8263
OMHIN	0.6238	0.895	0.8739	0.8707	0.8867	0.8635
MetaHIN	0.6325	0.8891	0.8755	0.8584	0.8408	0.8204

Regarding the metric analysis, we examined two ablation models: OMHIN-P-A-M (basic relational metric model) and OMHIN-A-M (metric with practical meaning of offset). Introducing the offset significantly improved OMHIN’s prediction accuracy, especially on MovieLens with difficult tasks. However, the nDCG@K metric showed a slight dip due to introduced uncertainty and continuity. Other improvement modules compensated for these issues, as OMHIN consistently outperformed OMHIN-A-M across all metrics. Table.4(a) shows the experimental results.

We also explored the benefits of feature fusion and prototype extraction using OMHIN-A-M. Experimental results are recorded in Table.4(b). OMHIN-M added an adaptive user

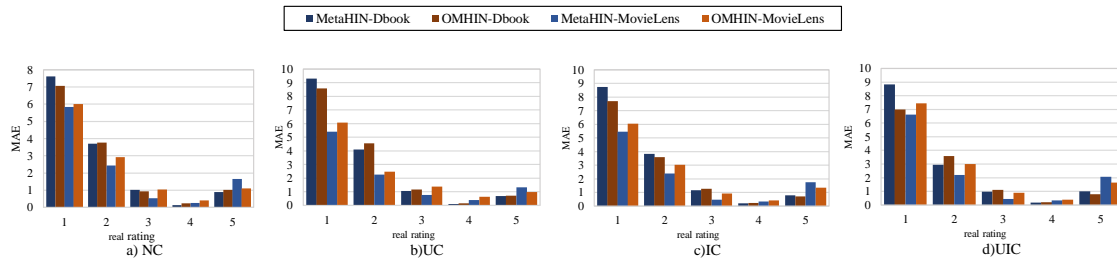


Figure 5: MAE of the predicted ratings for the various true ratings.

feature fusion module, while OMHIN-A added a meta-path prototype extraction module. On the DBook dataset, where users have only one feature, the improvement effect of OMHIN-M was not apparent. However, on the MovieLens dataset, the improvement modules made significant progress. The meta-path prototype extraction module leveraged rich semantic information, while the feature fusion module followed a traditional data representation pattern.

Additionally, the metric-based learning models consistently outperformed the model optimization-based approach in terms of nDCG@K metrics.

#### 4.5. Possible applications

OMHIN excels in top-K item ranking based on previous experiments. We conducted additional experiments to explore potential application scenarios. In Fig. 5, We compare the predicted ratings in NC, UC, IC, and UIC scenarios with MetaHIN. Notably, OMHIN markedly outperformed MetaHIN when the true rating was 5 in cold-start scenarios, and the margin of error for low-rated items had minimal impact on the 5-point sample, which further explaining OMHIN’s strong performance on the nDCG@K metric. In summary, it is easy to conclude that our model has applications in sequential cold-start recommendations that are more concerned with the descending fine ranking of ratings.

## 5. Conclusion

We propose a dynamic offset metric-learning approach for cold-start recommendation on heterogeneous information networks (OMHIN). We attempt to give metrics practical meaning so that metric-learning models can be adapted to more complex and difficult tasks. Specifically, we use the concept of data augmentation to alleviate the cold-start problem associated with both traditional data representation and HIN. The model is specifically designed to treat metrics as signed rating offsets rather than simple relational scores, while also introducing dynamic feature fusion and extracting meta-path prototypes. Experiments on two datasets show that OMHIN achieves state-of-the-art performance in a variety of scenarios, with significant improvements in complex and difficult scenarios, and is especially well-suited for use in sequence cold-start recommendations.

In the future, we will leverage the aforementioned advantages of OMHIN to conduct large-scale experiments and applications on more complex datasets. Additionally, this paper

demonstrates that metric-based approaches possess generality comparable to optimization-based methods, which will contribute to our further research on universal advanced meta-learning methods.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 61976025.

## References

- Proctof: Prototypical collaborative filtering for few-shot recommendation. In *Proceedings of the 15th ACM Conference on Recommender Systems, RecSys '21*, page 166–175, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450384582. doi: 10.1145/3460231.3474268.
- Manqing Dong, Feng Yuan, Lina Yao, Xiwei Xu, and Liming Zhu. MAMO: memory-augmented meta-optimization for cold-start recommendation. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 688–697. ACM, 2020.
- Yuxiao Dong, Nitesh V. Chawla, and Ananthram Swami. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pages 135–144. ACM, 2017. doi: 10.1145/3097983.3098036.
- Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 2017a.
- Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 2017b.
- Jiayu Han, Lei Zheng, Yuanbo Xu, Bangzuo Zhang, Fuzhen Zhuang, Philip S. Yu, and Wanli Zuo. Adaptive deep modeling of users and items using side information for recommendation. *IEEE Transactions on Neural Networks and Learning Systems*, 31(3): 737–748, 2020. doi: 10.1109/TNNLS.2019.2909432.
- Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 173–182. ACM, 2017. doi: 10.1145/3038912.3052569.

- Zhenya Huang, Qi Liu, Chengxiang Zhai, Yu Yin, Enhong Chen, Weibo Gao, and Guoping Hu. Exploring multi-objective exercise recommendations in online education systems. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 1261–1270. ACM, 2019. doi: 10.1145/3357384.3357995.
- Valerio La Gatta, Vincenzo Moscato, Mirko Pennone, Marco Postiglione, and Giancarlo Sperlí. Music recommendation via hypergraph embedding. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2022. doi: 10.1109/TNNLS.2022.3146968.
- Hoyeop Lee, Jinbae Im, Seongwon Jang, Hyunsouk Cho, and Sehee Chung. Melu: Meta-learned user preference estimator for cold-start recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 1073–1082. ACM, 2019. doi: 10.1145/3292500.3330859.
- Hu Linmei, Tianchi Yang, Chuan Shi, Houye Ji, and Xiaoli Li. Heterogeneous graph attention networks for semi-supervised short text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4821–4830, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1488.
- Jiawei Liu, Chuan Shi, Cheng Yang, Zhiyuan Lu, and Philip S. Yu. A survey on heterogeneous information network based recommender systems: Concepts, methods, applications and resources. *AI Open*, 3:40–57, 2022. ISSN 2666-6510. doi: <https://doi.org/10.1016/j.aiopen.2022.03.002>.
- Yuanfu Lu, Yuan Fang, and Chuan Shi. Meta-learning on heterogeneous information networks for cold-start recommendation. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1563–1573. ACM, 2020a.
- Yuanfu Lu, Yuan Fang, and Chuan Shi. Meta-learning on heterogeneous information networks for cold-start recommendation. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1563–1573. ACM, 2020b.
- Feiyang Pan, Shuokai Li, Xiang Ao, Pingzhong Tang, and Qing He. Warm up cold-start advertisements: Improving CTR predictions via learning to learn ID embeddings. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pages 695–704. ACM, 2019. doi: 10.1145/3331184.3331268.
- Furong Peng, Xuan Lu, Jianfeng Lu, Richard Yi-Da Xu, Cheng Luo, Chao Ma, and Jingyu Yang. Metricrec: Metric learning for cold-start recommendations. In *Advanced Data Mining and Applications*, pages 445–458, Cham, 2016. Springer International Publishing. ISBN 978-3-319-49586-6.

- Steffen Rendle, Zeno Gantner, Christoph Freudenthaler, and Lars Schmidt-Thieme. Fast context-aware recommendations with factorization machines. In *Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, Beijing, China, July 25-29, 2011*, pages 635–644. ACM, 2011. doi: 10.1145/2009916.2010002.
- Chuan Shi, Yitong Li, Jiawei Zhang, Yizhou Sun, and Philip S. Yu. A survey of heterogeneous information network analysis. *IEEE Transactions on Knowledge and Data Engineering*, 29(1):17–37, 2017. doi: 10.1109/TKDE.2016.2598561.
- Chuan Shi, Binbin Hu, Wayne Xin Zhao, and Philip S. Yu. Heterogeneous information network embedding for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 31(2):357–370, 2019. doi: 10.1109/TKDE.2018.2833443.
- Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4077–4087, 2017.
- Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S. Yu, and Tianyi Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proc. VLDB Endow.*, 4(11):992–1003, 2011. ISSN 2150-8097. doi: 10.14778/3402707.3402736.
- Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1199–1208. IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00131.
- Sebastian Thrun. Lifelong learning algorithms. *Learning to learn*, 8:181–209, 1998.
- Manasi Vartak, Arvind Thiagarajan, Conrado Miranda, Jeshua Bratman, and Hugo Larochelle. A meta-learning perspective on cold-start recommendations for items. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6904–6914, 2017.
- Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S. Yu. Heterogeneous graph attention network. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 2022–2032. ACM, 2019. doi: 10.1145/3308558.3313562.
- Xiaoru Wang, Bing Ma, Zhihong Yu, Fu Li, and Yali Cai. Multi-scale decision network with feature fusion and weighting for few-shot learning. *IEEE Access*, 8:92172–92181, 2020. doi: 10.1109/ACCESS.2020.2994805.
- Yu Zhu, Jinghao Lin, Shibi He, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. Addressing the item cold-start problem by attribute-driven active learning. *IEEE Transactions on Knowledge and Data Engineering*, 32(4):631–644, 2020. doi: 10.1109/TKDE.2019.2891530.