

# Multi-Feature Video Recommendation Based on Hypergraph Convolution for Mobile Edge Environment

Haiyan Wang, Nanjing University of Posts and Telecommunications, China\*

Jun Hong, Nanjing University of Posts and Telecommunications, China

Kaixiang You, Nanjing University of Posts and Telecommunications, China

Jian Luo, Nanjing University of Posts and Telecommunications, China

## ABSTRACT

With the massive growth of edge devices, how to provide users with video recommendation services in a mobile edge environment has become a research hotspot. Most traditional video recommendation methods regard the relationship between user and neighbor to be linear and ignore higher-order connectivity among users, which results in poor recommendation performance. Besides, these methods use a single feature to represent user preferences, which cannot effectively alleviate the data sparsity problem. To improve the performance of video recommendation, this article proposes a multi-feature video recommendation method based on hypergraph convolution (MVRHC). Hypergraph convolution is adopted to compute user neighborhood-level features for modeling high-order correlations among users. Final features are obtained by fusing multi-party features through attention mechanism. And video recommendation is then implemented based on the obtained features. Experimental results on two real-world datasets demonstrate that MVRHC has better performance compared with baseline methods.

## KEYWORDS

Attention Mechanism, Feature Extraction, Hypergraph, Mobile Edge Computing, Multi-Feature, Video Recommendation

## INTRODUCTION

With the broad development of mobile edge computing, various types of edge devices are deployed on edge nodes, and more and more services can be obtained through these devices. However, when faced with a large number of services, users find it difficult to discover required services according to their preferences. Recommendation systems have therefore become an effective mitigation method, and accurate recommendation methods can push products/services in terms of users' interests.

DOI: 10.4018/JDM.325351

\*Corresponding Author

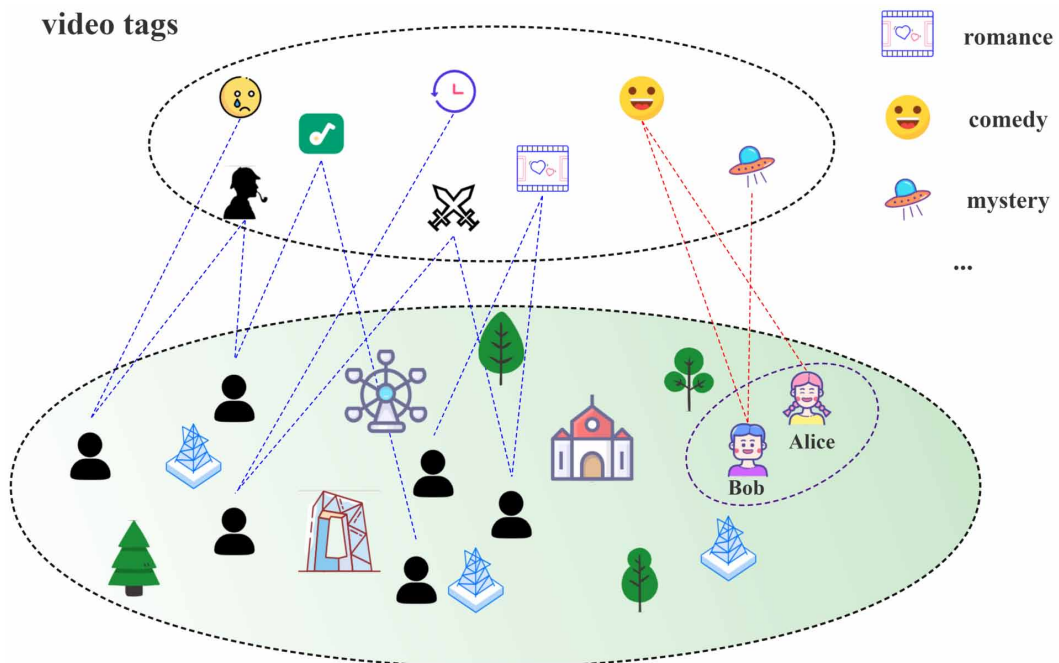
This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

When it comes to video recommendation, the attributes of videos are not as explicit as that of other products. Challenges of how to obtain features successfully by certain keywords from videos need to be addressed. As a result, video recommendation has become a research hotspot.

In mobile edge environment, there will be a lot of interactions between users. And these data contain a wealth of information, which is of great significance to helping feature extraction and video recommendation. Suppose a scenario to push videos to customers/users, we can make recommendations based on the characteristics of neighbors. Videos can be classified into different types such as romance, comedy and mystery. As shown in Figure 1, both users, *Bob* and *Alice*, like to watch comedy, then *Bob* can be defined as *Alice*'s neighbor. According to the preferences of neighbors, videos that have not been watched by *Alice*, such as mystery, can be recommended to her.

Traditional video recommendation methods, such as content-based video recommendation (Dong et al., 2018; Ramadhan & Musdholifah, 2021; Subercaze et al., 2016), collaborative filtering video recommendation (Di Yu & Chen, 2020; Shen et al., 2020) and hybrid video recommendation (Pérez-Marcos et al., 2020; Yan et al., 2015; Zhou et al., 2019). Content-based recommendation methods use video content to predict the preferences of target users, recommending videos to users that are similar to what the user previously liked or watched. Collaborative filtering recommendation methods use user feedback information, such as previous ratings and viewing history, to predict user preferences. Hybrid recommendation methods combine user feedback and consumed video content to improve recommendations. Those methods only consider the simple interactive relationships that cannot mine users' comprehensive interest representations (Stoica & Chaintreau, 2019; Xu et al., 2021; Yang et al., 2019; Yang et al., 2021). Some methods (Cai et al., 2022; Pingali et al., 2022) improve the performance of model by mining the information in the video data, but the excessive computational overhead makes it unsuitable for some low latency scenarios, such as mobile edge environments. In recent years, some studies have used deep learning to construct users' features. J. Chen et al. (2017) proposed two attention models to construct user's features at component-level and item-level, and

Figure 1. Scenarios of user Interaction in mobile edge environment



Chen et al. (2018) proposed an attention model to combine users' features at category-level and item-level. However, such recommendation systems may lead to poor recommendation quality due to the following reasons:

1. Most traditional video recommendation methods regard the relationship between user and neighbor to be linear and ignore higher-order connectivity among users, and cannot obtain the user feature based on neighborhood well, which leads to poor recommendation quality.
2. Most traditional video recommendation methods mine users' single-feature from single-source data, which are usually sparse and cannot obtain valid users' interest representations.

In view of the above analysis, this paper proposes a multi-feature video recommendation method based on hypergraph convolution (MVRHC). In real video recommendation, users often have interactive relationships with multiple neighbors, and simply treating this complex high-order relationship as a pairwise relationship will inevitably lead to the loss of valuable information. A natural extension is to rely on hypergraphs, where hyperedges can accurately model high-price relationships among users. In addition, multi-feature is the aggregation of information from multi-source data, which makes the representation learned by the model more complete. Data from multiple sources are semantically interrelated and provide complementary information to each other, which is beneficial to alleviate the data sparsity problem and obtain more reliable predictions. Specifically, the authors first use hypergraph convolution to learn neighborhood-level feature of users by modeling higher-order correlations among users. Then, attention mechanism is used to learn user item-level feature according to the user's historical interaction record with the video. Neighborhood-level feature and item-level feature are fused to obtain users' multi-feature representations. Video recommendation is finally implemented based on users' multi-feature representations.

Main contributions of this paper are summarized as follows:

1. This paper proposes a video recommendation method based on hypergraph convolution which combines multi-feature representations of users.
2. This paper uses hypergraph convolution for modeling the high-order correlations among users to obtain user neighborhood-level feature. Moreover, attention mechanism is used to obtain user item-level feature to reflect the different importance of videos.
3. Experimental results on two real-world datasets show that the proposed method achieves better recommendation performance than those baseline methods.

## RELATED WORK

### Recommendation Methods Based on Hypergraph

Hypergraph is a generalization of graph which uses hyperedges to connect multiple vertexes. The recommendation methods based on hypergraph are similar to those of graph. However, hypergraph can provide more information and reflect the high-order correlations between vertexes (Gao et al., 2019; Zhao et al., 2018). On the basis of traditional hypergraph learning, hypergraph is gradually combined with deep learning, which is often used to solve problems such as classification and image processing. Hypergraph neural network was first used to deal with complex interactions between data (Feng et al., 2019; Jiang et al., 2019), and it was also used in recommendation in recent studies. Jiang et al. (2019) proposed a dynamic hypergraph construction method, which adopts k-NN method to generate basic hyperedge and extends adjacent hyperedge set by clustering algorithm. By dynamic hypergraph construction method, local and global relations will be extracted. Wu et al. (2019) proposed a collaboration matrix factorization model CMF that combines projection methods with convolutional matrix factorization to extract the collaboration between rating-based latent factors

and review-based latent factors. Ji et al. (2020) proposed DHCF model, which combines hypergraph and Collaborative filtering to learn the high-order correlations between users and items. Xia et al. (2021) proposed a dual channel hypergraph convolutional network DHCN model, which modeled session data as a hypergraph to extract user preference embeddings and integrated a self-supervised task into the training of network to enhance hypergraph modeling and improve the recommendation task. Yu et al. (2021) proposed a multi-channel hypergraph convolutional network to improve social recommendation by leveraging high-order user relations.

## Multi-Feature Representations

Multi-feature is the aggregation of information from multi-source data, which makes the representation learned by the model more complete. There have been many studies on multi-feature representations. AFM (Xiao et al., 2017) used the attention mechanism to improve factorization machines to distinguish the importance of feature interaction. ACF (J. Chen et al., 2017) used collaborative filtering based on the attention mechanism to model features at item-level and component-level. THACIL (Chen et al., 2018) used the temporal hierarchical attention model to construct user features in category-level and item-level. LS-PLM (Gai et al., 2017) first used the embedding layer to process the sparse initial input and then uses the transfer function to obtain the relationships between features. DIN (Zhou et al., 2018) is a deep interest network that adaptively expresses users' interest representations through the correlation of historical behaviors. DAFC (Yin et al., 2020) uses auto-encoder and clustering methods to mitigate overfitting in feature extraction. MUIR (Chen et al., 2020) used a deep network to extract multi-feature representations of users. However, these methods still do not make good use of the high-order correlations to represent user preferences.

## MULTI-FEATURE VIDEO RECOMMENDATION BASED ON HYPERGRAPH CONVOLUTION

This paper proposes a multi-feature video recommendation based on hypergraph convolution for mobile edge environment. As illustrated in Figure 2, MVRHC consists of four parts: 1) Feature extraction; 2) Neighborhood-level feature through hypergraph convolution; 3) Item-level feature through attention mechanism; 4) User-side feature representation and prediction.

This paper uses bold uppercase letters to denote a matrix and bold lowercase letters to denote a vector in this paper. To compress user  $i$ 's tag-aware feature vector and video  $j$ 's tag-aware feature vector into a low-dimensional feature space,  $\mathbf{u}_i^{tag}$  and  $\mathbf{v}_j^{tag}$  are fed into MLP with shared parameters. Then, user neighborhood-level feature  $\mathbf{x}_u^N$  ( $u$  means user-side and  $N$  means neighborhood-level) is obtained through hypergraph convolution, and user item-level feature  $\mathbf{x}_u^V$  ( $u$  means user-side and  $V$  means item-level) is obtained through the attention mechanism. The final feature  $\mathbf{x}_u^*$  on the user side is obtained by fusing multiple features through the attention mechanism. Finally, video recommendation is implemented based on the probability obtained by the inner product of the user final feature vector and the video feature vector.

Frequently used notations in this paper are shown in Table 1.

### Feature Extraction

First, a tuple  $F = (U, V, T)$  is used to represent user set  $U$ , video set  $V$  and tag set  $T$ . Specifically, in the user-tag feature, user feature vector is  $u_i^{tag} = (n_1^{ui}, n_2^{ui}, \dots, n_p^{ui})$ , where  $n_q^{ui}$  is the number of times that user  $i$  annotates videos with tag  $q$ . Similarly, in the video-tag feature, video feature vector is  $v_j^{tag} = (n_1^{vj}, n_2^{vj}, \dots, n_p^{vj})$ , where  $n_q^{vj}$  is the number of times that video  $j$  is annotated with tag  $q$  by

Figure 2. Multi-feature video recommendation based on hypergraph convolution for mobile Edge Environment

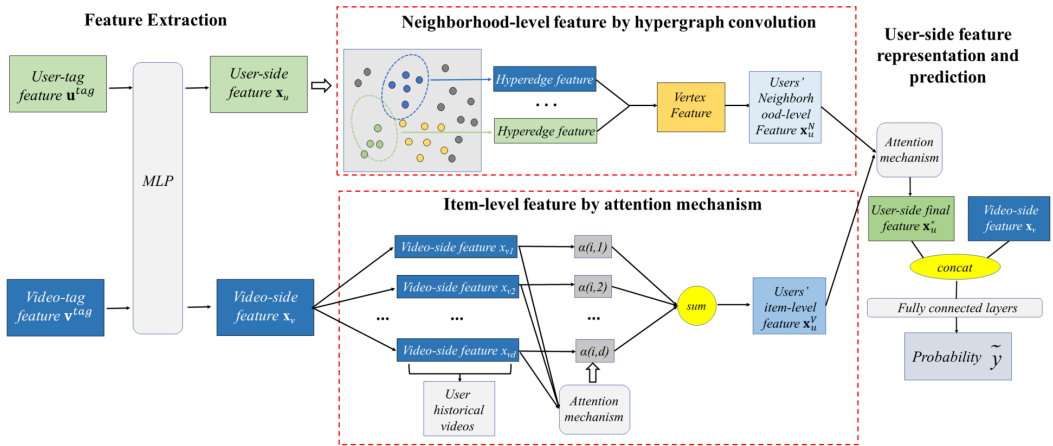


Table 1. Frequently used notations

Notation	Definition
$U$	User set
$V$	Video set
$T$	Tag set
$u_i^{tag}$	User $i$ 's tag-aware feature representation
$v_j^{tag}$	Video $j$ 's tag-aware feature representation
$x_u(i)$	User-side feature after MLP ( $u$ means user-side)
$x_v(j)$	Video-side feature after MLP ( $v$ means video-side)
$H_G$	Hypergraph $G$ 's incidence matrix
$x_u^V$	User-side item-level feature ( $u$ means user-side and $V$ means item-level)
$x_u^N$	User-side neighborhood-level feature ( $u$ means user-side and $N$ means neighborhood-level)
$x_u^*$	User-side final feature ( $u$ means user-side)

users. For example, if the number of times that user  $i$  annotates videos with tag 1, 2, 3, 4, 5, ...,  $p$  is 0, 0, 1, 0, 3, ..., 0, then the initial feature of user  $i$  is  $u_i^{tag} = (0, 0, 1, 0, 3, \dots, 0)$ .

MLP is used to compress sparse user-tag feature vector  $u_i^{tag}$  and video-tag feature vector  $v_j^{tag}$  from high-dimensional space to low-dimensional space. and by sharing parameters, map them to the same space. Take the user side as an example,  $w^{(l)}$  represents the  $l$ -th layer shared parameter between

the user side and the video side, and use ReLU as the activation function. The input and output of the first layer can be expressed as follows:

$$x_u^{(1)}(i) = \text{ReLU}\left(w^{(1)}u_i^{\text{tag}} + b^{(1)}\right) \quad (1)$$

For each layer of MLP, the input and output can be expressed as follows:

$$x_u^{(l)}(i) = \text{ReLU}\left(w^{(l)}x_u^{(l-1)}(i) + b^{(l)}\right) \quad (2)$$

Therefore, the feature representation  $x_u$  on the user side is obtained. Similarly, the feature representation on the video side is  $x_v$ .

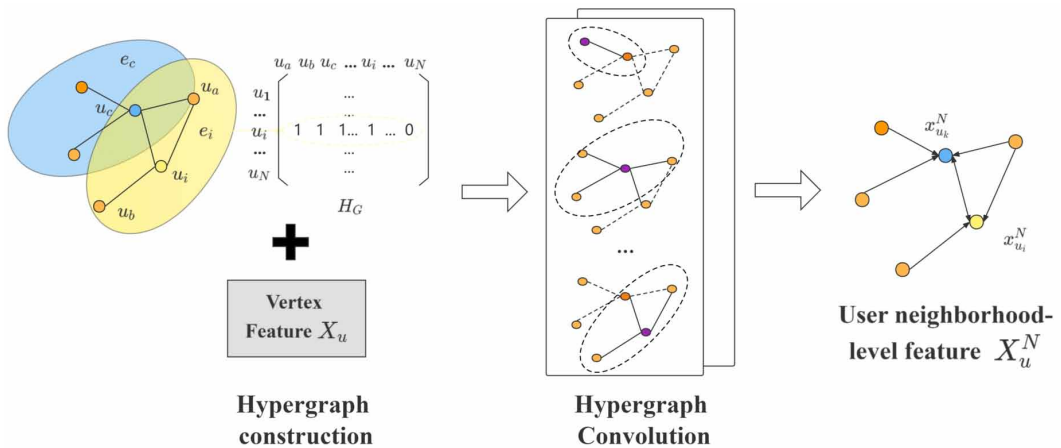
### Neighborhood-Level Feature Through Hypergraph Convolution

According to the idea of the user-based collaborative filtering method, the features of neighbors with high similarity can reflect the features of the target user. As illustrated in Figure 3, the authors use the hypergraph convolution method to model higher-order connectivity among users and extract neighborhood-level feature.

#### Hypergraph Construction

Unlike ordinary graph, a hyper-edge in a hypergraph can contain three or more vertexes. Hypergraph can be expressed as  $G = (V, E)$ , where  $V$  is the set of vertexes and  $E$  is the set of hyperedges. The incidence matrix  $H \in R^{|V| \times |E|}$  can be used to represent a hypergraph. In this paper, the vertexes of hypergraph represent users feature  $x_u$ . The Euclidean distance is calculated for every two users, and for each vertex, regard the set of its nearest  $k$  neighbors as a hyperedge. Then, the incidence matrix  $H_G$  of hypergraph  $G$  is obtained. Each element  $h(v, e)$  indicate whether vertex  $v$  belongs to hyperedge  $e$ :

Figure 3. Neighborhood-level feature through hypergraph convolution



$$h(v, e) = \begin{cases} 1 & (v \in e) \\ 0 & (v \notin e) \end{cases} \quad (3)$$

Therefore,  $\mathbf{H}_G$  can be used to represent the user hypergraph.

The overall algorithm of the hypergraph construction is presented in Algorithm 1. In lines 1-9, the algorithm constructs hyperedge set  $E$  based on the input user feature  $x_u$ . Specifically, the algorithm computes the Euclidean distance between users in line 3, and constructs hyperedges through each user's top  $k$  similar users in lines 4-8. In lines 10-18, the algorithm constructs the hypergraph  $G$  and obtain the incident matrix  $\mathbf{H}_G$  of  $G$ . In general, it is to select the most similar  $k$  users from the user set to construct a hyperedge for each user, so the complexity of Algorithm 1 is  $O(n^2)$ , where  $n$  is the size of user set  $U$ .

Table 2. Hypergraph construction algorithm

Algorithm 1. Hypergraph Construction
<b>Input:</b> feature $x_u$ , user set $U$ , hyperparameter $k$
<b>Output:</b> hypergraph incident matrix $\mathbf{H}_G$
<b>Function:</b> compute Euclidean distance $dis()$ ; insert object into list $insert()$ ; sort the list $sort()$ ; get the top $k$ objects of the list $top-k()$ ;
1 <b>for</b> $i$ in $U$ <b>do</b>
2 <b>for</b> $j$ in $U$ <b>do</b>
3 $d(i, j) = dis(x_u(i), x_u(j))$
4 $S_i.insert(d(i, j))$
5 $D_i = sort(S_i)$
6 <b>end</b>
7 $e_i = top-k(D_i, k)$
8 $E.insert(e_i)$
9 <b>end</b>
10 <b>for</b> $e$ in $E$ <b>do</b>
11 <b>for</b> $i$ in $U$ <b>do</b>
12 <b>if</b> $i \in e$ <b>then</b>
13 $h(i, e) = 1$
14 <b>else</b>
15 $h(i, e) = 0$
16 $\mathbf{H}_G.insert(h)$
17 <b>end</b>
18 <b>end</b>
19 <b>return</b> $\mathbf{H}_G$

## Hypergraph Convolution

Similar to graph convolution, hypergraph convolution is to update vertex features through the process of vertex-hyperedge-vertex. Referring to the spectral hypergraph convolution proposed in (Feng et al., 2019), the hypergraph convolution can be expressed as follows:

$$X_u^{(l+1)} = \sigma(D_V^{-\frac{1}{2}} H_G D_E^{-1} H_G^T D_V^{-\frac{1}{2}} X_u^{(l)} \Theta^{(l)} + X_u^{(l)}) \quad (4)$$

$\Theta^{(l)}$  represents the learnable parameter matrix on the  $l$ -th layer.  $D$  is a degree matrix, and the elements on the diagonal are the degrees of  $\mathbf{H}_G$ . The multiplication operation with  $H_G^T$  represents the aggregation from vertex feature to hyperedge feature, and the multiplication operation with  $\mathbf{H}_G$  represents the aggregation from the hyperedge feature to vertex feature. In addition, according to the idea of ResNet (He et al., 2016), in each layer of the hypergraph convolution, the previous features will be preserved by the add operation with  $X_u^{(l)}$ . After passing  $x_u$  through  $l$  hypergraph convolutional layer, taking the final output as user neighborhood-level feature  $X_u^N$ .

## Item-Level Feature Through Attention Mechanism

According to users' historical interaction records with videos, the features of videos can be used to express the features of users to infer users' interest representations. In order to obtain the influence weights of different video features on user interest representation, the authors introduce the attention mechanism (Bahdanau et al., 2014). According to the user-video rating matrix, the historical interactive video set  $V_i$  for each user  $i$  is obtained. Then user  $i$ 's interest representations on  $V_i$  is constructed through the attention mechanism.

The attention score calculated by the two-layer network is calculated as follows:

$$a(i, j) = w_1^T \tanh(w_1^u x_{ui} + w_1^v x_{vj} + b_1) + c_1 \quad (5)$$

Table 3. Hypergraph convolution algorithm

Algorithm 2. Hypergraph Convolution
<b>Input:</b> feature $x_u$ , hypergraph incident matrix $\mathbf{H}_G$
<b>Output:</b> user neighborhood-level feature $X_u^N$
<b>Function:</b> activation function $ReLU$
<pre> 1 <math>X_u^{(0)} = X_u</math> 2 for <math>l = 1, 2, \dots, L</math> do 3 <math>O_u^{(l+1)} = HGConv(X_u^{(l)})</math> 4 <math>X_u^{(l+1)} = ReLU(O_u^{(l+1)})</math> </pre>
5 end
6 $X_u^N = X_u^{(L)}$
7 return $X_u^N$



$$\alpha(i, j) = \text{soft max}(a(i, j)) = \frac{\exp(a(i, j))}{\sum_{k \in V_i} \exp(a(i, k))} \quad (6)$$

where  $w_1^u, w_1^v, \mathbf{b}_1$  are the parameters of the first layer, and  $w_1^T, c_1$  are the parameters of the second layer.  $a(i, j)$  represents the weight of video  $j$  relative to user  $i$ .

The item-level feature by historical interaction records is calculated as follows:

$$x_{ui}^v = \sum_{k \in V_i} \alpha(i, k) x_{vk} \quad (7)$$

### User-Side Feature Representation and Prediction

In order to integrate the user-side neighborhood-level feature  $x_u^N$  and item-level feature  $x_u^V$ , a self-attention mechanism is used to calculate weights of multi-feature.

$$b(i, k) = w_2^T \tanh(w_2 x_{ui}^k + c_2) \quad (8)$$

$$\beta(i, k) = \text{soft max}(b(i, k)) = \frac{\exp(b(i, k))}{\sum_{k=n,v} \exp(b(i, k))} \quad (9)$$

where  $w_2, c_2$  are the parameters of the attention mechanism, and  $x_{ui}^k$  is neighborhood-level feature  $x_u^N$  or item-level feature  $x_u^V$ . The final feature representation  $x_u^*$  on the user side is obtained.

$$x_u^* = \sum_{k=n,v} \beta(i, k) x_{ui}^k \quad (10)$$

In order to make full use of the interaction between features, we first concatenate the final feature vector  $x_u^*$  on the user side and the video feature vector  $x_v$  into  $z = [x_u^*, x_v^*]$ . Then, we use full connection layer and sigmoid function to obtain user  $i$ 's predicted probability of the video  $j$ .

$$\tilde{y} = \text{sigmoid}(w^T z + b) \quad (11)$$

The cross-entropy loss function is used as the loss function for model training, and  $TS$  is the training sample set.

$$L = -\sum^{TS} y \log \tilde{y} + (1 - y) \log (1 - \tilde{y}) \quad (12)$$

### Experiment

In this section, the authors verify the feasibility and effectiveness of the model proposed in this paper through experiments. It is worth noting that users' interaction in mobile edge environment is abstracted as neighbor relations, and the features of target users are obtained through hypergraph convolution.

The authors first introduce the datasets, experimental environment, parameter settings, baseline methods and evaluation metrics. Then, the experimental results are compared and analyzed in detail.

## Datasets

In order to test the effectiveness of MVRHC, MovieLens-Latest and MovieLens-10M are adopted as datasets for experiments. MovieLens is a dataset of multiple versions of interactive information such as ratings, reviews, tags of users on the IMDB provided by the University of Minnesota. MovieLens-Latest and MovieLens-10M are two different versions of MovieLens. The statistics of the above two datasets are shown in Table 4:

## Experimental Environment and Parameter Settings

The experimental hardware environment is CPU Intel(R) Core(TM) i9-9980XE @ 3.00GHz, memory 128GB. Using Python programming language, PyTorch deep learning framework and Numpy scientific computing library.

The Adam optimizer is adopted for training. The number of MLP layers is 3, the number of neurons in each layer is 512, 256, 128, the number of hypergraph convolution layers is 2, the dropout rate of the dropout layer is 0.5, and the batch size is 128.

## Baseline Methods and Evaluation Metrics

The method proposed in this paper will be compared with four baseline methods:

- (1) **NCF** (He et al., 2017): A collaborative filtering algorithm which uses neural networks to obtain features for recommendation.
- (2) **CFA** (Zuo et al., 2016): This algorithm uses deep neural networks to extract deep features from tags, and combines them with user-based collaborative filtering for recommendation.
- (3) **TNAM** (Huang et al., 2020): A tag-aware neural attention model for Top-K recommendation. This method considers both user-item interaction and tag information, and uses neural attention networks for recommendation.
- (4) **DHCF** (Ji et al., 2020): A dual channel hypergraph collaborative filtering method. This method uses hypergraph convolution on the user-side and the item-side for recommendation.

This paper uses the following evaluation metrics:

**HR@K**: Hit ratio for Top-K recommendation:

$$HR@K = \frac{NumberOfHits@K}{GT} \quad (13)$$

Table 4. Statistics of datasets

Dataset	Users	Videos	Tags	Interactions
MovieLens-Latest	610	9742	3684	100836
MovieLens-10M	71567	10681	95580	10000054

where  $GT$  is the size of test set, and  $\text{NumberOfHits@K}$  is the number of successfully predicted items in the Top-K recommendation.

**nDCG@K**: Normalized discounted cumulative gain for Top-K recommendation:

$$DCG = \sum_{i=1}^K \frac{rel_i}{\log_2(i+1)} \quad (14)$$

$$nDCG @ K = \frac{DCG}{IDCG} \quad (15)$$

where  $rel_i$  represents the degree of correlation of the  $i$ -th recommendation,  $DCG$  represents discounted cumulative gain, and  $IDCG$  represents the maximum  $DCG$  value under ideal conditions.

**MRR@K**: Mean reciprocal rank for Top-K recommendation:

$$MRR @ k = \frac{1}{m} \sum_u \frac{1}{rank_i} \quad (16)$$

where  $rank_i$  refers to the rank position of the first relevant item for target user  $u$  in the recommended list.

## Experimental Results

Experimental results are analyzed from four aspects: *a*) Overall analysis; *b*) compare and analyze MVRHC with variant models; *c*) compare and analyze MVRHC on training datasets of different sizes; and *d*) compare and analyze MVRHC with different parameters.

### Overall Analysis

In this section, MVRHC proposed in this paper is compared with baseline methods and analyzed based on the experimental results of the datasets. From Table 5 and Table 6, it can be seen that the overall performance of MVRHC is better than baseline methods, which verifies the effectiveness of the proposed method. The improvements over the best baseline are 4.23% on MovieLens-Latest and 5.08% on MovieLens-10M. Specifically, since NCF and CFA only use neural network to extract static features of users and items, without considering the relationship between users and the historical interactions of users, these two baseline methods are less effective, while MVRHC uses hypergraph convolution to model users' high-order correlations and extract user neighborhood-level feature. MVRHC adopts the idea of user-based collaborative filtering and constructs feature through similar users. The attention mechanism is used to obtain the influence of different videos in the users' historical interaction records so as to construct the user item-level feature. Therefore, the method proposed in this paper is superior to other baseline methods. In addition, some methods improve model performance by mining video data but also bring a significant computational overhead (Carreira et al., 2017). Our model combines the characteristics of hypergraph convolution and multi-features to obtain recommendation performance by mining textual information, which is more suitable for scenarios with high real-time requirements, such as mobile edge environments.

### Compare With Variant Models

Compare and analyze two variants of the model:

Table 5. Performance on MovieLens-latest

Method	HR@5	HR@10	nDCG@5	nDCG@10	MRR
NCF	0.4322	0.4706	0.4173	0.4431	0.3891
CFA	0.4678	0.5053	0.4429	0.4864	0.4123
TNAM	0.5589	0.5928	0.4812	0.5123	0.4527
DHCF	0.5827	0.6245	0.5070	0.5429	0.4671
MVRHC	0.6106	0.6509	0.5241	0.5647	0.4789

Table 6. Performance on MovieLens-10M

Method	HR@5	HR@10	nDCG@5	nDCG@10	MRR
NCF	0.4006	0.4532	0.3879	0.4240	0.3671
CFA	0.4410	0.4872	0.4130	0.4703	0.3986
TNAM	0.5451	0.5838	0.4597	0.4964	0.4213
DHCF	0.5646	0.6065	0.4672	0.5118	0.4329
MVRHC	0.5882	0.6373	0.4858	0.5295	0.4573

- (1) **MVRHC-1**: No hypergraph convolution is used;
- (2) **MVRHC-2**: No attention mechanism is used.

According to the experimental results shown in Figure 4-5, these two variant models show poor recommendation performance. This is because the hypergraph convolution can extract neighborhood-level feature based on the high-order correlations of vertexes so as to feed back the features of similar users to the target user. The attention mechanism can learn the weights of different historical interactions on the user so as to reflect the influence of different videos on the construction of user preferences. The experimental results reflect the necessity and rationality of these two parts of the model proposed.

### *Compare on Training Datasets of Different Sizes*

The authors select 5%, 10%, 20%, 40%, 60%, and 80% of MovieLens-10M as the training dataset and analyze the impact of data sparsity on performance. It can be seen in Figure 6 that MVRHC is less affected by data sparsity than other baseline methods. Because MVRHC uses hypergraph convolution and attention mechanism to obtain high-order correlations and multi-feature information, which make the representation learned by the model more complete. These semantically related information will provide complementary information to each other, alleviating the sparsity problem of single-feature representation.

### *Compare on Model Parameter*

The authors analyze the impact of hyperparameter  $k$  on recommendation performance, where  $k$  represents the number of the top- $k$  nearest neighbors selected by each vertex. As can be seen from Figure 7 that the performance improves with the increase of  $k$ . However, after the value of  $k$  is incremented to 10, the experimental results show poor performance. This is because when the value of  $k$  is small, neighborhood-level feature of users cannot be well represented. Features of neighbors may be similar to that of target user on some dimensions. However, when there are few neighbor

Figure 4. HR@K and nDCG@K of variant models on MovieLens-latest

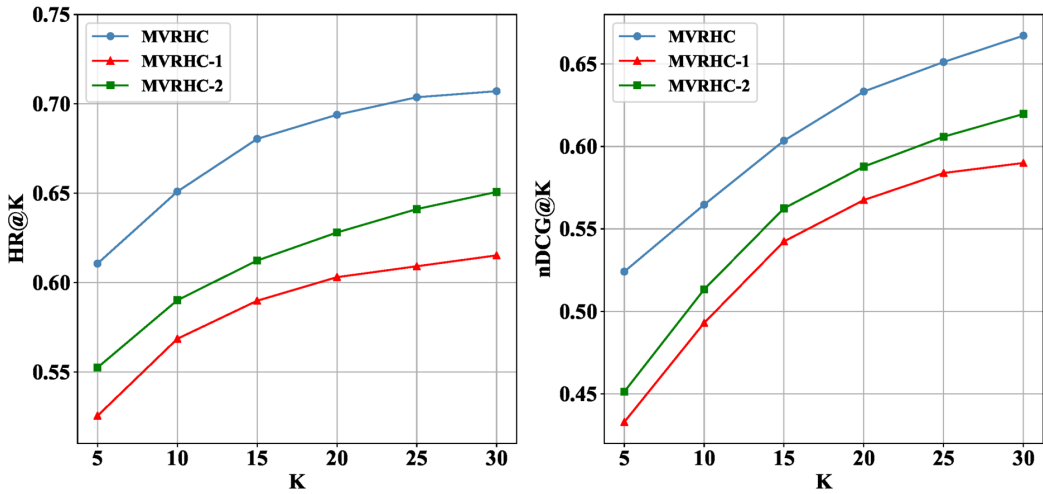
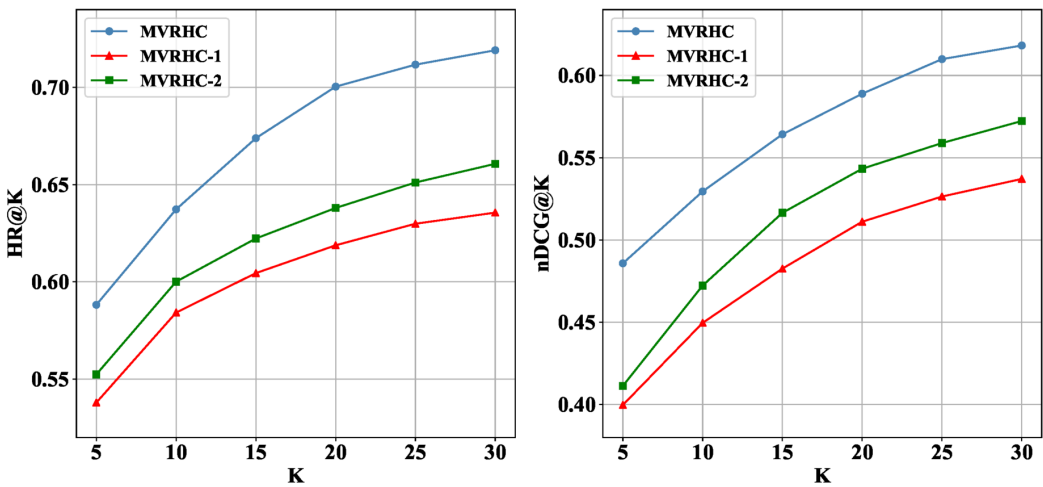


Figure 5. HR@K and nDCG@K of variant models on MovieLens-10M



nodes available, it is not enough to enrich the features of the target user. On the other hand, a larger value of  $k$  means that features of low-similarity neighbors are included in the hyperedge, which will introduce noisy information.

## DISCUSSION

In this paper, we propose a multi-feature video recommendation method based on hypergraph convolution. The theoretical and practical contributions of our work are listed as follows:

### Theoretical Contributions

Our model makes a systematic investigation on the relationships among users for recommendation system. By introducing hypergraph convolution to model the nonlinear relationship among users,

Figure 6. nDCG@K on MovieLens-10M with different sizes

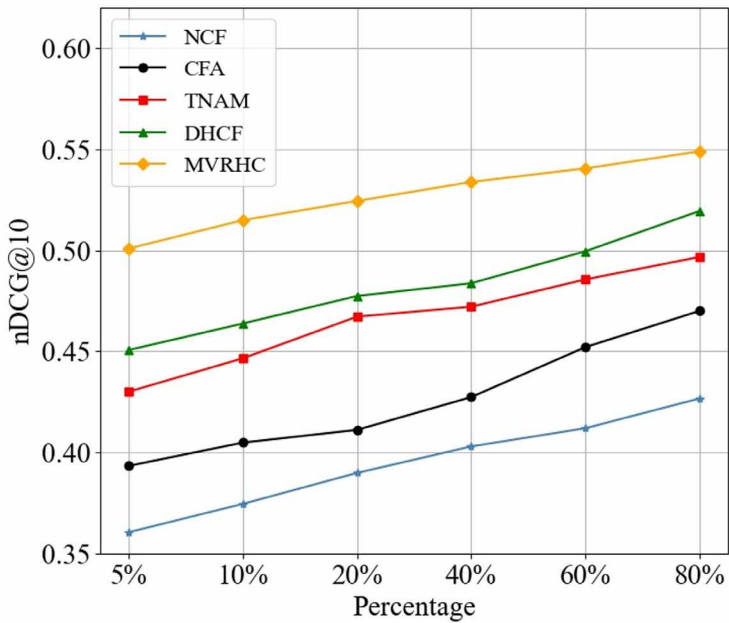
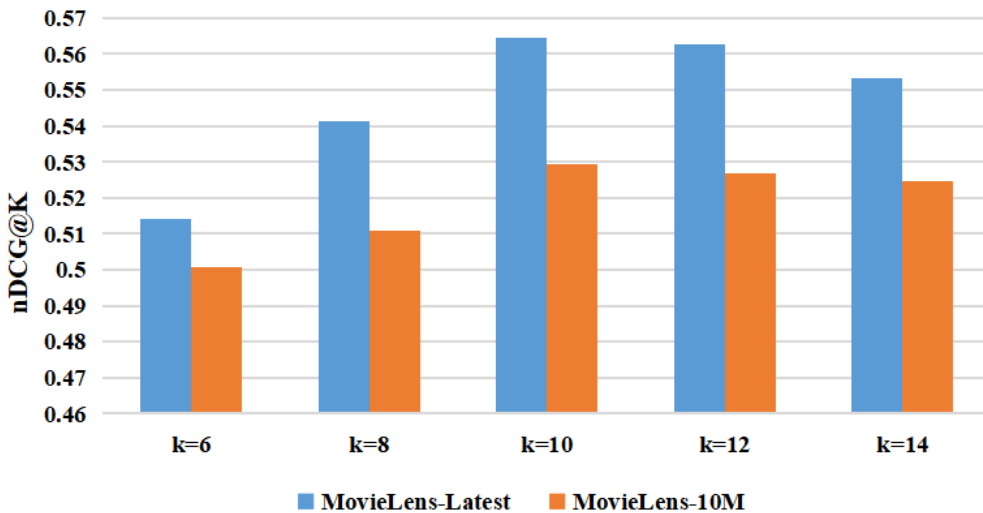


Figure 7. nDCG@K on MovieLens-Latest and MovieLens-10M with different values of k



the preference features in users' interaction records are obtained, which extends the research from linear to nonlinear on users' social relationship modeling. Additionally, our model takes the semantic correlation between different types of source data into consideration, including user-user interaction information and user-item historical interaction information. By introducing multi-feature method to fuse information, the feature representation has been enriched in semantics, which provides an alternative solution for the problem of data sparsity in recommendation system.

## Practical Contributions

Our model aims at achieving a good balance between recommendation performance and computational overhead in mobile edge environment. Due to the inherent ambiguity of text information, it may have a negative impact on the performance of traditional recommendation models based on text mining. Some methods (Cai et al., 2022; Pingali et al., 2022) improve the performance of recommendation system by mining video data features, but the model takes significant time to process video data, which cannot meet the low latency requirements of users in mobile edge environment. Our model combines the characteristics of hypergraph convolution and multiple features, and accurate recommendation results are obtained by mining textual information, which has more advantages in practical applications. In addition, our model is designed based on the mobile edge environments, which can handle large amounts of interaction data. The noise information in these interaction data will lead to poor performance for some traditional models. Our model employs deep learning theory and technology to explore the common rules existing in disordered data from the perspective of data governance, thereby obtaining effective video recommendation prediction sequences provides a diverse approach to data management.

## CONCLUSION

This paper proposed a multi-feature video recommendation method based on hypergraph convolution for mobile edge environment. Through hypergraph convolution and attention mechanism, the users' neighborhood-level feature and item-level feature are integrated to obtain the final feature representations, so as to complete video recommendation by multi-feature. Experimental results indicate that MVRHC can reflect high-order correlations between users and integrate multi-feature effectively.

The authors' current work focuses on how to improve the performance of video recommendation through mining interaction information between users and neighbors. However, the authors will investigate on fine-grained feature for the representation of user preferences, such as real-time property and location, to model user dynamic preferences for video recommendation in future research.

## CONFLICT OF INTEREST

The authors declare that there is no conflict of interest regarding the publication of this paper.

## FUNDING STATEMENT

This research was supported by National Natural Science Foundation of China (62272243) and Jiangsu Key Laboratory of Big Data Security & Intelligent Processing.

## REFERENCES

- Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
- Cai, D., Qian, S., Fang, Q., Hu, J., & Xu, C. (2022, October). Adaptive Anti-Bottleneck Multi-Modal Graph Learning Network for Personalized Micro-video Recommendation. In *Proceedings of the 30th ACM International Conference on Multimedia* (pp. 581-590). ACM. doi:10.1145/3503161.3548420
- Carreira, J., & Zisserman, A. (2017). Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 6299-6308). ACM. doi:10.1109/CVPR.2017.502
- Chen, J., Zhang, H., He, X., Nie, L., Liu, W., & Chua, T. S. (2017, August). Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval* (pp. 335-344). ACM. doi:10.1145/3077136.3080797
- Chen, X., Liu, D., Xiong, Z., & Zha, Z. J. (2020). Learning and fusing multiple user interest representations for micro-video and movie recommendations. *IEEE Transactions on Multimedia*, 23, 484–496. doi:10.1109/TMM.2020.2978618
- Chen, X., Liu, D., Zha, Z. J., Zhou, W., Xiong, Z., & Li, Y. (2018, October). Temporal hierarchical attention at category-and item-level for micro-video click-through prediction. In *Proceedings of the 26th ACM international conference on Multimedia* (pp. 1146-1153). ACM. doi:10.1145/3240508.3240617
- Di Yu, R. C., & Chen, J. (2020). Video Recommendation Algorithm based on Knowledge Graph and Collaborative Filtering. *International Journal of Performability Engineering*, 16(12), 1933.
- Dong, J., Li, X., Xu, C., Yang, G., & Wang, X. (2018, October). Feature re-learning with data augmentation for content-based video recommendation. In *Proceedings of the 26th ACM international conference on Multimedia* (pp. 2058-2062). ACM. doi:10.1145/3240508.3266441
- Feng, Y., You, H., Zhang, Z., Ji, R., & Gao, Y. (2019, July). Hypergraph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01), 3558–3565. doi:10.1609/aaai.v33i01.33013558
- Gai, K., Zhu, X., Li, H., Liu, K., & Wang, Z. (2017). Learning piece-wise linear models from large scale data for ad click prediction. *arXiv preprint arXiv:1704.05194*.
- Gao, T., Asoodeh, S., Huang, Y., & Evans, J. (2019, July). Wasserstein soft label propagation on hypergraphs: Algorithm and generalization error bounds. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01), 3630–3637. doi:10.1609/aaai.v33i01.33013630
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778). IEEE.
- He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017, April). Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web* (pp. 173-182). IEEE. doi:10.1145/3038912.3052569
- Huang, R., Wang, N., Han, C., Yu, F., & Cui, L. (2020). TNAM: A tag-aware neural attention model for Top-N recommendation. *Neurocomputing*, 385, 1–12. doi:10.1016/j.neucom.2019.11.095
- Ji, S., Feng, Y., Ji, R., Zhao, X., Tang, W., & Gao, Y. (2020, August). Dual channel hypergraph collaborative filtering. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2020-2029). ACM. doi:10.1145/3394486.3403253
- Jiang, J., Wei, Y., Feng, Y., Cao, J., & Gao, Y. (2019, August). Dynamic Hypergraph Neural Networks. In *IJCAI* (pp. 2635-2641). doi:10.24963/ijcai.2019/366
- Pérez-Marcos, J., Martín-Gómez, L., Jiménez-Bravo, D. M., López, V. F., & Moreno-García, M. N. (2020). Hybrid system for video game recommendation based on implicit ratings and social networks. *Journal of Ambient Intelligence and Humanized Computing*, 11(11), 4525–4535. doi:10.1007/s12652-020-01681-0



- Pingali, S., Mondal, P., Chakder, D., Saha, S., & Ghosh, A. (2022, July). Towards developing a multi-modal video recommendation system. In *2022 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE. doi:10.1109/IJCNN55064.2022.9892382
- Shen, M., Wang, J., Liu, O., & Wang, H. (2020). Expert detection and recommendation model with user-generated tags in collaborative tagging systems. [JDM]. *Journal of Database Management*, 31(4), 24–45. doi:10.4018/JDM.2020100102
- Stoica, A. A., & Chaintreau, A. (2019, May). Hegemony in Social Media and the effect of recommendations. In *Companion Proceedings of The 2019 World Wide Web Conference* (pp. 575-580). doi:10.1145/3308560.3317589
- Subercaze, J., Gravier, C., & Laforest, F. (2016, January). Real-time, scalable, content-based Twitter users recommendation. In *Web Intelligence* (Vol. 14, No. 1, pp. 17-29). IOS Press. doi:10.3233/WEB-160329
- Wu, Z., Liu, H., Xu, Y., & Jing, L. (2019). Collaboration matrix factorization on rate and review for recommendation. [JDM]. *Journal of Database Management*, 30(2), 27–43. doi:10.4018/JDM.2019040102
- Xia, X., Yin, H., Yu, J., Wang, Q., Cui, L., & Zhang, X. (2021, February). Self-supervised hypergraph convolutional networks for session-based recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(5), 4503–4511. doi:10.1609/aaai.v35i5.16578
- Xiao, J., Ye, H., He, X., Zhang, H., Wu, F., & Chua, T. S. (2017). Attentional factorization machines: Learning the weight of feature interactions via attention networks. *arXiv preprint arXiv:1708.04617*. 10.24963/ijcai.2017/435
- Xu, C., Ding, A. S., & Zhao, K. (2021). A novel POI recommendation method based on trust relationship and spatial-temporal factors. *Electronic Commerce Research and Applications*, 48, 101060. doi:10.1016/j.elerap.2021.101060
- Yan, M., Sang, J., & Xu, C. (2015, June). Unified youtube video recommendation via cross-network collaboration. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval* (pp. 19-26). doi:10.1145/2671188.2749344
- Yang, D., Qu, B., Yang, J., & Cudre-Mauroux, P. (2019, May). Revisiting user mobility and social relationships in lbsns: a hypergraph embedding approach. In *The world wide web conference* (pp. 2147-2157). ACM. doi:10.1145/3308558.3313635
- Yang, G., Yang, Q., & Jin, H. (2021). A novel trust recommendation model for mobile social network based on user motivation. *Electronic Commerce Research*, 21(3), 809–830. doi:10.1007/s10660-019-09344-9
- Yin, Y., Cao, Z., Xu, Y., Gao, H., Li, R., & Mai, Z. (2020). QoS prediction for service recommendation with features learning in mobile edge computing environment. *IEEE Transactions on Cognitive Communications and Networking*, 6(4), 1136–1145. doi:10.1109/TCCN.2020.3027681
- Yu, J., Yin, H., Li, J., Wang, Q., Hung, N. Q. V., & Zhang, X. (2021, April). Self-supervised multi-channel hypergraph convolutional network for social recommendation. In *Proceedings of the Web Conference 2021* (pp. 413-424). ACM. doi:10.1145/3442381.3449844
- Zhao, X., Wang, N., Shi, H., Wan, H., Huang, J., & Gao, Y. (2018, April). Hypergraph learning with cost interval optimization. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Zhou, G., Zhu, X., Song, C., Fan, Y., Zhu, H., Ma, X., & Gai, K. (2018, July). Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 1059-1068). ACM. doi:10.1145/3219819.3219823
- Zhou, R., Xia, D., Wan, J., & Zhang, S. (2019). An intelligent video tag recommendation method for improving video popularity in mobile computing environment. *IEEE Access : Practical Innovations, Open Solutions*, 8, 6954–6967. doi:10.1109/ACCESS.2019.2961392
- Zuo, Y., Zeng, J., Gong, M., & Jiao, L. (2016). Tag-aware recommender systems based on deep neural networks. *Neurocomputing*, 204, 51–60. doi:10.1016/j.neucom.2015.10.134

*Haiyan Wang received the Ph.D. degree from the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China, in 2008. She is currently a Professor with the School of Computer Science, Nanjing University of Posts and Telecommunications, where she is also the Deputy Director of the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing and the Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks. Her research interests mainly focus on service computing, trusted computing, and big data intelligent processing technology. Dr. Wang has hosted two National Natural Science Foundation of China projects and received the second prize twice for progress in Jiangsu Science and Technology.*

*Jun Hong is currently pursuing the M.Sc. degree with the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing. His main research interests include deep learning, big data, and recommendation systems.*

*Kaixiang You is a Master Candidate with the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing. His main research interests include deep learning, big data, and recommendation systems.*

*Jian Luo is currently an associate Professor with the School of Computer Science, Nanjing University of Posts and Telecommunications. Her main research interests include deep learning, big data, and recommendation systems.*