An Item Orientated Recommendation from Multi-view Perspective

Qi-Ying Hu[1] Zhi-Lin Zhao[1]
Chang-Dong Wang[1] Jian-Huang Lai[1]

[1]School of Data and Computer Science
Sun Yat-sen University, P. R. China.

The 2016 International Conference on Intelligence Science and Big Data Engineering

Outline

Traditional Recommendation Algorithms

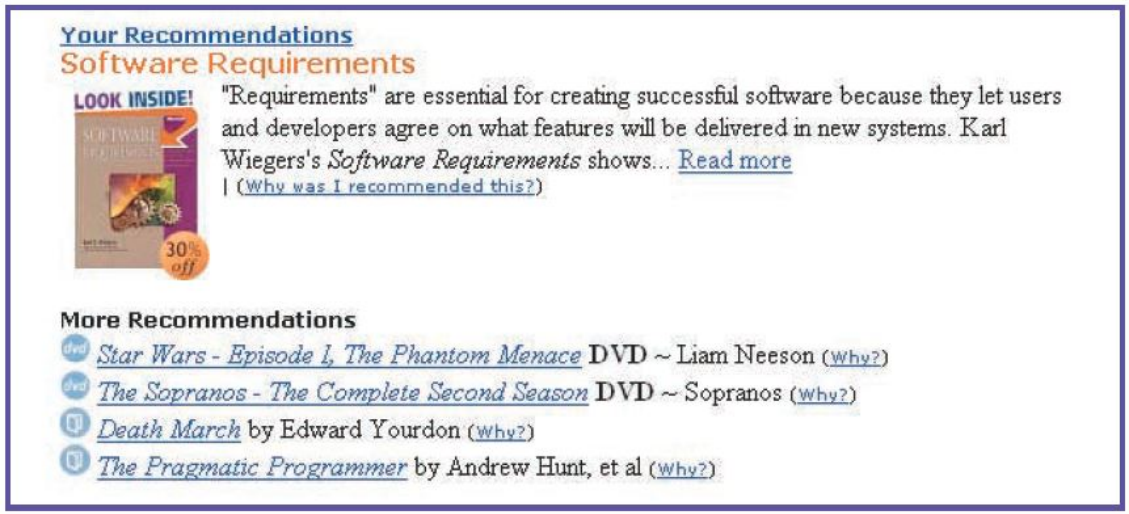

Figure: Recommendation based on user's preference.

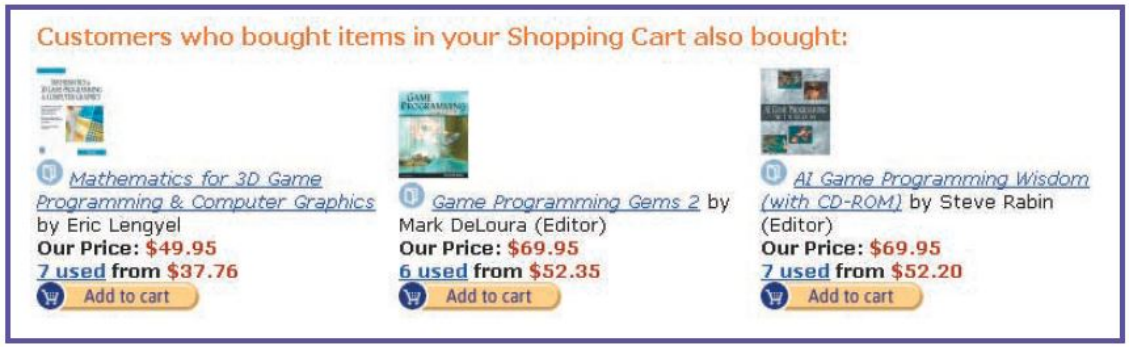

Figure: Recommendation based on user's purchasing history.

Problem Definition

- Assume that, a manufacturer has a limited budget for an item's advertisement. With this budget, it is only possible for him to market this item to the limited number of users.
- How to select the most suitable users that will maximize the advertisement benefits?
- To address the above issue, we present an approach based on the multi-view learning and perform item orientated recommendation.

The Multi-view Model

The rating records of each user are taken as a view where the items rated by the user can be represented as nodes.

- **Within-view Relationship**: The item relationship computed in an individual view, depicting the relationship between items revealed by the single user.
- **Cross-view Relationship**: The edge between items across different views, simultaneously considering both the consistency and diversity of item preferences among different users.

The Multi-view Model

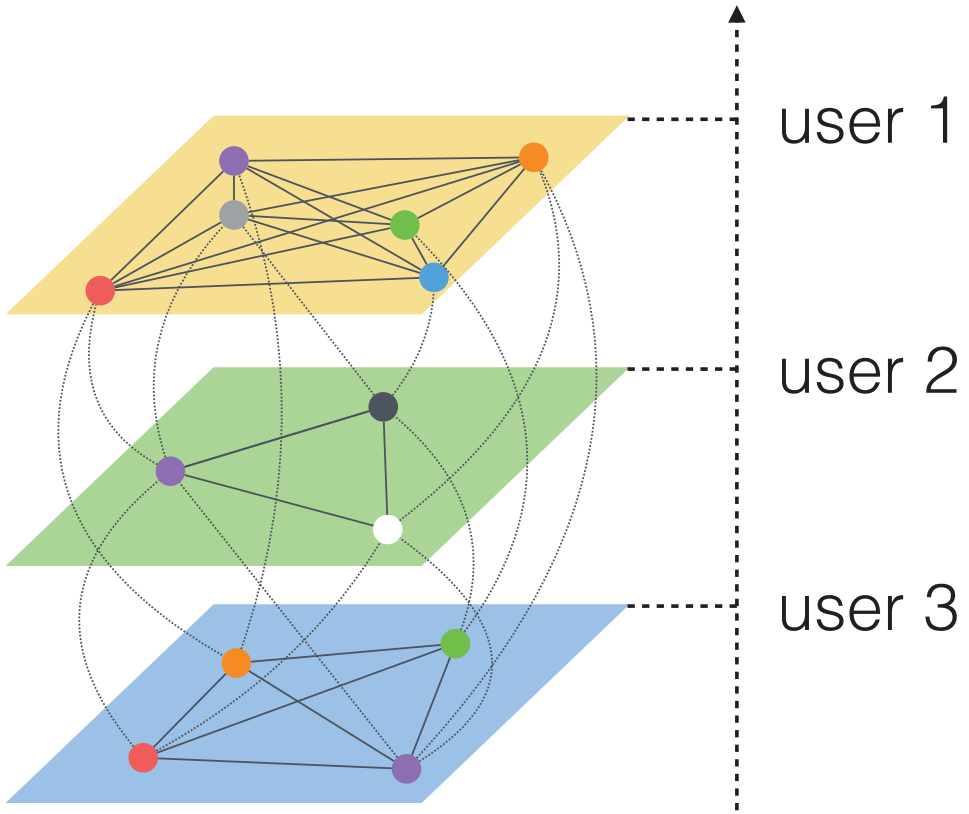

Figure: Within-view relationship and cross-view relationship in multi-view model.

Within-View and Cross-View Relationships

Given a user-item rating matrix $R = [r_{u,i}]_{M \times N}$, whose two dimensions correspond to user and item with sizes M and N, respectively. Each entry $r_{u,i}$ denotes the rating of user u to item i.

1. Within-view relationship:
   Different items i and j corresponding to user u, denoted as $S^u_{i,j}$:

$$S^u_{i,j} = \sqrt{r^2_{u,i} + r^2_{u,j}}. \tag{1}$$

2. Cross-view relationship:
   Item i purchased by user u and item j purchased by user v, denoted as $S^{u,v}_{i,j}$:

$$S^{u,v}_{i,j} = \sqrt{r^2_{u,i} + r^2_{v,j}}. \tag{2}$$

Within-View and Cross-View Relationships

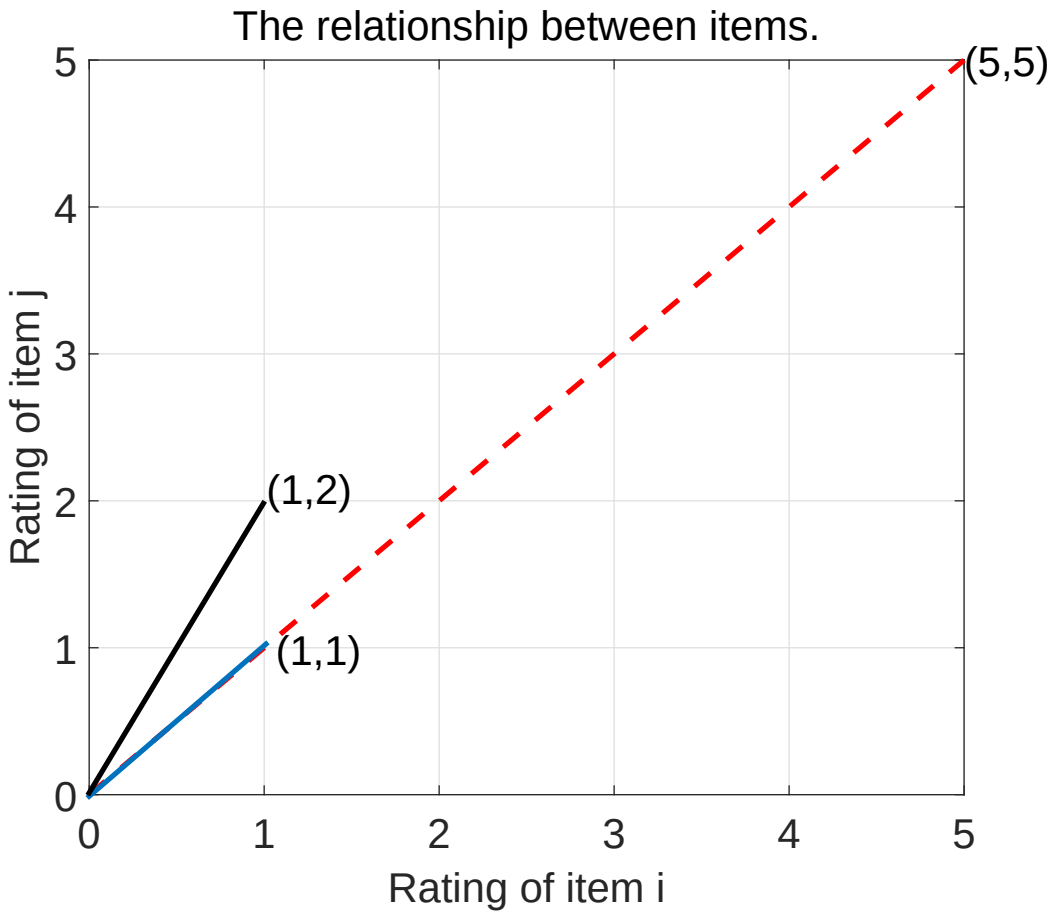

Figure: The relationship between items. We define that only when two items have high ratings and the ratings of them are similar, the value of relationship will be higher and the items are closer.

Objection Function

- The goal of the multi-view model is tolearn the relationship matrix α and the rating difference matrix β.
  - α: The entry $\alpha_{i,j}$, representing the relationship between items, is consistent both in an individual view and multi view.
  - β: The entry $\beta_{u,v}$ represents the rating difference between users.
- The objective function can be written as follows.

$$\begin{aligned}\alpha,\beta = \arg\min_{\alpha,\beta} E = \arg\min_{\alpha,\beta} \delta \sum_{u=1}^{M} \sum_{i,j\in\mathcal{N}(u),i<j} \left(\alpha_{i,j} - S_{i,j}^{u}\right)^2 \\ + (1-\delta) \sum_{u=1,v=2,u<v}^{M} \sum_{i\in\mathcal{N}(u),j\in\mathcal{N}(v),i<j}^{N} \left(\alpha_{i,j} - S_{i,j}^{u,v} - \beta_{u,v}\right)^2 .\end{aligned} \tag{3}$$

Objection Function

- The goal of the multi-view model is tolearn the relationship matrix α and the rating difference matrix β.
  - α: The entry $\alpha_{i,j}$, representing the relationship between items, is consistent both in an individual view and multi view.
  - β: The entry $\beta_{u,v}$ represents the rating difference between users.
- The objective function can be written as follows.

$$\alpha, \beta = \arg\min_{\alpha,\beta} E = \arg\min_{\alpha,\beta} \delta \sum_{u=1}^{M} \sum_{i,j \in \mathcal{N}(u), i<j} \left(\alpha_{i,j} - S^{u}_{i,j}\right)^2 + (1-\delta) \sum_{u=1,v=2,u<v}^{M} \sum_{i \in \mathcal{N}(u), j \in \mathcal{N}(v), i<j}^{N} \left(\alpha_{i,j} - S^{u,v}_{i,j} - \beta_{u,v}\right)^2 . \quad (3)$$

Objection Function

- The goal of the multi-view model is tolearn the relationship matrix α and the rating difference matrix β.
  - α: The entry $\alpha_{i,j}$, representing the relationship between items, is consistent both in an individual view and multi view.
  - β: The entry $\beta_{u,v}$ represents the rating difference between users.
- The objective function can be written as follows.

$$\begin{aligned}\alpha,\beta = \arg\min_{\alpha,\beta} E = \arg\min_{\alpha,\beta} \delta \sum_{u=1}^{M} \sum_{i,j\in\mathcal{N}_(u),i<j} \left(\alpha_{i,j} - S_{i,j}^{u}\right)^2 \\ + (1-\delta) \sum_{u=1,v=2,u<v}^{M} \sum_{i\in\mathcal{N}_(u),j\in\mathcal{N}_(v),i<j}^{N} \left(\alpha_{i,j} - S_{i,j}^{u,v} - \beta_{u,v}\right)^2 .\end{aligned} \tag{3}$$

Model Optimization

We can use the gradient descent method to get the solution of the above function:

$$\begin{aligned}\nabla\alpha_{i,j} = \frac{\partial E}{\partial \alpha_{i,j}} &= 2\delta \sum_{u=1}^{M}(\alpha_{i,j} - S_{i,j}^{u}) \\ &+ 2(1-\delta) \sum_{u=1,v=2,u<v}^{M} (\alpha_{i,j} - S_{i,j}^{u,v} - \beta_{u,v}).\end{aligned} \tag{4}$$

$$\nabla\beta_{u,v} = \frac{\partial E}{\partial \beta_{u,v}} = -2(1-\delta) \sum_{i\in\mathcal{N}_(u),j\in\mathcal{N}_(v),i<j}^{N} (\alpha_{i,j} - S_{i,j}^{u,v} - \beta_{u,v}). \tag{5}$$

Model Optimization

After computing the $\nabla\alpha_{i,j}$ and $\nabla\beta_{u,v}$, the entries in α and β can be updated by the following equations in which γ is the learning rate.

$$\alpha_{i,j} = \alpha_{i,j} - \gamma\nabla\alpha_{i,j}. \tag{6}$$

$$\beta_{u,v} = \beta_{u,v} - \gamma\nabla\beta_{u,v}. \tag{7}$$

Recommend Users to Items.

As to item i, let

- k_1: the number of the chosen items that are closest to i.
- k_2: the number of the users recommended to i who are most likely to buy the item.

Recommend Users to Items.

1. Find k_1 items by sorting the i-th row in α.
2. Find two disjoint sets h_1 and h_2 that represent the users have purchased i and the users have purchased at least one item in the k_1 items list,
3. $\forall u \in h_2$, compute the mean of the total differences between u and all users in h_1. The D_i set is used to record the differences of user u.
4. Sort D_i according to the ascending order, find the smallest k_2 users. Recommend these users to item i.

Recommend Users to Items.

1. Find k_1 items by sorting the i-th row in α.
2. Find two disjoint sets h_1 and h_2 that represent the users have purchased i and the users have purchased at least one item in the k_1 items list,
3. $\forall u \in h_2$, compute the mean of the total differences between u and all users in h_1. The D_i set is used to record the differences of user u.
4. Sort D_i according to the ascending order, find the smallest k_2 users. Recommend these users to item i.

Recommend Users to Items.

1. Find k_1 items by sorting the i-th row in α.
2. Find two disjoint sets h_1 and h_2 that represent the users have purchased i and the users have purchased at least one item in the k_1 items list,
3. $\forall u \in h_2$, compute the mean of the total differences between u and all users in h_1. The D_i set is used to record the differences of user u.
4. Sort D_i according to the ascending order, find the smallest k_2 users. Recommend these users to item i.

Recommend Users to Items.

1. Find k_1 items by sorting the i-th row in α.
2. Find two disjoint sets h_1 and h_2 that represent the users have purchased i and the users have purchased at least one item in the k_1 items list,
3. $\forall u \in h_2$, compute the mean of the total differences between u and all users in h_1. The D_i set is used to record the differences of user u.
4. Sort D_i according to the ascending order, find the smallest k_2 users. Recommend these users to item i.

Recommend Users to Items.

1. Find k_1 items by sorting the i-th row in α.
2. Find two disjoint sets h_1 and h_2 that represent the users have purchased i and the users have purchased at least one item in the k_1 items list,
3. $\forall u \in h_2$, compute the mean of the total differences between u and all users in h_1. The D_i set is used to record the differences of user u.
4. Sort D_i according to the ascending order, find the smallest k_2 users. Recommend these users to item i.

Dataset & Evaluation Metrics

- **Dataset**: The four datasets used in our experiments are *BaiduMovie*, *EachMovie*, *Jester* and *MovieLens*.

- **Evaluation Metrics**: The performances of the recommendation are measured by *Precision*, *Recall* and F_1.

$$Precision = \frac{\sum_i^N |R(i) \bigcap T(i)|}{\sum_i^N |R(i)|} \tag{8}$$

$$Recall = \frac{\sum_i^N |R(i) \bigcap T(i)|}{\sum_i^N |T(i)|} \tag{9}$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{10}$$

where $R(i)$ is the recommendation list to the target item i and $T(i)$ is the behavior list of item i on the testing dataset.

Dataset & Evaluation Metrics

- **Dataset**: The four datasets used in our experiments are *BaiduMovie*, *EachMovie*, *Jester* and *MovieLens*.

- **Evaluation Metrics**: The performances of the recommendation are measured by *Precision*, *Recall* and F_1.

$$Precision = \frac{\sum_i^N |R(i) \bigcap T(i)|}{\sum_i^N |R(i)|} \tag{8}$$

$$Recall = \frac{\sum_i^N |R(i) \bigcap T(i)|}{\sum_i^N |T(i)|} \tag{9}$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{10}$$

where $R(i)$ is the recommendation list to the target item i and $T(i)$ is the behavior list of item i on the testing dataset.

Dataset & Evaluation Metrics

- **Dataset**: The four datasets used in our experiments are *BaiduMovie*, *EachMovie*, *Jester* and *MovieLens*.

- **Evaluation Metrics**: The performances of the recommendation are measured by *Precision*, *Recall* and F_1.

$$Precision = \frac{\sum_i^N |R(i) \bigcap T(i)|}{\sum_i^N |R(i)|} \tag{8}$$

$$Recall = \frac{\sum_i^N |R(i) \bigcap T(i)|}{\sum_i^N |T(i)|} \tag{9}$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{10}$$

where $R(i)$ is the recommendation list to the target item i and $T(i)$ is the behavior list of item i on the testing dataset.

Parameter Analysis on δ

Vary the value of δ from 0 to 0.9 with $k_1 = 5$ and k_2=100, and the discussions of other values of k_1 and k_2 are similar.

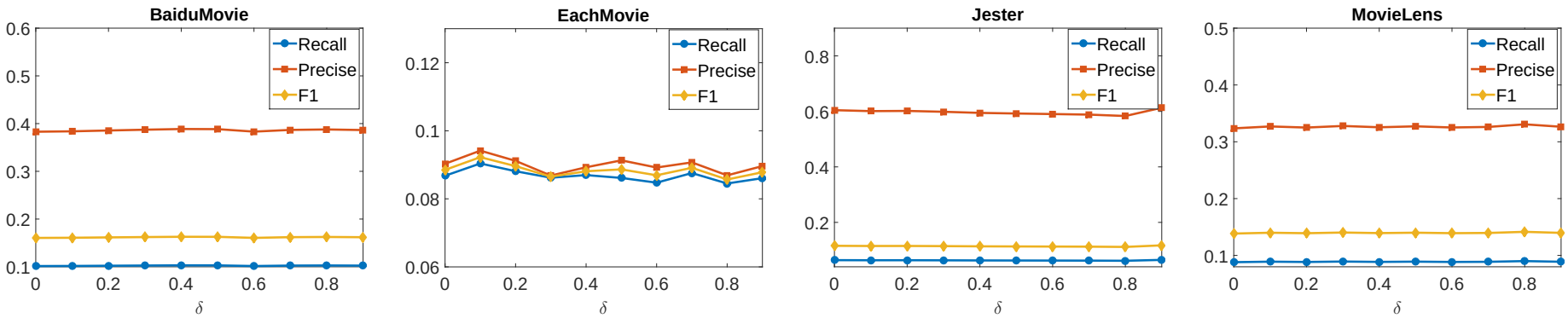

Figure: The value of metrics with different δ values on four datasets in MVIR.

Apparently varying the value of δ in a interval [0,1) does not cause significant changes. Overall, the performances of the MVIR algorithm are robust to the parameter δ on the four datasets.

Parameter Analysis on k_1 and k_2

For each dataset, we conduct the analysis of k_1 and k_2 on the value of δ that achieves the maximum recommended effects. We set k_1 = 5, 10, 15 and k_2= 100, 150, 200.

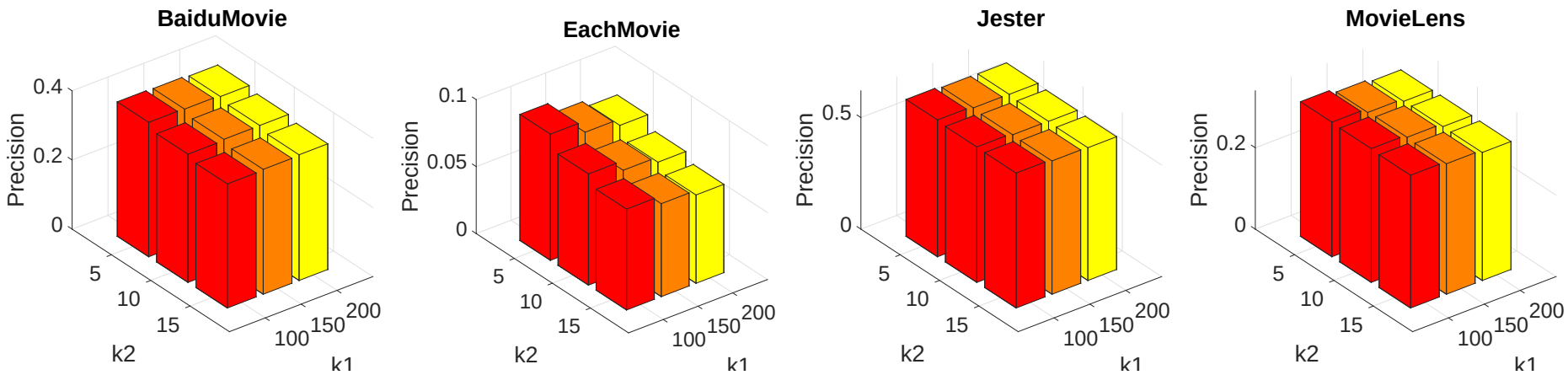

Figure: The value of *Precision* with different k_1 and k_2 values on four datasets in MVIR.

Overall, the performance of the recommendation will be better when the value of k_1 is smaller and the value of k_2 is larger.

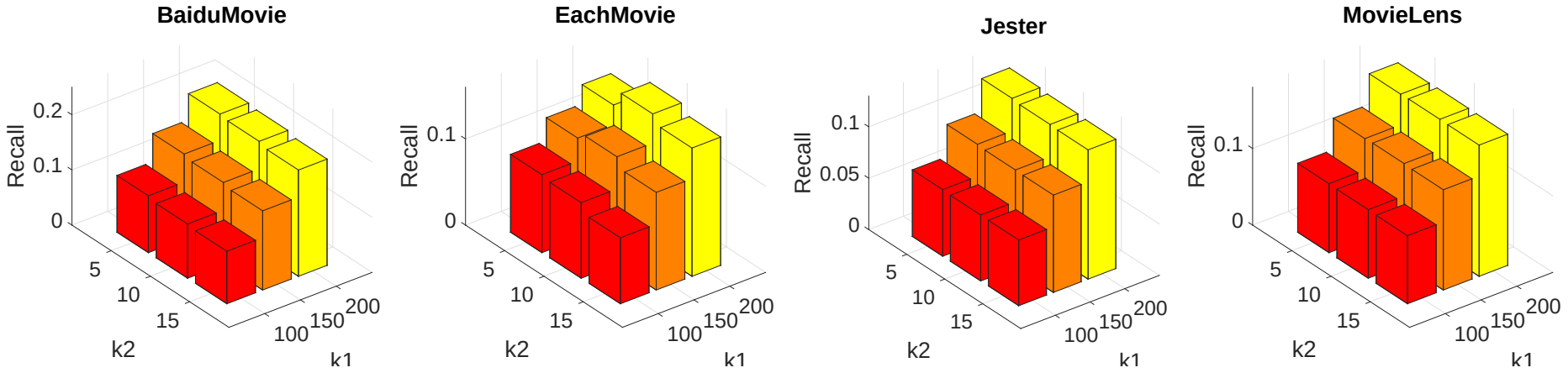

Figure: The value of *Recall* with different k_1 and k_2 values on four datasets in MVIR.

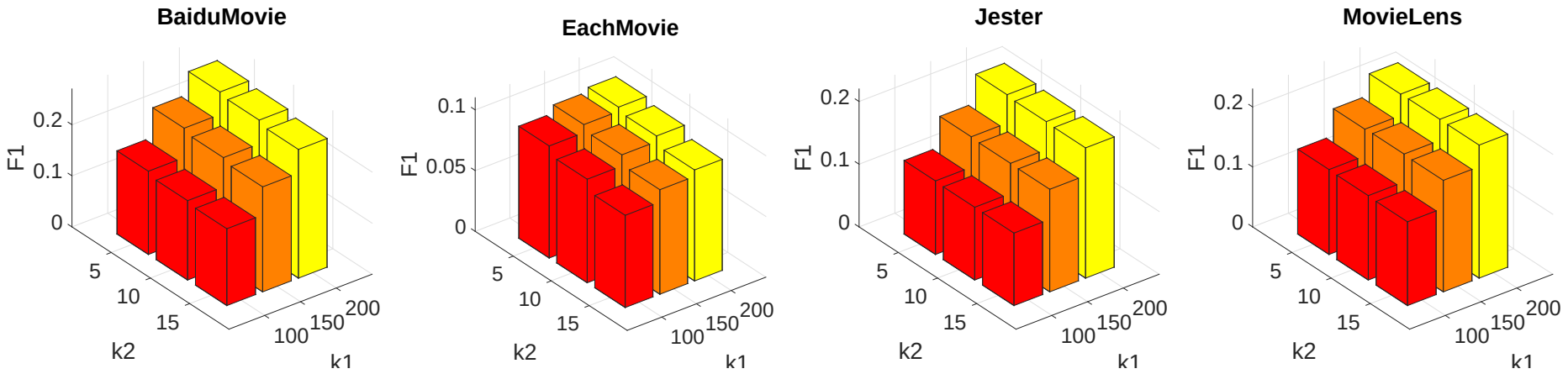

Figure: The value of F_1 with different k_1 and k_2 values on four datasets in MVIR.

Comparison Methods

We present three state-of-the-art recommendation algorithms used to compare the proposed MVIR algorithm, namely **Item-Based Collaborative Filtering** (IBCF), **Slope One** (SO) and **Probabilistic Matrix Factorization** (PMF).

Comparison Result

We set $k_1 = 5$, $k_2 = 100$ and set the best δ value according to the datasets on MVIR algorithm. For PMF, we set D =5.

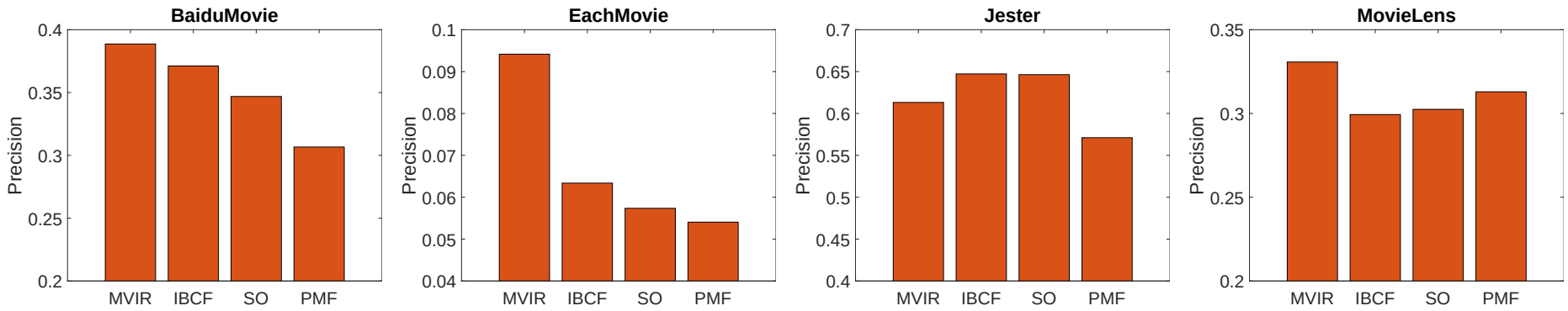

Figure: The value of *Precision* on four dataset of the four recommendation algorithms.

Comparison Result

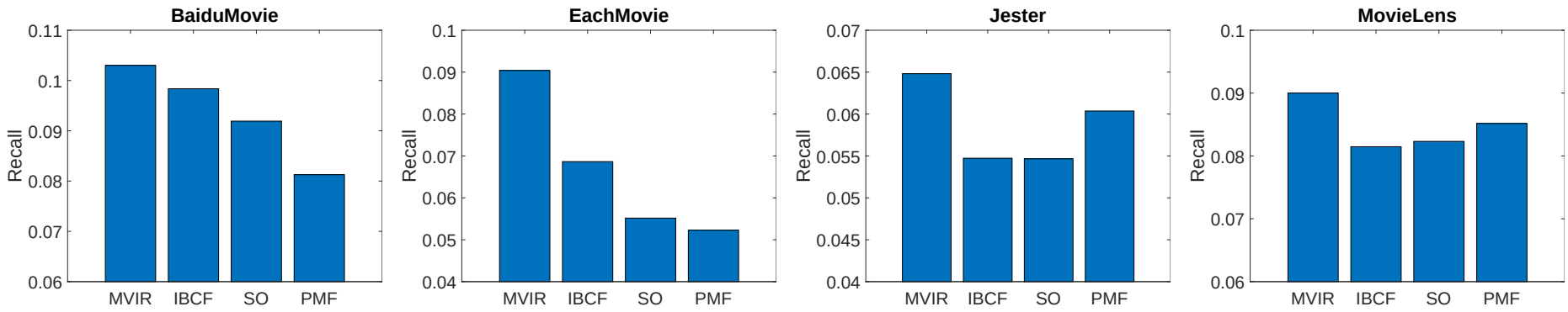

Figure: The value of *Recall* on four dataset of the four recommendation algorithms.

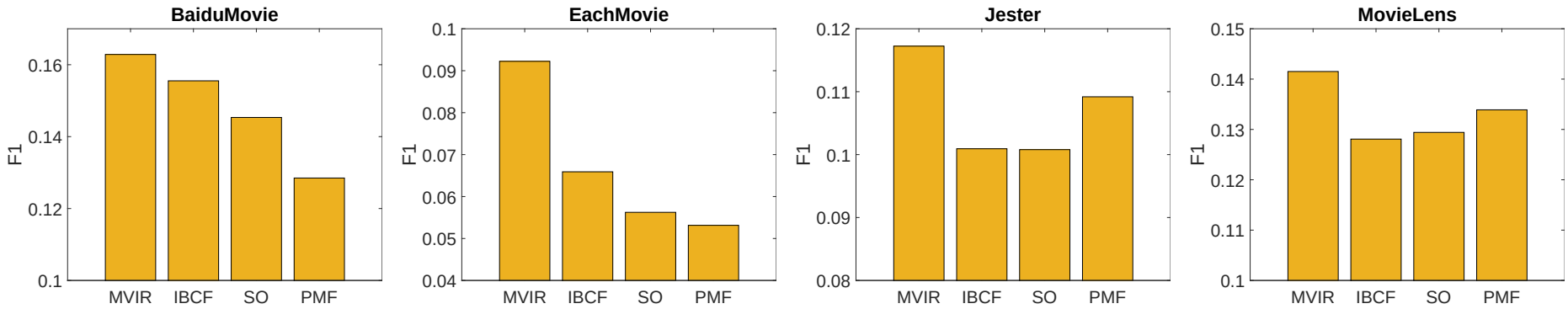

Figure: The value of F_1 on four dataset of the four recommendation algorithms.

Comparison Result

Overall, the recommendation results in MVIR outperform the other algorithms because both the item relationships and user differences are simultaneously considered in MVIR. Different from our model, IBCF and SO only pay attention to the relationship of items and PMF is designed to predict the missing value of a user to an item.

Conclusion

- A novel item orientated recommendation algorithm from the multi-view perspective (MVIR) is proposed in our paper. We use multi-view to capture the relationships between items and the differences between users.
- We construct the multi-view model according to users' rating records in which each user is represented by a view and the nodes in a view are the items rated by the user.
- The experimental results show that our model outperforms other state-of-the-art recommendation algorithms and can predict users who will buy target items accurately.

Conclusion

- A novel item orientated recommendation algorithm from the multi-view perspective (MVIR) is proposed in our paper. We use multi-view to capture the relationships between items and the differences between users.
- We construct the multi-view model according to users' rating records in which each user is represented by a view and the nodes in a view are the items rated by the user.
- The experimental results show that our model outperforms other state-of-the-art recommendation algorithms and can predict users who will buy target items accurately.

Conclusion

- A novel item orientated recommendation algorithm from the multi-view perspective (MVIR) is proposed in our paper. We use multi-view to capture the relationships between items and the differences between users.
- We construct the multi-view model according to users' rating records in which each user is represented by a view and the nodes in a view are the items rated by the user.
- The experimental results show that our model outperforms other state-of-the-art recommendation algorithms and can predict users who will buy target items accurately.

Conclusion

- A novel item orientated recommendation algorithm from the multi-view perspective (MVIR) is proposed in our paper. We use multi-view to capture the relationships between items and the differences between users.
- We construct the multi-view model according to users' rating records in which each user is represented by a view and the nodes in a view are the items rated by the user.
- The experimental results show that our model outperforms other state-of-the-art recommendation algorithms and can predict users who will buy target items accurately.

Thank you very much!
Q&A