Transfer Learning for Sequential Recommendation Model

Chi-Ruei Li
Department of Computer Science and Information Engineering
National Taiwan University
r03922154@csie.ntu.edu.tw

Addicam Sanjay
Intel Corporation
addicam.v.sanjay@intel.com

Shao-Wen Yang
Intel Corporation
shao-wen.yang@intel.com

Shou-De Lin
Department of Computer Science and Information Engineering
National Taiwan University
sdlin@csie.ntu.edu.tw

***Abstract*—In this work, we address the problem of transfer learning for sequential recommendation model. Most of the state-of-the-art recommendation systems consider user preference and give customized results to different users. However, for those users without enough data, personalized recommendation systems cannot infer their preferences well or rank items precisely. Recently, transfer learning techniques are applied to address this problem. Although the lack of data in target domain may result in underfitting, data from auxiliary domains can be utilized to assist model training. Most of recommendation systems combined with transfer learning aim at the rating prediction problem whose user feedback is explicit and not sequential. In this paper, we apply transfer learning techniques to a model utilizing user preference and sequential information. To the best of our knowledge, no previous works have addressed the problem. Experiments on real-world datasets are conducted to demonstrate our framework is able to improve prediction accuracy by utilizing auxiliary data.**

I. Introduction

Recommendation systems attract great attention for business application. For example, directed advertisements on the website can both increase click-through rate and decrease costs to buy banners on web pages. Recommendation systems have two main categories: content-based [1] and collaborative filtering (CF) based [2]. Content-based methods analyze contents of items such as price, tag and description and recommend items based on these contents. The main weakness of content-based methods is the lack of contents. In contrast, collaborative filtering does not rely on item contents but only on the user ratings. The idea of CF is that similar users will have similar preferences. Basic CF only leverage user ratings to calculate user similarity and recommend based on similar users ratings. CF is widely used by large-scale recommendation systems of numerous companies such as Facebook [3] and Amazon [4]. However, CF do not consider chronological order of data and suffers from lack of data. Our framework is proposed to address these two issues.

Conventional CF research solves rating prediction problems and does not consider sequential information. The goal of the rating prediction problem is to predict ratings for items not ever rated by users. Matrix factorization (MF) [5] is a popular method for these problems. However, MF does not consider the sequential information. Chronological order is significant information in some tasks such as next action prediction. For example, next song user may listen to is highly related to both user preference and the current and past songs.

Recommendation systems encounter a huge challenge when dealing with new users and new items, the so-called cold-start problem, because preferences of these users/items cannot be inferred well. Recently, transfer learning [6] is applied to recommendation systems for addressing these problems. Transfer learning attempts to leverage auxiliary domains and gain improvement in target domains. Source domain refers to where auxiliary data comes from and target domain means where we attempt to reach better performance by transferring information from source domains.

Nowadays users have data from different domains. Even though users do not have enough data for a model to infer their preference, other users information can be used to assist model learning. Pan et al. [7] proposed a framework named coordinate system transfer (CST) to address data sparsity problems in CF by utilizing auxiliary information of both users and items. The basic assumption behind CST and other works such as [8] is that the low-level features of users of items are similar. For example, music and video are different contents but both can be characterized by human preferences such as exciting video and exciting music. If these common low-level features were derived from both source and target domains, the information could be utilized to improve model performance in target domains.

In this work, we incorporate transfer learning techniques into personalized sequential recommendation models. For the sequential and personalized recommendation, Rendle et al. [9] proposed a model named factorizing personalized Markov chains (FPMC) to address this problem. FPMC utilizes a transition cube to model both user preference and item transition. Our experiments demonstrate FPMC outperforms traditional MF in the dataset with sequential information. Therefore, in our work, we utilize FPMC as a basic model and extend it with transfer learning to reach better performance. The applied transfer learning technique can be discussed from two aspects:

- **User mapping:** We focus on problems where some inactive users exist. Inactive users may lack data to represent their preference so the recommendations for them could be not so precise as that for active users. This condition could be solved if some users are similar

978-1-5090-5732-0/16/$31.00 ©2016 IEEE

to those inactive users by leveraging their similar users preferences. Therefore, how to map inactive users to active users could be important.

- **Information transfer:** Even if we construct the links of users between source and target domains, the active users features cannot be utilized directly. We do not assume that linked users are the same but only similar. Consequently, the flexibility of exploiting similar users features would affect the result of transfer learning. Guo et al. [10] proposed transfer Bayesian personalized ranking (TBPR) and we associate this work with FPMC to achieve our goal.

The contribution of this paper is that a framework incorporating FPMC with TBPR is proposed to solve the sequential recommendation problems by utilizing auxiliary data. Experiments on two real-world datasets demonstrate this combination could provide a solution to transfer learning for sequential recommendation model.

The rest of the paper is organized as follows. We will first introduce competitive methods and transfer learning models in collaborative filtering. Then we will explain the related work, FPMC and TBPR. In the experiments section, we evaluate proposed model on Last.fm dataset and a mobile application usage dataset. The last section will be the conclusion.

II. Related Work

A. Matrix Factorization (MF)

Matrix Factorization [5] is a widely-used method in rating prediction problems. Figure 1 shows a basic pattern of MF. We denote the set of users as U, set of items as I and $R \in \mathbb{R}^{|U|\times|I|}$ is a rating matrix whose entry $r_{u,i}$ is the rating user u gives item i. The rating prediction problem means that users may not rate all items and we would like to predict missing entries not in R, so-called missing values. MF method estimates two latent matrices, $P \in \mathbb{R}^{|U|\times K}$ and $Q \in \mathbb{R}^{|I|\times K}$, and the product of these two matrices, $\hat{R}$, is the approximation of the original observed R. The K in the above expression is the number of features used to represent each user and item.

MF estimates latent matrices by minimizing the root mean square error (RMSE) of observed ratings between observed R and approximate $\hat{R}$:

$$RMSE = ||W \odot (R - PQ^T)||^2$$

where W is the mask of observation and $W_{uj} = 1$ if user u rated item i or 0 otherwise. Stochastic gradient descent (SGD) can be applied to estimate the model parameters, P and Q, by minimizing this error function. The $\hat{R}$ is a full matrix and thus MF recommends according to those filled rating.

The basic idea of MF is a low-rank approximation. The parameter K is the rank MF utilizes to approximate observed R. The two latent matrices P and Q are user latent matrix and item latent matrix respectively. Each row in P can be regarded as user features representing in K factors and so on item latent matrix Q. Thus, these features can be used to such works as clustering, similarity measurement and so on.

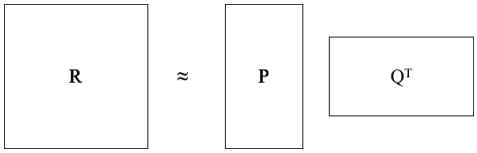

Fig. 1. Matrix Factorization (MF).

B. Transfer Learning for Recommendation Systems

In MF, the two latent matrices are not unique due to missing values in origin matrix. In other words, even for one matrix R, different latent matrices P and Q are derived every time MF is applied to R. Hence, latent matrices from two domains cannot be directly used to estimate the relationship between two domains. For example, we cannot calculate similarity of users from different domains based on user latent matrix. Due to the non-uniqueness of latent matrices of MF, transfer learning applied to recommendation systems can be divided into two categories: with and without correspondence. The user correspondence means user set in the target domain is the same as that in the source domain, and so for item correspondence.

1) With Correspondence: Many previous works [7], [11], [12], [13] have been proposed for transfer learning in collaborative filtering. Most of them assume the user or item correspondences, i.e. the mappings, are known and thus utilize the correspondence as a bridge to solve the non-uniqueness of latent space.

Pan et al. [7] proposed coordinate system transfer (CST) to transfer heterogeneous auxiliary information to target domain. In CST, the non-uniqueness of MF is solved by two source domains with user and item correspondence respectively. Instead of two-matrix factorization, CST utilize sparse SVD [14] as factorization method:

$$\min_{U^i,V^i,B^i} ||W^i \odot (R^i - U^i B^i V^{iT})||^2$$

where $R^i, i = 1, 2$ indicates two source domains. Due to the correspondence of two sides, the trained model can be used to initialize model of the target domain. We denote R^1 has the same user set with target domain and R^2 has the same item set with the target domain. Thus, $U_0 = U^1, V_0 = V^1$ where U_0 and V_0 are initialized latent matrices of the target domain. After that, the target model is trained:

$$\min_{U,V,B} (||W \odot (R - UBV^T)||^2 + \frac{\rho_u}{2}||U - U_0||^2 + \frac{\rho_v}{2}||V - V_0||^2)$$

where ρ_u and ρ_v are regularization constants here. Even with user and item correspondence, the variation between source and target domain should be allowed. The regularization term provides flexibility for transferring information. In summary, CST use initialization from trained source domain models and regularization term to transfer information.

With the similar idea, lots of works are proposed to solve transfer learning with know correspondence. Pan et al. proposed TCF [11] to solve transfer between two heterogeneous

matrices. [12] proposed collective matrix factorization (CMF), where item side is shared. In CMF, there are two matrices, user-movie rating matrix and genre-movie label matrix. The movie side is treated as a bridge for transferring. In SoRec [13], instead, the user side is fixed. SoRec utilize social relation as auxiliary data and transfer the information to assist rating prediction. With user set fixed, CDTF [15] utilize tri-factorization to capture domain-specific factors to more effectively transfer information between domains.

In conclusion, due to non-unique latent factors, these methods solve problems with user or item set fixed. Our framework also assumes user and item correspondence, but only the item correspondence is known.

2) Without Correspondence: Li et al. proposed codebook transfer (CBT) [8] and rating-matrix generative model (RMGM) [16] to address transfer learning for recommendation without correspondence. Their methods can be summarized as follow:

$$R_1 = U_1 B V_1^T \qquad R_2 = U_2 B V_2^T$$

where B is a K by K shared matrix. This shared matrix serves as a bridge between source and target domain. Although these methods do not require user or item correspondence, some constraints are imposed. For example, CBT requires that U and V to be binary matrices and only one value 1 is allowed for each row. In RMGM, U and V are nonnegative and the sum of each row is 1.

Li et al. [17] proposed a method to find user and item correspondence and transfer based on correspondence. They do not assume the mappings of users and items between source and target domains are known. To find these mappings, they utilize SVD to alleviate the non-uniqueness issue of MF. SVD factorizes a matrix into three unique matrices. However, due to missing values, the rating matrix cannot be factorized by SVD directly. Instead, the two latent matrices are factorized by SVD:

$$\begin{aligned} PQ^T &= U_P D_P V_P^T (U_Q D_Q V_Q^T)^T \\ &= (U_P U_X) D_X (V_X^T U_Q^T) \\ &= U D V^T \end{aligned}$$

Based on the decomposition, they attempt to find the user and item mappings. After user and item mapping are constructed, regularization term are used to transfer the information:

$$\begin{aligned} &||W \odot (R_1 - P_1 Q_1^T)||^2 + ||W \odot (R_2 - P_2 Q_2^T)||^2 \\ &+\lambda(||P_1||^2 + ||Q_1||^2 + ||P_2||^2 + ||Q_2||^2) \\ &+\beta \sum_{u \in U_1} \arctan(||P_1(u,:) - P_2(gu,:)||^2) \\ &+\beta \sum_{i \in I_1} \arctan(||Q_1(i,:) - Q_2(gi,:)||^2) \end{aligned}$$

where U_1 denotes user set in the target domain, I_1 denotes item set in the target domain, gu denotes the mapped user of user u and gi denotes the mapped item of item i. In [17], they demonstrate their method outperforms RMGM mentioned above.

In this paper, we propose a framework incorporating sequential recommendation model with transfer learning techniques inspired by these works. Although these works cannot solve sequential recommendation problems directly, we still choose [17] as our main competitor and apply it to our scenario.

III. Methodology

In this section, we will first introduce a optimization criterion, Bayesian personalized ranking (BPR). With concept of BPR, two previous works, FPMC and TBPR, are described as the two main components of our framework. Then, we will introduce our framework FPMC-TBPR. The proposed framework has two main points. One is the construction of user mapping and the other is to transfer information based on the mapping by TBPR. We also describe the main competitive model FPMC-Reg, which utilize regularization term to transfer instead of TBPR.

A. Bayesian Personalized Ranking (BPR)

Both the two main methods, FPMC and TBPR, utilized the concept of BPR [18]. Unlike conventional rating-based problems whose feedback is explicit, BPR attempts to solve problems with implicit feedback and optimize for ranking. For explicit feedback, e.g. ratings, data is usually numerical and represents different levels of preference. However, for implicit feedback, e.g. clicks or usage, is usually binary and not related to preference directly. For example, browsing a website cannot represent one like it or not. BPR is widely used for problems of implicit feedback.

The basic assumption of BPR is that a user prefers observed items over unobserved items. Here we introduce a structure notation $>_u$ that means the preference of user u, and we can denote user u prefers item i over item j:

$$i >_u j \quad i \in I_u^+,\ j \in I \backslash I_u^+$$

where I_u^+ represents observed item set of user u. Specifically, the preference of items means the order of items potential scores. The ranking is defined:

$$i >_u j \Leftrightarrow x_{u,i} > x_{u,j}$$

where $x_{u,i}$ is the potential score of item i for user u. For clarity, based on the assumption and notations, the data format for BPR can be written as:

$$D_s = \{(u,i,j) \mid i \in I_u^+ \wedge j \in I \backslash I_u^+\}$$

In order to estimate the model parameters to capture the ranking structure, the following posterior probability is maximized:

$$p(\theta| >_u) \propto p(>_u |\theta) p(\theta)$$

where θ represent model parameters. Before optimization of the model parameters, we have to define the individual probability using logistic function $\sigma(x) = \frac{1}{1+e^{-x}}$:

$$p(i >_u j|\theta) = \sigma(x_{u,i} - x_{u,j})$$

Then, we formulate maximum a posteriori (MAP) estimator to calculate the model parameters:

$$\begin{aligned}
&\arg\max_{\theta} \ln p(\theta| >_u) \\
=&\arg\max_{\theta} \ln p(>_u |\theta) p(\theta) \\
=&\arg\max_{\theta} \ln \prod_{(u,i,j)\in D_s} \sigma(x_{u,i} - x_{u,j}) p(\theta) \\
=&\arg\max_{\theta} \sum_{(u,i,j)\in D_s} \ln \sigma(x_{u,i} - x_{u,j}) - \lambda||\theta||^2
\end{aligned}$$

where λ is regularization constant.

As the objective function is differentiable, BPR utilizes stochastic gradient descent (SGD) with bootstrap sampling to solve the optimization. BPR is a general framework of ranking optimization and is applied to MF as illustration in [18]. The combination of BPR and MF can reach better performance on datasets with implicit feedback.

B. Factorizing Personalized Markov Chains (FPMC)

FPMC is a recommendation model including both user preference and sequential information. Two fundamental approaches for user preference and sequential information are matrix factorization (MF) and Markov chains (MC) respectively. MF learns user preferences by factorizing observed rating matrix into user latent matrix and item latent matrix. MC utilizes a transition graph to representing probabilities from one item to others. FPMC combines these two concepts and proposes an idea of personalized transition graph. In other words, the transition probability depend on not only item but also user. Based on this concept, FPMC can recommend according to users preference and recent actions.

In [9], FPMC is designed to solve the basket recommendation problem. The basket recommendation problem is to predict next set of items according to current set of items. Although we focus on single item transition, i.e., size of basket equals one, we still follow the origin paper to introduce FPMC and then apply FPMC to our case.

Given a basket, a set of items appearing at the same time, we would like to predict items in next basket $p(B_t^u|B_{t-1}^u)$ where B_t^u means the user u interacts with items in the item set B_t^u in time t. The basic idea of basket transition is that transition probability from basket to an item is the average of transition probability of all items in basket:

$$p(i \in B_t^u|B_{t-1}^u) = \frac{1}{|B_{t-1}^u|} \sum_{l\in B_{t-1}^u} p(i \in B_t^u|l \in B_{t-1}^u)$$

Note that for prediction for next basket, actually we predict the probabilities of all items and thus the rank list of items.

Personalized transition graph is underlying idea and simple to realize. However, it is not feasible in practice. The transition graph can be regarded as a matrix, so-called transition matrix, whose entries represent the probabilities of transition from one to another. Thus the personalized transition graph can be viewed as a 3-dimensional transition matrix or a transition cube $A \in [0,1]^{|U|\times|I|\times|I|}$ where U is the set of users and I is the set of items, and each entry in A is denoted as $a_{u,l,i} = p(i \in B^u|l \in B_{t-1}^u)$.

A transition cube is fully parameterized, which means each parameter only affects one kind of transition. The number of parameters is the number of entries in transition cube, $|U||I|^2$, and each parameter in this cube is independent. Therefore, the negative effect of data sparsity will become serious, especially for those transitions unobserved in training data. To overcome the disadvantage of full parameterization, FPMC exploits canonical decomposition to factorize the transition cube. After factorization, the $a_{u,l,i}$ become:

$$a_{u,l,i} = \langle v_u^{U,I}, v_i^{I,U}\rangle + \langle v_i^{I,L}, v_l^{L,I}\rangle + \langle v_u^{U,L}, v_l^{L,U}\rangle$$

The main advantage of this decomposition is that number of parameters decreases and they become dependent on others. That is, now an observation affects several parameters and it is more likely that all parameters can be estimated. The formula of FPMC becomes:

$$\begin{aligned}
&p(i \in B_t^u|B_{t-1}^u) \\
=&\frac{1}{|B_{t-1}^u|} \sum_{l\in B_{t-1}^u} p(i \in B_t^u|l \in B_{t-1}^u) = \frac{1}{|B_{t-1}^u|} \sum_{l\in B_{t-1}^u} a_{u,l,i} \\
=&\frac{1}{|B_{t-1}^u|} \sum_{l\in B_{t-1}^u} \langle v_u^{U,I}, v_i^{I,U}\rangle + \langle v_i^{I,L}, v_l^{L,I}\rangle + \langle v_u^{U,L}, v_l^{L,U}\rangle \\
=&\langle v_u^{U,I}, v_i^{I,U}\rangle + \frac{1}{|B_{t-1}^u|} \sum_{l\in B_{t-1}^u} \langle v_i^{I,L}, v_l^{L,I}\rangle + \langle v_u^{U,L}, v_l^{L,U}\rangle
\end{aligned}$$

FPMC utilizes the concept of BPR for optimization. For clarity, we denote $x_{u,t,i} = p(i \in B_t^u|B_{t-1}^u)$, and then the ranking structure becomes:

$$i >_{u,t} j \Leftrightarrow x_{u,t,i} > x_{u,t,j}$$

As the derivation of subsection III-A, the MAP estimator for model parameters:

$$\begin{aligned}
&\arg\max_{\theta} \ln p(\theta| >_{u,t}) \\
=&\arg\max_{\theta} \ln p(>_{u,t} |\theta) p(\theta) \\
=&\arg\max_{\theta} \ln \prod_{u\in U} \prod_{B_t^u\in B^u} \prod_{i\in B_t^u} \prod_{j\notin B_t^u} \sigma(x_{u,t,i} - x_{u,t,j}) p(\theta) \\
=&\arg\max_{\theta} \sum_{u\in U} \sum_{B_t^u\in B^u} \sum_{i\in B_t^u} \sum_{j\notin B_t^u} \ln \sigma(x_{u,t,i} - x_{u,t,j}) - \lambda||\theta||^2
\end{aligned}$$

As the optimization for ranking, $x_{u,t,i} - x_{u,t,j}$, the $v^{U,L}$ and $v^{L,U}$ in FPMC are independent to i and j. Therefore, the simpler expression can be used:

$$x_{u,t,i} = \langle v_u^{U,I}, v_i^{I,U}\rangle + \frac{1}{|B_{t-1}^u|} \sum_{l\in B_{t-1}^u} \langle v_i^{I,L}, v_l^{L,I}\rangle$$

In our work, we utilize FPMC with single item transition, i.e. basket size $|B_t^u| = 1$ for all baskets, which means:

$$x_{u,t,i} = \langle v_u^{U,I}, v_i^{I,U}\rangle + \langle v_i^{I,L}, v_l^{L,I}\rangle$$

FPMC exploits SGD and bootstrap sampling to estimate the model parameter as we describe in subsection III-A. The gradients of BPR are:

$$\frac{\partial}{\partial\theta}(\ln\sigma(x_{u,t,i}-x_{u,t,j})\lambda\theta^2)$$

C. Transfer Bayesian Personalized Ranking (TBPR)

TBPR is a transfer learning technique proposed in CroRank [10]. In CroRank, the authors assume user set of source and target domains are the same and then find the relationship of items between these two domains. There are three concepts of TBPR:

1) **Related Item Group.** For an item i in the target domain, there is a group of similar items in the source domain and its notation is I_i^g.
2) **Inner Domain Preference.** This is the origin concept of BPR, i.e. the ranking structure of target domain.
3) **Related Item Preference.** The ranking structure in the source domain. The assumption is that similar items ranking will be the same in every domain. Therefore, if $x_{u,i} > x_{u,j}$, the ranking of related group is:

$$x_{u,I_i^g} > x_{u,I_j^g} \qquad i,j \in I_t \wedge I_i^g, I_j^g \in I_s$$

where I_t represents item set of the target domain and I_s represents item set of the source domain.

Based on these concepts, the ranking structure of source and target domains can be combined as:

$$\rho x_{u,i} + (1-\rho)x_{u,I_i^g} > \rho x_{u,j} + (1-\rho)x_{u,I_j^g}$$

where ρ in [0,1] is weight parameter of source and target domain. When ρ equals to 1, the effect from source domain is ignored. ρ is determined by cross-validation. The new ranking structure can be estimated in the same way of subsection III-A.

D. Transfer Learning for FPMC

Our goal is to apply transfer learning to sequential recommendation systems. In our framework, we utilize FPMC as our basic model and then transfer learning can be divided into two parts: construct user mapping and transfer information based on the mapping. The first subsection describes the way to construct user mapping. The second and third subsections introduce how to transfer information based on user mapping by regularization term and TBPR, respectively.

1) User Mapping from Latent Space: The first step of our framework is to construct the user mapping. Our goal is to solve the lack of data of inactive users. Therefore, we have to find similar users so that inactive users can leverage their features.

In our framework, we use latent features similarity with shared space. Source domain and target domain are used to pre-train two FPMC models with the item space shared. By this way, the user spaces of two models are forced to aligned to item space and thus the underlying bases of them become the same. For FPMC model, we regard $v^{U,I}$ as user latent features and the other 3 latent matrices, $v^{I,U}$, $v^{I,L}$ and $v^{L,I}$, as shared item latent features. Cosine similarity is applied to estimate the similarity between users. In our current settings, each user in target domain is linked to the most similar user. Experiments demonstrate that this method can effectively construct the user mapping.

2) Transfer Information by Regularization (FPMC-Reg): The second step of out framework is to transfer information to gain improvement based on the user mapping. Since users in target domain may not have enough data for a model to infer their features, an extreme idea to leverage similar users information is to directly use well-trained features from similar users. However, even though two users are very similar, it is highly possible that they have slight difference. Moreover, the user mapping is not perfect. Therefore, to transfer with flexibility is a challenging issue.

We propose a combination of FPMC and regularization term, called FPMC-Reg. FPMC-Reg leverages regularization term for user feature difference on update formula of $v^{U,I}$, the user latent matrix. This method leverages information from source domain in a flexible way because regularization terms do not enforce a parameter to be same as others. That is, the MAP estimator becomes:

$$\arg\max_\theta \sum_{u\in U}\sum_{B_t^u\in B^u}\sum_{i\in B_t^u}\sum_{j\notin B_t^u}\ln\sigma(x_{u,t,i}-x_{u,t,j}) - \lambda||\theta||^2 - \beta\sum_{u\in U_{tgt}}||v_u^{U,I}-v_{gu}^{U,I}||^2$$

where U_{tgt} is the user set of the target domain and gu means mapped user of u. Here we assume one user in the target domain is only mapped to one user in the source domain. Then, the gradients of BPR become:

$$\frac{\partial}{\partial\theta}(\ln\sigma(x_{u,t,i}-x_{u,t,j})\lambda\theta^2 - \beta(v_u^{U,I}-v_{gu}^{U,I})^2)$$

where β is the regularization constant for user difference. The only one changed update formula is the one for $v^{U,I}$:

$$\begin{aligned} v_u^{U,I} &= v_u^{U,I} + \alpha(\delta(v_i^{I,U}-v_j^{I,U}) - \lambda v_u^{U,I}) \\ \Rightarrow v_u^{U,I} &= v_u^{U,I} + \alpha(\delta(v_i^{I,U}-v_j^{I,U}) - \lambda v_u^{U,I} - \beta(v_u^{U,I}-v_{gu}^{U,I}) \end{aligned}$$

where $\delta = 1-\sigma(x_{u,t,i}-x_{u,t,j})$. With this form of update formula, whenever target domain user's features in $v^{U,I}$ are updated, the regularization term of user difference provides information of source domain by restricting the difference of two similar users. FPMC-Reg utilizes the straightforward regularization term as a transferring technique, and thus become the main competitor of our FPMC-TBPR.

3) Transfer Information by TBPR (FPMC-TBPR): Regularization term is a simple but useful method to transfer information, but we are dedicated to more effective way to transfer information. In our framework, we incorporate FPMC with TBPR, called FPMC-TBPR, to outperform FPMC-Reg. Regularization term does not directly transfer the ranking structure. Instead, TBPR considers similar user's effect by

joining ranking potential scores, which is the objective function of FPMC. The MAP estimator of FPMC-TBPR is:

$$\arg\max_{\theta} \sum_{u\in U} \sum_{B_t^u \in B^u} \sum_{i\in B_t^u} \sum_{j\notin B_t^u} \ln \sigma((\rho x_{u,t,i} + (1-\rho)x_{gu,t,i}) - (\rho x_{u,t,j} + (1-\rho)x_{gu,t,j})) - \lambda||\theta||^2$$
$$= \arg\max_{\theta} \sum_{u\in U} \sum_{B_t^u \in B^u} \sum_{i\in B_t^u} \sum_{j\notin B_t^u} \ln \sigma(x_{\hat{u},t,i} - x_{\hat{u},t,j}) - \lambda||\theta||^2$$

Then the gradients of BPR becomes:

$$\frac{\partial}{\partial\theta}(\ln\sigma(x_{\hat{u},t,i} - x_{\hat{u},t,j})\lambda\theta^2)$$

FPMC-TBPR utilizes information from the source domain, which is the same as FPMC-Reg. Observing the gradient formula, we can realize that TBPR directly incorporates auxiliary information into BPR structure. Our experiment results demonstrate FPMC-TBPR can transfer information more effectively.

Figure 2 shows the flowchart of FPMC-TBPR. A pre-train model with item latent matrices shared is trained with bootstrap sampling on data from both source and target domains. This model is used to construct user mapping. Target model is initialized by latent matrices from pre-train model and then trained only on target domain data with the user mapping.

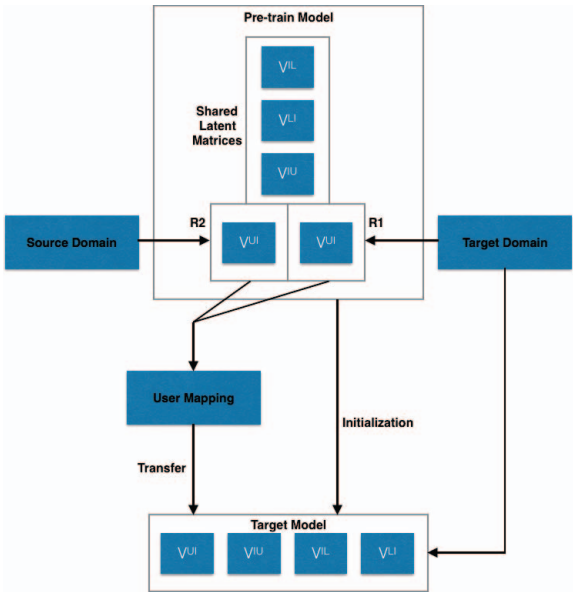

Fig. 2. Flowchart of FPMC-TBPR.

IV. Experiments

In this section, we compare proposed framework, FPMC-TBPR, to other models including FPMC-Reg and baseline methods mentioned later. To evaluate the effective of transfer learning, two real-world datasets are split into source and target domains according to three scenarios.

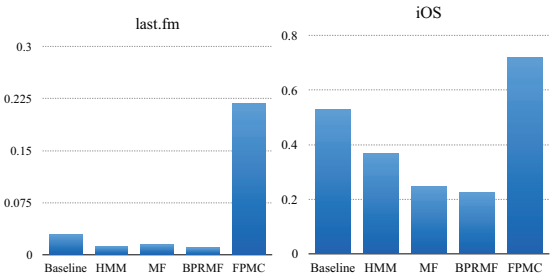

Fig. 3. The effectiveness of FPMC.

A. Dataset

Two real-world datasets, Last.fm[1] [19] and iOS mobile devices logs[2] are used to evaluate our framework. The Last.fm dataset contains records of music playing. The iOS dataset contains records of application usage. Both datasets contain user IDs and timestamps. The two datasets are used because they both include user preferences and chronological order and thus are proper dataset for FPMC.

We first clean data and take subsets of them. Then, we sort these records in chronological order and split into several sequences. The statistics of preprocessed datasets are shown in table I.

B. Split Scenario

To evaluate the improvement of transfer learning, we split datasets into target domain R^1 and source domain R^2 by three different scenarios.

1) **Split_a.** Sequences of a user are split into two parts, 80% for R^2 and 20% for R^1. Besides, the sequences divided into R^1 are assigned to new user IDs.
2) **Split_b1.** Randomly split users into two parts, 80% users for R^2 and 20% users for R^1.
3) **Split_b2.** Randomly split users into two parts, 80% users for R^2 and 20% users for R^1. Besides, for users divided to R^1, only 20% sequences of each user are used. The remaining 80% sequences are ignored.

C. Evaluation

The goal of a single FPMC model is to predict next item based on current item. Mean reciprocal rank (MRR) is used to evaluate item prediction. For a query (prediction) $q \in Q$, a rank list of item is generated by model. Reciprocal rank means the reciprocal of rank of answer in this rank list. MRR of whole query set Q is denoted as:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i}$$

D. Baseline Methods

Two baseline methods are considered for the overall framework comparison.

[1]www.last.fm
[2]The dataset, iOS mobile device logs, is provided by Intel corporation.

TABLE I
DATASET STATISTICS

dataset	users $\|U\|$	items $\|I\|$	sequences	avg. sequences length
Last.fm subset	500	10,000	224,956	20.22
iOS subset	700	10,000	161,611	10.91

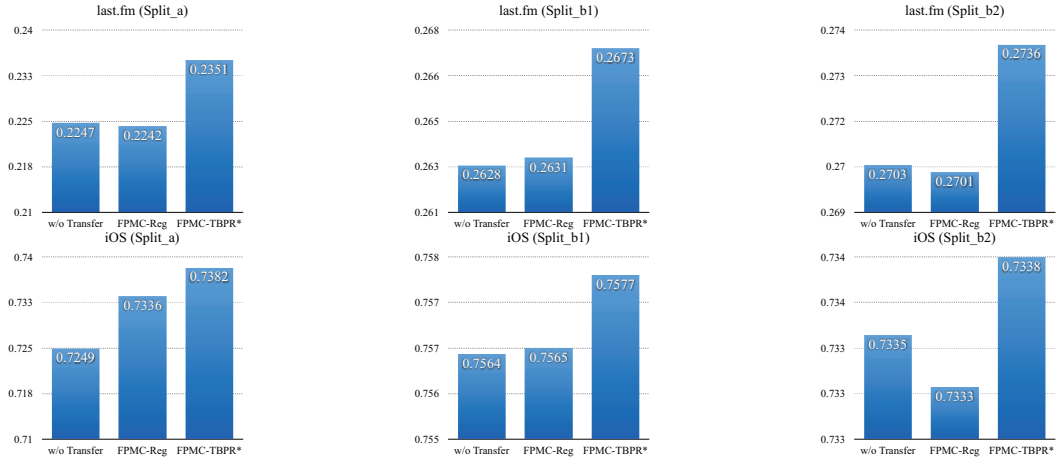

Fig. 4. Experiment results on split.

TABLE II
EXPERIMENT RESULTS OF LI 2014

dataset	Split_a	Split_b1	Split_b2
Last.fm subset	0.0050	0.0168	0.0063
iOS subset	0.2259	0.2335	0.2523

1) **Without Transfer.** The target model trained with initialization from the pre-trained model but without transferring user information. This will defeat a single FPMC model trained only with the target domain. Even though users information is not transferred, the well-trained item latent features can be leveraged in improve the performance. Therefore, instead of single FPMC model trained on target domain, we choose this method as a basic baseline.
2) **FPMC-Reg.** Mentioned in section III-D2. This is the main competing method FPMC-TBPR attempting to defeat.

E. The Effectiveness of FPMC

Before the results of transfer learning, we first present the effectiveness of FPMC on these two datasets. FPMC are compared to these models:

1) **Baseline - Most Popular (MP).** MP means items are ranked according to their counts of usage in their personal history. This method is widely used in many previous works of recommendation systems as a baseline.
2) **Hidden Markov Model (HMM).** HMM [20] is a famous and basic model in sequence generation. Thus, we compare FPMC to HMM in these two datasets.
3) **Matrix Factorization (MF).** MF are mentioned in subsection II-A. MF cannot be applied to datasets with sequential and implicit feedback but is a basic model for many related works. Hence, in these datasets, we regard counts of usage as ratings and apply MF as a competitive model of FPMC.
4) **Bayesian Personalized Ranking Matrix Factorization (BPRMF).** Due to implicit feedback of these two datasets, we utilize [18] to solve these problems. BPRMF does not consider sequential information and the occurrence frequency of items.

Figure 3 indicates that FPMC outperforms these conventional models, and thus our work focuses on how to extend it.

F. Results of Transfer Learning

Comparing the overall performance of our framework with other two methods, we can see that FPMC-TBPR can reach the best performance. Split_a in Figure 4 demonstrate FPMC-TBPR is able to improve performance with auxiliary data on both two datasets while FPMC-Reg only improves on iOS dataset. This means that FPMC-TBPR utilizes the user mapping and source domain in a better way. Split_b1 in Figure 4 shows that even if users in target domain do not have exactly the same users in the source domain, FPMC-TBPR can still extract information from auxiliary data. In Split_b2 of Figure 4, we can observe that the improvement of FPMC-TBPR is less than in split_b1. The reason is that transfer learning is still limited. Therefore, users in target domain should own enough data for the model to learn the difference between users.

Transfer learning with conventional MF is also applied as a competitive model. Here we utilize [17] to conduct the

experiment and the results are shown in table II. The result demonstrate even with transfer learning the conventional MF cannot be used to solve sequential recommendation problems.

G. Results of User Mapping

Figure 5 shows the results on split_a scenario of the Last.fm dataset if only part of users are transferred. We sort users according to their distance to linked users and take only the most similar part to transfer. We think if the lower distance means the confidence of correct mapping and the improvement of transfer are higher. Figure 5 demonstrates that the improvement of transfer increases faster when users with higher similarity are take into transfer.

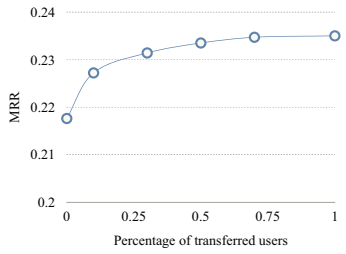

Fig. 5. Part of users are transferred.

V. Summary

The experiment results show our model could improve prediction accuracy by utilizing source domain. We demonstrate FPMC outperforms other methods on the datasets with sequential information and thus choose FPMC as our basic model. Then, we split these two real-world datasets by three scenarios to evaluate our framework. These results demonstrate our framework enables FPMC to leverage auxiliary data and reach better performance.

VI. Conclusion

In this paper, we propose a framework FPMC-TBPR, which incorporates sequential recommendation model FPMC and transfer learning technique TBPR. This two-step framework includes construct user mapping and transfer information based on the mapping. The mapping is constructed according to the similarity in user latent space with item space shared by source and target domains. This method has been demonstrated its usefulness in two real world datasets. After user mapping is constructed, we utilize TBPR to transfer information of similar users during model training. The main competitive model is FPMC-Reg. Experiments on real-world datasets demonstrate the effectiveness of our framework.

A possible future work is flexible user mapping. So far for one user in target domain, only the most similar user in the source domain is considered. However, the user mapping constructed by similarity in latent space may be not perfect. With more flexible mapping, the improvement of transfer learning may be more consistent.

Acknowledgment

We would like to acknowledge Kalpana Algotar and Michael Wu at Intel in their tireless work in data collection and management.

References

[1] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 325–341.

[2] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in artificial intelligence*, vol. 2009, p. 4, 2009.

[3] M. Kabiljo and A. Ilic, "Recommending items to more than a billion people," 2015. [Online]. Available: https://code.facebook.com/posts/861999383875667/recommending-items-to-more-than-a-billion-people/

[4] G. Linden, B. Smith, and J. York, "Amazon. com recommendations: Item-to-item collaborative filtering," *IEEE Internet computing*, vol. 7, no. 1, pp. 76–80, 2003.

[5] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, aug 2009.

[6] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. on Knowl. and Data Eng.*, vol. 22, no. 10, pp. 1345–1359, oct 2010.

[7] W. Pan, E. W. Xiang, N. N. Liu, and Q. Yang, "Transfer learning in collaborative filtering for sparsity reduction," in *AAAI*, vol. 10, 2010, pp. 230–235.

[8] B. Li, Q. Yang, and X. Xue, "Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction," in *IJCAI*, vol. 9, 2009, pp. 2052–2057.

[9] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized Markov chains for next-basket recommendation," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 811–820.

[10] Y. Guo, X. Wang, and C. Xu, "CroRank: cross domain personalized transfer ranking for collaborative Filtering," in *Proceedings of the 2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, ser. ICDMW '15. Washington, DC, USA: IEEE Computer Society, 2015, pp. 1204–1212.

[11] W. Pan and Q. Yang, "Transfer learning in heterogeneous collaborative filtering domains," *Artificial intelligence*, vol. 197, pp. 39–55, 2013.

[12] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2008, pp. 650–658.

[13] H. Ma, H. Yang, M. R. Lyu, and I. King, "Sorec: social recommendation using probabilistic matrix factorization," in *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM, 2008, pp. 931–940.

[14] N. D. Buono and T. Politi, "A continuous technique for the weighted low-rank approximation problem," in *ICCSA*. Springer Berlin Heidelberg, 2004, pp. 988–997.

[15] L. Hu, J. Cao, G. Xu, L. Cao, Z. Gu, and C. Zhu, "Personalized recommendation via cross-domain triadic factorization," in *Proceedings of the 22nd international conference on World Wide Web*. ACM, 2013, pp. 595–606.

[16] B. Li, Q. Yang, and X. Xue, "Transfer learning for collaborative filtering via a rating-matrix generative model," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 617–624.

[17] C.-Y. Li and S.-D. Lin, "Matching users and items across domains to improve the recommendation quality," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '14. New York, NY, USA: ACM, 2014, pp. 801–810.

[18] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, vol. cs.LG. AUAI Press, 2009, pp. 452–461.

[19] O. Celma, *Music Recommendation and Discovery in the Long Tail*. Springer, 2010.

[20] L. Rabiner and B. Juang, "An introduction to hidden Markov models," *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986.