Ranking-based Personalized Recommendation Using Social and Sequential Signals

***Abstract*—In this Assignment, we address the one-class recommendation problem which arises commonly in real-world settings when explicit user feedback is unavailable for making the recommendations. In such cases, we adopt a ranking-based recommendation approach that models the relative item preferences for each user. Previous works have shown that efficient ranking-based models can be developed by utilizing a user's social connections and sequential (time-dependent) feedback information. Hence we implemented the Socialized Personalized Markov Chains (SFPMC) Model as introduced by Cai et. al., 2017 [1] on the LibraryThing dataset to provide a personalized feed of recommended books to users. We then compared the results of the SFPMC model with other sequential models, such as FPMC and FMC and socially aware models, such as BPR and SBPR.**

I. Dataset and exploratory analysis

Dataset

We have used the LibraryThing[1] reviews dataset, which includes both the user ratings on books and the users' social connections. The dataset, which was open-sourced in Zhao et. al, 2014[2] was collected from LibraryThing, a popular book-reviewing website. The dataset provides user feedback in the form of ratings on books where each entry specifies a user name, book id, comment (descriptive review), date of the review, and star rating. Some ratings are flagged either as 'abuse', 'not a review' or both. The dataset contains 10,00,000 such data points. The most recent review was on 26th August 2013, and the earliest was provided on 31st December 1969. Along with user feedback, it also includes information about a user's friends.

Preprocessing

The task we have identified is to solve the top-K recommendation problem where we recommend K most 'likely to be read books' for a user. Hence we explicitly filter out negative feedback that is reviews with rating less than 4. This is inspired by Zhao et. al., 2014[2]'s work on social BPR. We also filter out reviews with the flag as 'abuse' or 'not a review' since such feedbacks are somewhat deviant. After performing the above-mentioned preprocessing and dropping the data points which don't specify the rating, we have 4,97,306 data points and performed train, validation and test split following the 8:1:1 ratio.

Exploratory Analysis

The basic statistics of the dataset are presented in Table I. We notice the ratio of social connections to users for our taken dataset to be 1.845, which is significantly lower than other datasets: Ciao, Foursquare, Epinions and Flixter, which have the ratio of 20.93, 4.72, 4.55 and 13.84 respectively as per Cai et. al, 2017[1]

Analyzing the social relations count, we found that 99.2% of the users have less than 50 social connections and 10 users (0.02% of the total users) have more than 400. Four among the 10 users with more than 400 social connections have fewer than 5 interactions. In total, 34681 users, i.e 67.45% of users have less than 5 interactions, and their average social interaction count is 2.058, which is again significantly lower than other datasets: Ciao, Delicious and Epinions, which have a ratio of 35.95, 8.14, 3.369 respectively as per Zhao et. al, 2014[2]. Hence, leveraging friends' interactions information in our dataset can circumvent the cold-start problem though we are at a slight disadvantage compared to the other datasets with higher average social relations for cold-start users. 23656 users, i.e 46% of users have less than 5 interactions and zero social connections; only sequential feedback information can be used to obtain ranking-based item recommendations for such users. The maximum number of social connections a user has in our dataset is 3415.

Zero previous interactions will result in the absence of sequential feedback information, but we found only three such cases in our chosen dataset, and hence we can ignore this case.

TABLE I
Dataset Statistics.

Statistic	LibraryThing
#users	51416
#items	189971
#feedback	497306
#trusts	94894
#trusts/#users	1.845

II. Predictive Taks and Baselines

Predictive Task

Our predictive task is the Top-N recommendation problem i,e personalized recommendation of ranked N most 'likely to be read books'. We approach this by defining a score function $x_{i,u}$ which gives a score for a {user, book} pair u, i and $x_{i,u,l}$ which gives a score for a {user, book} pair u, i where l is the last reviewed book by the user. Higher the score the more likely the user is to read the book. The baseline models we used for this task are BPR (Bayesian Personalized Ranking), Social BPR, FMC (Factorized Markov Chains) and FPMC (Factorized Personalized Markov Chain) and we

[1]LibraryThing Dataset

compare their results with SFPMC (Socialized Personalized Markov Chains). The score function used in BPR and SPBR considers only $user/item$ interactions whereas the function of FMC considers only $item/item$ interactions. FPMC's score function includes both the $user/item$ and $item/item$ interactions.

The SFPMC model considers all $user/item$, $item/item$ and $user/user$ interactions. By including $user/user$ interactions, we leverage the social connections of users and by including $item/item$ interactions we also consider the sequential information of the user (previous positive feedback information). Hence the SFPMC model utilizes both social and sequential information simultaneously for improved recommendation performance. We use σ to denote the logistic (sigmoid) function in the following sections.

In our model implementation we only leverage on the user, item interactions without considering any explicit feedback such as rating, sentiment of the review text, etc. In many real-life applications, explicit feedback such as ratings might not be available and one must instead try to model some form of implicit feedback, such as the media users consume, the pages they browse, the music they listen to, or whom they befriend. This setting is called "one-class" recommendation and a variety of solutions have been proposed to solve it by directly modeling relative preferences, or rankings, of items for personalized recommendation.

Before discussing about the evaluation metric we define the following concepts:
1. Positive feedback: $P_u = \{(u,i)\}$, set of (user, book) pairs where i was given a rating not less than 4 by the user in previous interactions.
2. Negative feedback: $N_u = \{(u,i)\}$, set of (user, book) pairs where user hasn't demonstrated a positive feedback towards i in previous interactions.
3. Social feedback: $S_u = \{(u,i)\}$, set of (user, book) pairs where atleast one of users' friends have given positive feedback on i but the users have not interacted with the item before.

Evaluation Metric

We use the Area under the ROC curve (AUC) metric to evaluate the results of our implemented models. The range for AUC values is $[0,1]$ and the larger value of AUC depicts that the model is more likely to rank the positive feedback items higher than negative feedback items. In SBPR and SFPMC which consider the social feedback too, large value of AUC depicts the extend of the model ranking positive feeback higher than social feedback and ranking social feedback higher than negative feedback.

The AUC equation used to evalute BPP and SBPR is as follows.

$$AUC = \frac{1}{|U|}\sum_{u\in U}\frac{1}{|P_u||N_u|}\sum_{i\in P_u}\sum_{j\in N_u} I(x_{i,u} > x_{j,u}) \quad (1)$$

Here I is the indicator function. Since in FPMC, FMC and SFPMC we also consider l (item last reviewed by user) in the score function, the AUC used to evaluate these two models is different from the previous equation and it is as follows.

$$AUC = \frac{1}{|U|}\sum_{u\in U}\frac{1}{|P_u||N_u|}\sum_{i\in P_u}\sum_{j\in N_u} I(x_{i,u,l} > x_{j,u,l}) \quad (2)$$

Baselines

For this predictive task, we used several baselines seen in class such as Bayesian Personalized Ranking (BPR), Social Bayesian Personalized Ranking (SPBR) introduced by Zhao et. al., 2014[2], Factorized Markov Chains (FMC) and Factorized Personalized Markov Chains (FPMC).

Notations used in the model descriptions.

T	Train set
U, I	user set, item set
u, i	user $u \in U$, item $i \in I$
$x_{i,u}$	predicted score for u, i pair
β_i	bias term for item i
γ_u^{UI}	latent representation for user u
γ_u^{IU}	latent representation for item i
l	item reviewed by user u prior to item i
$x_{i,u,l}$	predicted score for u, i given l as the last item
γ_i^{IL}	latent representation for item i
γ_l^{LI}	latent representation for item l
θ_i^{IL}	latent representation for item i
γ_l^{LI}	latent representation for item l
F_u	friends of user u
ψ_u	latent representation for user u
ϕ_i	latent representation for item i
α	hyperparameter weighting the influence of the friend set

Bayesian Personalized Ranking (BPR)

$$x_{i,u} = \beta_i + \gamma_u^{UI} \cdot \gamma_i^{IU} \quad (3)$$

$$f_{obj} = -\log\sigma(x_{i,u} - x_{j,u}) \quad (4)$$

BPR considers the affinity between user latent factor and item latent factor. The model is trained considering the above stated objective function f_{obj} where j is an item chosen at random such that u has not interacted with j. Here we define $x_{i,u}$ as the positive instance and $x_{j,u}$ as the negative instance.

Social Bayesian Personalized Ranking (SBPR)

$$x_{i,u} = \beta_i + \gamma_u^{UI} \cdot \gamma_i^{IU} \quad (5)$$

$$f_{obj} = -\sum_{u,i,k\in T}\log\sigma(x_{i,u} - x_{k,u}) - \sum_{u,k,j\in T}\log\sigma(x_{k,u} - x_{j,u}) \quad (6)$$

Here k is an item reviewed by the friend of user u. j again is an item u has not interacted with. The ranking function $x_{i,u}$ considers the affinities between user latent factor and item latent factor. The intuition here is to train the model so that $x_{u,i} > x_{u,k}$ and $x_{u,k} > x_{u,j}$. That is *positive instance* $>$ *social* and *social* $>$ *negative instance*.

Factorized Markov Chains (FMC)

$$x_{i,u,l} = \beta_i + \gamma_i^{IL} \cdot \gamma_l^{LI} \quad (7)$$

$$f_{obj} = -\log \sigma(x_{i,u,l} - x_{j,u,l}) \quad (8)$$

Here l denotes the previous item (just before i) reviewed by u. FMC uses compatibility of an item with a previous item to predict the ranking score.

Factorized Personalized Markov Chains (FPMC)

$$x_{i,u,l} = \beta_i + \gamma_u^{UI} \cdot \gamma_i^{IU} + \gamma_i^{IL} \cdot \gamma_l^{LI} \quad (9)$$

$$f_{obj} = -\log \sigma(x_{i,u,l} - x_{j,u,l}) \quad (10)$$

FPMC considers the compatibility between user, item latent factors as well as item and last reviewed item latent factors. Intuitively, this factorization simply states that the next item should be compatible with both the user and the previous item consumed.

III. The SFPMC Model

The SFPMC Model is an extension of FPMC (Factorized Personalized Markov Chains) Model. It was introduced by Cai. et. al, 2017[1]. It incorporates the social interactions between users to predict a score for a user, item pair. Given a user u, and an item i, that has not be reviewed by the user u, we define a score associated with u,i. l represents the latest reviewed item by user u before i. The score $x_{i,u,l}$ is given as follows

$$x_{i,u,l} = \gamma_u^{UI} \cdot \gamma_i^{IU} + \theta_i^{IL} \cdot \theta_l^{LI} + \frac{2}{|F_u|^\alpha} \sum_{u' \in F_u} \sigma(\psi_u \cdot \psi_{u'})(\phi_i \cdot \phi_{i'}) + \beta_i$$

F_u denotes the all the friends of user u. The summation term is carried out for every friend u' of u. $\psi_u, \psi_{u'}$ denote the latent representations of user u and friend u'. The product is normalized using the sigmoid function. Different friends can have different impact on a user. Hence the affinity between a user and a friend is measured by the product of their latent factors $\psi_u, \psi_{u'}$. i' denotes the latest reviewed item by friend u'. $\phi_i, \phi_{i'}$ denote the latent represenatations of items i, i'. β_i represents the bias term for item i.

Motivation

The Library thing dataset contains user, book interactions with timestamps along with the information about a users friends. In class we have explored models that take into account sequential information such as Markov Chain models and those that take into account social information such as Social and Group-based Bayesian Ranking. Hence we estimated that a model that combines both social and sequential information would be promising. We decided to leverage only on implicit feedback that is interactions and not exploit explicit feedback such as ratings and review texts since many real-life applications have only implicit information available. Considering the above concerns we arrived at the SFPMC model since it is socially as well as sequentially aware and only needs implicit interaction information.

Limitations

The SFPMC model we implemented took 21 minutes 8 seconds to complete 10 epochs. Since it iterates over all $\langle user, item \rangle$ instances and for each such instance, it iterates over all $\langle userfriend, item \rangle$ instances, it is relatively slow.

The hyperparameters in our model are α which weighs in the influence of the friend set, λ which is the regularizer term for the latent factor weights, K which is dimensionality of the latent factors and learning rate. Since the model is slow we couldn't tune in the hyperparameters optimally suited for the LibraryThing dataset and model. We implemented the model with $\alpha = 1$ as used by Cai et. al., 2017[1]. We used $K = 20$ as the dimensionality of latent factors. It is possible that with different K values for different latent factors, our model can train better. We used learning rate as 0.07. For the regularizer term λ we implemented the baselines for $\lambda = \{0.01, 0.001, 0.0001, 0.00001, 0.000001\}$. However, we used $\lambda = 0.000001$ for the SFPMC model.

The SFPMC model implemented by Cai et. al.,2017[1] uses the dataset Ciao, Foursquare, Epinions and Flixster. They have social connections to users ratio as 20.93, 4.72, 4.55 and 13.84 respectively while our dataset has the ratio of 1.845. Considering the cold-start statistics, that is considering only the users that have less than 5 interactions, the average social interaction count for LibraryThing is 2.058, which is again significantly lower than other datasets: Ciao, Delicious and Epinions, which have a ratio of 35.95, 8.14, 3.369 respectively as per Zhao et. al, 2014. Hence the SFPMC model could be at a slight disadvantage when evaluated on the LibraryThing dataset.

Unsuccessful Attempts

We intially modeled the SFPMC ranking function as follows.

$$x_{i,u,l} = \gamma_u^{UI} \cdot \gamma_i^{IU} + \theta_i^{IL} \cdot \theta_l^{LI} + \frac{2}{|F_u|^\alpha} \sum_{u' \in F_u} \sigma(\psi_u \cdot \psi'_{u'})(\phi_i \cdot \phi'_{i'}) + \beta_i$$

We considered separate latent factor representations for user and friends, ψ, ψ'. We also considered separate latent factor representations for item and friend's item $\phi_i, \phi'_{i'}$. Since the model now had increased latent matrices to learn, the model had an extremely slow convergence rate. We decided to instead model it as by Cai. et. al.[1].

Model Comparison

The SFPMC model leverages on both social as well as sequential information. The other baselines include only either one of them. For example, Social BPR only includes the social information, while other Markov Chain based approaches include only sequential information. [2]

IV. Literature Review

LibraryThing in Earlier Works

LibraryThing is popular library catalog website, where users can rate books and share their reviews. It also allows poeple to connect to other readers with similar interests. Our dataset, including user, book reviews and social circles of a user has been extracted from this website. The dataset has been used extensively for evaluating different recommendation models. Zhao et. al, 2014[2] used the LibraryThing dataset along with other datasets, Ciao, Delicious and Epinions to evaluate the Social BPR model proposed in their work. The other datasets are also user, item reviews along with social information. The SBPR model was built on the assumption that: rank of items a user consumes > items user's friends consume > items neither user not friends consume. The work compared SBPR to other models such as BPR and 'Group-based' BPR introduced by Pan and Chen, 2016[3]. The evaluation metric used was AUC along with Recall@K and Normalized Discounted Cumulative Gain (NDCG). The work showed that the Social BPR effectively improved the recommendation accuray.

The LibraryThing dataset was also used by Zhao et. al., 2015[4] to build a personalized feature projection method to model users' preferences over items.

The SFPMC model implemented in the assignment was proposed by Cai. et. al, 2017[1]. It was implemented on four different datasets Ciao, Foursquare, Epinions, and Flixster, all of them collected from different user review websites. Cai et. al. compared the SFPMC model against different baselines such as BPR-MF, Social BPR, Group-based BPR and FPMC. The evaluation metric used was AUC (area under the curve). The work went on to show that the SFPMC model showed improved AUCs when compared to the baselines.

Deep Neural Network Based Approaches

Markov Chains assume that the next user, item interaction is dependent on only the previous interaction, however there have been several works exploiting Recurrent Neural Networks that use the summary of all previous interactions. Kang and McAuley, 2018[5] proposed self-attention based sequential model based on RNNs for next item recommendation. Their work shows that the attention-based model outperforms baselines such as popularity based recommendation, BPR, FMC, FPMC. Other works have also employed deep neural networks for one-class reccommendation. Sun et. al, 2020[6] incorporated attentive graph convolutional layers to explore high-order relationships in the user-item bipartite graph and dynamically capture the latent tendencies of users toward the items they interact with.

[2]SFPMC Source Code

Comparison with Literary Works

Cai et. al, 2017 [1]'s implementation of SFPMC model showed that leveraging social interactions along with sequential information improves recommendation accuracy. We see this finding in our implementation as well. From Table II we see that Social BPR improves the AUC as compared to BPR and SFPMC has the highest AUC among all other models.

V. Results and Conclusion

In this section, we present the results of our experiments with the BPR, SBPR, FPMC, FMC and SFPMC models in terms of AUC values on the validation and test dataset, which are both 10% of the total LibraryThing dataset taken. The results of the models on the test and validation datasets are summarized in Table II and Figure I.

TABLE II
Model Statistics.

Model	*Test AUC*	*Validation AUC*
BPR	0.58865	0.60295
SBPR	0.62483	0.63044
FMC	0.61179	0.62649
FPMC	0.61211	0.63104
SFPMC	0.66479	0.67721

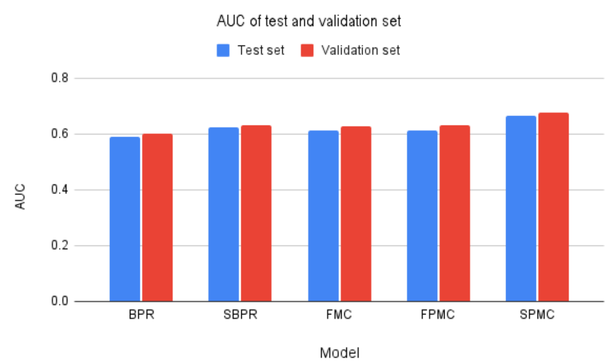

Fig. 1. This figure shows the AUC values obtained by all the implemented models (BPR, SBPR, FMC, FPMC, SFPMC) on the test and validation sets.

Results Comparison

From Table II and Figure I, we observe that the SFPMC model gave the best results with an AUC of 0.66479 on the test dataset, while the BPR model performed the least, giving an AUC of 0.58865 on the test dataset. We notice that the SBPR model, which also takes into account the social connections of the users, performed better than the BPR model and gave an AUC of 0.62483. An interesting observation is that the SBPR performed better than both FMC and FPMC datasets, which use the users' sequential feedback information for the ranking recommendations. We also notice a very slight improvement in the AUC on the test dataset when we compare FMC with FPMC. This could be because FPMC includes the $user/item$ interaction component in addition to the $item/item$ interaction in the score function when compared with FMC.

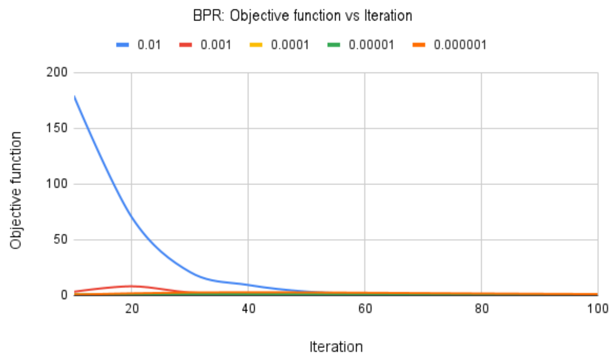

Fig. 2. This figure shows how the loss function varied as the training progressed for the BPR model for different values of λ taking K=20 and NSamples=100000.

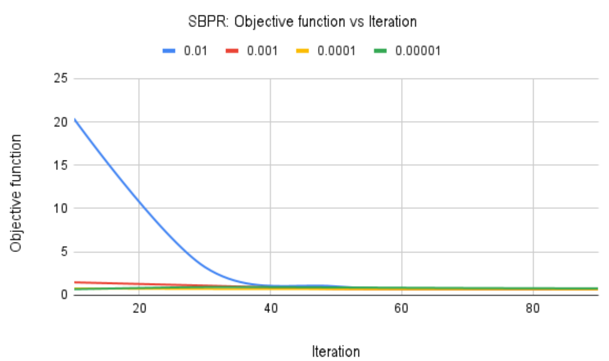

Fig. 3. This figure shows how the loss function varied as the training progressed for the SBPR model for different values of λ taking K=20 and NSamples=100000.

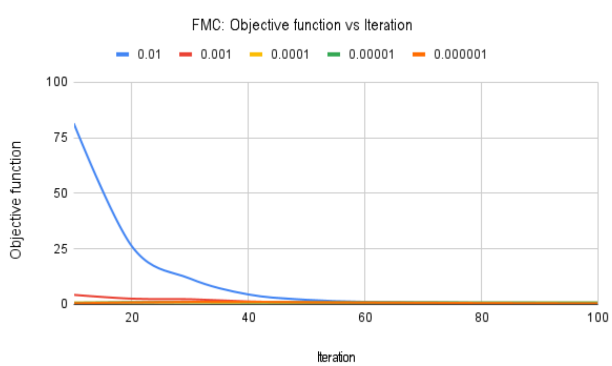

Fig. 4. This figure shows how the loss function varied as the training progressed for the FMC model for different values of λ taking K=20 and NSamples=100000.

Parameter Interpretation

We tuned the λ regularizer to 0.000001 after experimenting with the following different values $\lambda = \{0.01, 0.001, 0.0001, 0.00001, 0.000001\}$ for BPR, SBPR, FMC, FPMC using the validation dataset for the best results. We trained all the models for 100 iterations and the below plots visualize the varying of the loss function with the number of iterations for different values of lambda for each of the implemented models on the training set of the LibraryThing dataset. We notice that all the models have similar learning behaviour and the initial value of the objective function is highest when λ = 0.01 for all the implemented models. Since our dataset is sufficiently large (497306 interactions after preprocessing), we took K=20 instead of a smaller value.

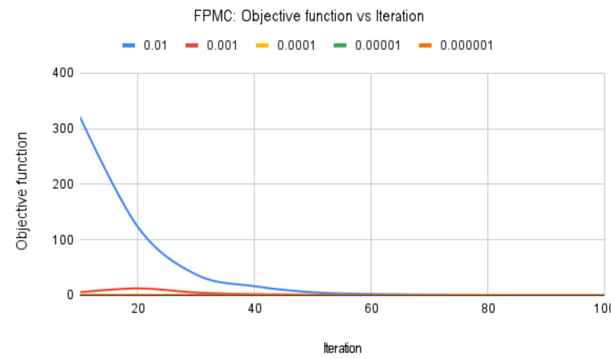

Fig. 5. This figure shows how the loss function varied as the training progressed for the FPMC model for different values of λ taking K=20 and NSamples=100000.

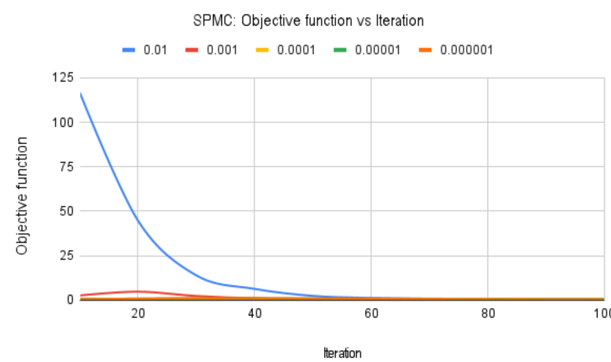

Fig. 6. This figure shows how the loss function varied as the training progressed for the SFPMC model for different values of λ taking K=20 and NSamples=100000.

REFERENCES

[1] C. Cai, R. He, and J. McAuley, "Spmc: Socially-aware personalized markov chains for sparse sequential recommendation," 2017.

[2] T. Zhao, J. McAuley, and I. King, "Leveraging social connections to improve personalized ranking for collaborative filtering," 2014.

[3] W. Pan and L. Chen, "Group bayesian personalized ranking with rich interactions for one-class collaborative filtering," 2016.

[4] T. Zhao, J. McAuley, and I. King, "Improving latent factor models via personalized feature projection for one class recommendation," 2015.

[5] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," 2018.

[6] J. Sun, Y. Zhang, C. Ma, M. Coates, H. Guo, R. Tang, and X. He, "Multi-graph convolution collaborative filtering," 2020.