

# TOCCF: Time-Aware One-Class Collaborative Filtering

Xinchao Chen, Weike Pan\* and Zhong Ming\*  
College of Computer Science and Software Engineering, Shenzhen University  
xcchen1993@gmail.com, {panweike, mingz}@szu.edu.cn  
\*: corresponding author.

## ABSTRACT

One-class collaborative filtering (OCCF), or recommendation with one-class feedback such as shopping records, has recently gained more attention from researchers and practitioners in the community. The main reason is that one-class feedback in the form of (user, item) pairs are often more abundant than numerical ratings in the form of (user, item, rating) triples as exploited by traditional collaborative filtering algorithms. However, most of the previous work on OCCF do not consider the temporal context, which is known of great importance to users' preferences and behaviors. In this paper, we first formally define a new problem called time-aware OCCF (TOCCF), and then design a novel time-aware similarity learning (TSL) model accordingly. Our TSL is based on a novel time-aware weighting scheme and a seminal work on similarity learning, and is able to learn the item similarities more accurately. Empirical studies on two large real-world datasets show that our TSL model can integrate the temporal information effectively, and perform significantly better than several state-of-the-art recommendation algorithms.

## CCS Concepts

•Information systems → Personalization; •Human-centered computing → Collaborative filtering;

## Keywords

Time-Aware; One-Class Collaborative Filtering; Factored Item Similarity Model

## 1. INTRODUCTION

One-class collaborative filtering (OCCF) [2, 5] is a recent research focus in the community of recommender systems. In OCCF, the data we can exploit for recommendation are the so-called one-class feedback such as “transactions” in e-commerce instead of multi-class feedback or numerical ratings in traditional collaborative filtering problems. The

reason that modeling one-class feedback is considered more important is simply due to the fact that users are somehow reluctant to assign a multi-class score to a product after purchasing.

In order to model the one-class feedback, two main lines of techniques are usually adopted, which are parallel to that of collaborative filtering, including memory-based OCCF and model-based OCCF. For memory-based OCCF, the only difference from that of memory-based CF is that the similarity between two users or two items are estimated based on the one-class feedback instead of the ratings. For model-based OCCF, the techniques are often different from that of model-based CF, in particular of the underlying assumption for the learning task of positive feedback only and the prediction rule based on similarity learning. The most well-known preference assumption for one-class feedback is probably the pairwise preference assumption called Bayesian personalized ranking defined on the difference between a purchased product and an un-purchased one [7]. And the most recent work on similarity learning approach is the factored item similarity model (FISM) [2], which learns the latent representation of items with the assumption that the inner product of two items' latent factors is their similarity.

The aforementioned advances in modeling one-class feedback in OCCF have indeed achieved great success in various recommendation applications. However, we find that very few work have explicitly studied the temporal effect in OCCF, though it has shown to be very helpful in user behavior modeling in CF [3, 4]. A recent work on microblog recommendation [8] shows that temporal information is helpful. However, the time-aware weighting scheme [8] is designed for the specific application, where the items (or tweets) are recorded with time when they arrive at the users instead of when they are retweeted by the users. In the studied general time-aware OCCF, we only have the temporal information when users have actions to items.

In this paper, we first design a time-aware weighting scheme for the *reliability* of the positive feedback, and then propose a *time-aware similarity learning* (TSL) model by integrating the weight as a confidence score into the similarity learning model [2]. The time complexity of TSL is the same with that of FISM [2]. We conduct extensive empirical studies on two public large datasets with the state-of-the-art baselines of memory-based methods and model-based methods. The empirical results show that our new similarity learning model is simple but very effective in exploiting the time context, and is significantly better than the algorithms without modeling the temporal effect.

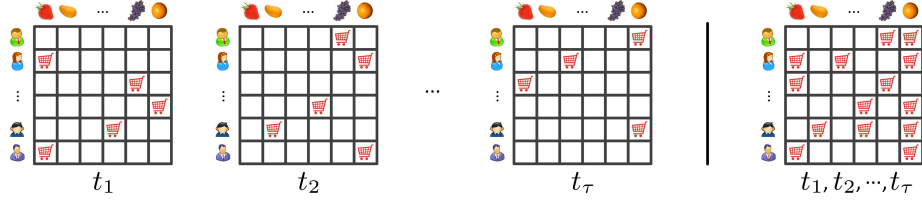


Figure 1: An illustration of time-aware one-class collaborative filtering (TOCCF) and OCCF.

## 2. TIME-AWARE SIMILARITY LEARNING

### 2.1 Problem Definition

In time-aware one-class collaborative filtering (TOCCF), we have  $n$  users,  $m$  items and their positive feedback in the form of (user, item, time) triples, e.g.,  $(u, i, t_{ui})$ , denoting user  $u$  has a positive feedback on item  $i$  at time  $t_{ui}$ . In TOCCF, our goal is to learn users' preferences from the positive feedback and associated temporal information, and provide a personalized ranked list of items for each user  $u$  that he or she may like in the future.

Notice that in OCCF [5], the temporal information is not exploited, i.e., the data is of (user, item) pairs. We illustrate the studied problem in Figure 1, where OCCF is a special case of TOCCF and is represented as a mixed (user, item) feedback matrix ignoring the time context.

Table 1: Some notations.

$\mathcal{T} = \{u, i, t_{ui}\}$	positive feedback
$\mathcal{T}_u$	positive feedback of user $u$
$\mathcal{T}'$	sampled unobserved feedback
$d \in \mathbb{R}$	latent feature number
$V_i, W_{i'} \in \mathbb{R}^{1 \times d}$	latent feature vectors
$b_u \in \mathbb{R}$	user bias
$p_i \in \mathbb{R}$	item bias
$r_{ui} \in \{1, 0\}$	preference of $(u, i)$
$\hat{r}_{ui}$	prediction of $(u, i)$
$\alpha$	tradeoff parameter
$T$	iteration number

### 2.2 Similarity Learning

It is well known that similarity measurement is critical in collaborative filtering, because it determines the neighborhood of a certain user  $u$  and thus affects the preference prediction of user  $u$  on other items. The state-of-the-art approach [2] does not adopt traditional similarity measurement such as Cosine similarity or Jaccard index, but turns to learn the similarity from the preference data, which is empirically more adaptive to different datasets. Mathematically, the learned similarity in FISM [2] is represented as follows,

$$s_{i'i} = \frac{1}{\sqrt{|\mathcal{N}_i \setminus \{i\}|}} V_i \cdot W_{i'}^T, \quad (1)$$

where  $\mathcal{N}_i = \{i' | (u, i', t_{ui'}) \in \mathcal{T}_u\}$ , and  $V_i, W_{i'} \in \mathbb{R}^{1 \times d}$  are latent feature vectors to be learned for item  $i$  and item  $i'$ , respectively. The similarity  $s_{i'i}$  or the latent vectors  $V_i$ .

and  $W_{i'}$  can be learned via some pointwise or pairwise loss functions in an optimization problem.

With the learned similarity  $s_{i'i}$ , the preference of user  $u$  on item  $i$  can then be predicted as follows [2],

$$\hat{r}_{ui} = \sum_{i' \in \mathcal{N}_u \setminus \{i\}} s_{i'i} + b_u + p_i, \quad (2)$$

where  $b_u$  and  $p_i$  are preference bias of user  $u$  and popularity bias of item  $i$ , respectively.

### 2.3 Time-Aware Similarity Learning

We introduce a confidence measurement for an observed positive feedback  $(u, i, t_{ui})$ ,

$$c_{ui} = \frac{1}{(t_\tau + 1) - t_{ui}}, \quad (3)$$

where  $t_\tau$  is the largest time stamp (in day) in the training data, and thus  $(t_\tau + 1)$  is used to denote the current day. Notice that we use the inverse of the difference between the current time  $(t_\tau + 1)$  and the time the positive feedback is issued  $t_{ui}$  because a more recent feedback is more reliable, and is thus of high confidence.

Our proposed confidence measurement shown in Eq.(3) looks similar to but is very different from the *pairwise* confidence weight in BPRC [8], which is defined on two (user, item) pairs, i.e.,  $c_{uij}$  for  $(u, i)$  and  $(u, j)$ , instead of on one single (user, item) pair in Eq.(3). Notice that in BPRC [8],  $(u, i)$  denotes a retweet feedback and  $(u, j)$  denotes a non-retweet feedback, while both are associated with temporal information of when the tweet is received by user  $u$ . In our TOCCF, we only have the temporal information of the observed positive feedback, and thus the approach in [8] is also not applicable to our studied problem.

Finally, we have a general time-aware weighting scheme,

$$\omega_{ui} = \begin{cases} c_{ui}, & \text{if } (u, i, t_{ui}) \in \mathcal{T}_u, \\ 1, & \text{otherwise,} \end{cases} \quad (4)$$

which means that we will weight the known (i.e., observed feedback) only. It is thus a *pointwise* confidence weight.

With the time-aware weighting scheme and the loss function of FISM [2], we propose to solve the following optimization problem,

$$\min_{\mathbf{V}, \mathbf{W}, \mathbf{b}, \mathbf{p}} \sum_{(u, i, t_{ui}) \in \mathcal{T} \cup \mathcal{T}'} \frac{1}{2} \omega_{ui} (r_{ui} - \hat{r}_{ui})^2 + \mathcal{R}(\mathbf{V}, \mathbf{W}, \mathbf{b}, \mathbf{p}), \quad (5)$$

where  $\mathcal{T}'$  is a set of randomly sampled unobserved feedback with  $|\mathcal{T}'| = 3|\mathcal{T}|$ ,  $\mathcal{R}(\mathbf{V}, \mathbf{W}, \mathbf{b}, \mathbf{p}) = \frac{\alpha}{2} \sum_{i=1}^m \|V_i\|^2 + \frac{\alpha}{2} \sum_{i'=1}^m \|W_{i'}\|^2 + \frac{\alpha}{2} \sum_{u=1}^n b_u^2 + \frac{\alpha}{2} \sum_{i=1}^m p_i^2$  is the regularization term commonly used to avoid overfitting. The optimization problem can be solved in a commonly used gradient descent algorithm [2].

**Table 2: Recommendation performance of TSL, FISM, BPR, ICF with Cosine similarity, i.e., ICF(CS), and Jaccard index, i.e., ICF(JI), on ML10M and Netflix using Precision@5, Recall@5, F1@5, NDCG@5 and 1-call@5. The best results are marked in bold. We also include the searched value of the tradeoff parameter and iteration number in model-based methods for reproductivity.**

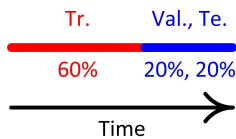
Dataset	Method	Parameter		Evaluation metric				
		$\alpha$	$T$	Precision@5	Recall@5	F1@5	NDCG@5	1-call@5
ML10M	ICF(JI)			0.0163	0.0052	0.0064	0.0188	0.0720
	ICF(CS)			0.0158	0.0050	0.0062	0.0182	0.0697
	BPR	0.01	100	0.1186	0.0437	0.0525	0.1305	0.4021
	FISM	0.001	1000	0.1169	0.0418	0.0509	0.1270	0.4029
	TSL	0.01	500	<b>0.1391</b>	<b>0.0533</b>	<b>0.0633</b>	<b>0.1442</b>	<b>0.4552</b>
Netflix	ICF(JI)			0.0333	0.0203	0.0196	0.0386	0.1247
	ICF(CS)			0.0333	0.0206	0.0198	0.0385	0.1258
	BPR	0.001	500	0.0518	0.0219	0.0233	0.0561	0.1878
	FISM	0.001	1000	0.0514	0.0237	0.0242	0.0563	0.1937
	TSL	0.001	100	<b>0.0628</b>	<b>0.0409</b>	<b>0.0365</b>	<b>0.0730</b>	<b>0.2451</b>

### 3. EXPERIMENTAL RESULTS

#### 3.1 Datasets and Evaluation Metrics

In order to verify the effectiveness of our proposed pointwise weighting scheme and time-aware similarity learning (TSL) model, we use two large public datasets, i.e., MovieLens 10M (we use ML10M for short) and Netflix, in our empirical studies. ML10M contains about 10 million numerical ratings from 71567 users and 10681 items, and Netflix contains about 100 million numerical ratings from 480189 users and 17770 items. In order to simulate the TOCCF problem setting with time-aware positive feedback, for each dataset, we first remove the  $(u, i, r_{ui}, t_{ui})$  quadruples with  $r_{ui} \leq 4$ , and then take the  $(u, i, t_{ui})$  triples from the remaining data.

For the resulted time-aware one-class feedback of each dataset, we further split it according to the time stamp in order to generate a copy of training data, validation data and test data. We illustrate the data generation procedure in Figure 3. Specifically, we first use 60% feedback with the smallest time stamps for training; and then from the left 40% feedback, we randomly sample 20% feedback for validation and the remaining 20% for test. We put the statistics of the resulted datasets in Table 3.



**Figure 3: An illustration of data generation for training data, validation data and test data, where training data are from the first 60% one-class feedback, and validation and test data are from the remaining 40% feedback.**

For one-class feedback in TOCCF, we adopt several commonly used evaluation metrics in ranking-oriented item recommendation or information retrieval scenarios. In particular, we check the top- $K$  performance using Precision@ $K$ , Recall@ $K$ , F1@ $K$ , NDCG@ $K$  and 1-call@ $K$ .

**Table 3: Description of the datasets used in the experiments, including the numbers of users ( $n$ ), items ( $m$ ), training feedback ( $|T_r|$ ), validation feedback ( $|V_{ai}|$ ), and test feedback ( $|T_e|$ ).**

Dataset	$n$	$m$	$ T_r $	$ V_{ai} $	$ T_e $
ML10M	71567	10681	1277900	425967	425967
Netflix	480189	17770	13900939	4633646	4633647

#### 3.2 Baselines and Parameter Settings

In our empirical studies, we include several state-of-the-art methods for modeling one-class feedback in recommender systems, including neighborhood-based methods and factorization-based methods:

- ICF(JI): item-oriented neighborhood-based collaborative filtering with Jaccard index as the similarity measurement.
- ICF(CS): ICF with Cosine similarity. Both ICF(JI) and ICF(CS) are simple but effective approaches for OCCF.
- BPR: Bayesian personalized ranking [7]. BPR is a seminal work based on pairwise preference assumption and usually produces the best performance in different recommendation tasks.
- FISM: Factored item similarity models [2]. FISM is one of the most recent work based on pointwise preference assumption and similarity learning.

For neighborhood-based methods ICF(JI) and ICF(CS), we fix the size of nearest neighbors as 20. For factorization-based methods BPR, FISM and our TSL, we fix the dimension of latent space as  $d = 20$  and the learning rate  $\gamma = 0.01$ . The iteration number in BPR, FISM and our TSL are chosen from  $T \in \{100, 500, 1000\}$  and the value of tradeoff parameters are chosen from  $\alpha \in \{0.001, 0.01, 0.1\}$  all through the NDCG@5 on the validation data, i.e., there are nine combinations of the value of the two types of parameters.

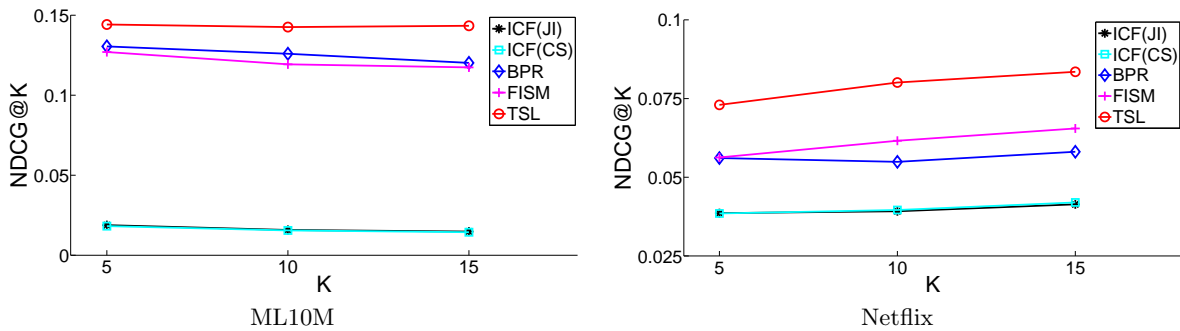


Figure 2: Recommendation performance of TSL and other methods on NDCG@K with different value of  $K$ .

### 3.3 Main Results

We report the main results in Table 2, from which we can have the following observations,

- Two neighborhood-based methods, i.e., ICF(JI) and ICF(CS), are poor regarding the recommendation performance, which is caused by the intransitivity of the similarity measurements for the scarce positive feedback. Notice that the density of the training data of ML10M and Netflix are smaller than 0.2%.
- Two factorization-based methods, i.e., BPR and FISM, perform much better than the neighborhood-based methods, which is expected because of the merit of transitivity via learned latent factors.
- Our proposed time-aware similarity learning method, i.e., TSL, further improves FISM and BPR significantly, from which we can clearly see the value of the temporal information and the effectiveness of our weighting scheme to integrate the time context.

For real-world deployment of a recommendation model, we usually pay more attention to its top- $K$  performance, because that will affect users' behaviors most. For this reason, we also check the recommendation performance with different value of  $K \in \{5, 10, 15\}$ . We show the results of NDCG@K in Figure 2. Notice that the results on other top- $K$  performance are similar, and are thus not included due to space limitation. From Figure 2, we can see that the performance ordering on different value of  $K$  over two datasets is ICF(JI), ICF(CS) < BPR, FISM < TSL, which is consistent to that of Table 2. The results on NDCG@K again show the usefulness of the temporal context and effectiveness of our time-aware weighting scheme in similarity learning.

## 4. CONCLUSIONS AND FUTURE WORK

In this paper, we study an important recommendation problem termed *time-aware one-class collaborative filtering* (TOCCF), and propose a novel *time-aware similarity learning* (TSL) model based on the seminal work of factored item similarity model [2]. Empirical results show that our TSL can incorporate the time information in a simple but effective way, and is able to recommend significantly more accurate ranked lists of items than several state-of-the-art methods without modeling the time information.

For future work, we are interested in generalizing our time-aware similarity learning model to more advanced similar-

ty learning approaches for recommendation with social and other auxiliary data [1, 6].

## 5. ACKNOWLEDGMENT

We thank the support of Natural Science Foundation of China (NSFC) No. 61502307, and Natural Science Foundation of Guangdong Province No. 2014A030310268 and No. 2016A030313038.

## 6. REFERENCES

- [1] H. Fang, G. Guo, and J. Zhang. Multi-faceted trust and distrust prediction for recommender systems. *Decision Support Systems*, 71:37–47, 2015.
- [2] S. Kabbur, X. Ning, and G. Karypis. Fism: Factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13*, pages 659–667, 2013.
- [3] Y. Koren. Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 447–456, 2009.
- [4] S. Liu, S. Wang, and F. Zhu. Structured learning from heterogeneous behavior for social identity linkage. *IEEE Transactions on Knowledge and Data Engineering*, 27(7):2005–2019, 2015.
- [5] R. Pan, Y. Zhou, B. Cao, N. N. Liu, R. Lukose, M. Scholz, and Q. Yang. One-class collaborative filtering. In *Proceedings of the 8th IEEE International Conference on Data Mining, ICDM '08*, pages 502–511, 2008.
- [6] W. Pan. A survey of transfer learning for collaborative recommendation with auxiliary data. *Neurocomputing*, 177:447–453, 2016.
- [7] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence, UAI '09*, pages 452–461, 2009.
- [8] S. Wang, X. Zhou, Z. Wang, and M. Zhang. Please spread: Recommending tweets for retweeting with implicit feedback. In *Proceedings of the 2012 Workshop on Data-driven User Behavioral Modelling and Mining from Social Media, DUBMMSM '12*, pages 19–22, 2012.