Exploiting Temporal and Social Factors for B2B Marketing Campaign Recommendations

Jingyuan Yang*, Chuanren Liu†, Mingfei Teng*, Hui Xiong*, March Liao‡, Vivian Zhu‡
*MSIS Department, Rutgers University, Email: {jingyyan, mingfei.teng, hxiong}@rutgers.edu
†Decision Sciences & MIS Department, Drexel University, Email: chuanren.liu@drexel.edu
‡Citrix Systems, Inc., Email: {march.liao, qianqian.zhu}@citrix.com

Abstract*—Business to Business (B2B) marketing aims at meeting the needs of other businesses instead of individual consumers. In B2B markets, the buying processes usually involve series of different marketing campaigns providing necessary information to multiple decision makers with different interests and motivations. The dynamic and complex nature of these processes imposes significant challenges to analyze the process logs for improving the B2B marketing practice. Indeed, most of the existing studies only focus on the individual consumers in the markets, such as movie/product recommender systems. In this paper, we exploit the temporal behavior patterns in the buying processes of the business customers and develop a B2B marketing campaign recommender system. Specifically, we first propose the temporal graph as the temporal knowledge representation of the buying process of each business customer. The key idea is to extract and integrate the campaign order preferences of the customer using the temporal graph. We then develop the low-rank graph reconstruction framework to identify the common graph patterns and predict the missing edges in the temporal graphs. We show that the prediction of the missing edges is effective to recommend the marketing campaigns to the business customers during their buying processes. Moreover, we also exploit the community relationships of the business customers to improve the performances of the graph edge predictions and the marketing campaign recommendations. Finally, we have performed extensive empirical studies on real-world B2B marketing data sets and the results show that the proposed method can effectively improve the quality of the campaign recommendations for challenging B2B marketing tasks.

***Keywords*—Recommender Systems, Temporal Patterns, Temporal Graph, Graph Reconstruction, Community Networks**

I. Introduction

In Business-to-Business (B2B) markets, the professional marketers need to know how to provide the most relevant and appropriate marketing campaigns to the right customers. Indeed, the right marketing campaigns, if adopted at the right time, can help to increase the customer conversion ratio, which is the percentage of customers who eventually sign the business contracts. The right marketing campaigns also help to expedite the conversion cycle and boost the profits. Therefore, given the growing large number of marketing campaigns, there is a critical need for a B2B marketing campaign recommender system to improve the marketing performances.

However, developing the B2B marketing campaign recommender systems is a nontrivial task. In B2B markets, the buying processes usually involve series of different marketing campaigns providing information to multiple decision makers with different interests and motivations. These processes are naturally dynamic and complex. As a result, it is important to consider the behavior patterns in the buying processes for designing the recommender system so as to meet the dynamically changing needs of the business customers.

More specifically, there are two unique challenges for providing B2B marketing campaign recommendations. First, the business customers often interact with different campaigns based on their current status for most relevant information, which is necessary for them to make further decisions. Therefore the temporal information, especially the temporal correlation/dependency among the marketing campaigns, is very useful for modelling the customer behaviors. In other words, temporal-aware models are essential for providing effective marketing campaign recommendations. However, the implicit temporal information is hidden in the noisy buying processes and we need robust statistics to capture the temporal dynamics. Second, for B2B marketing, there are usually multiple decision makers from the same business/company evaluating the potential products or services from different aspects. Accordingly, the individuals/customers from the same company can form a community, where each individual can have impact on the behavior preferences of others in the same community. Thus, to provide recommendations potentially expediting the overall conversion cycle, we need to consider not only each individual's own status, but also the overall behavior preferences of his/her colleagues in the same community.

To address these challenges, we consider the problem to recommend the "Next Campaign To Run" (**NCTR**) to the business customers, by incorporating both the customer behavior records and the community structure of the customers. In the literatures, there have been some related work on the "next-item" recommendation. For example, a factorization framework of Markov chains (FPMC) is proposed by Rendle et al. [19], Cheng et al. [5]. The idea is to transform the sequential data of each user into a transition matrix and then predict the user's next action by factorizing the matrices. Yap et al. [26] search the personalized sequential patterns for next-item recommendation. In addition, Zhao et al. [28, 29] consider the temporal intervals between the purchase behaviors to increase the temporal diversity in the recommendations. Generally s-peaking, these approaches first extract the temporal knowledge to capture the temporal dynamics of user's preference and then integrate the knowledge in the recommender system. Although these methods have been successfully applied in the B2C (Business-to-Consumer) markets, they are not designed for our B2B marketing scenarios. First, the temporal dynamics of consumer users' preferences on different products are different with the business customers' need on B2B campaigns. In particular, different campaigns reveal different levels/facets of information about the same B2B buying process, but such evolving levels/facets are rare in the consumer products. Sec-

1550-4786/15 $31.00 © 2015 IEEE
DOI 10.1109/ICDM.2015.71

ond, the existing temporal knowledge representations (such as Markov transition matrix and personalized sequential patterns) are not resistant to noisy behavior records in the complex buying processes. For example, the business customer may occasionally participate in campaigns irrelevant to the context in the process but this kind of random event can dramatically affect the simple Markov transitions and sequential patterns. Third, only a scarce of studies have integrated both the behavior records and the community structures within a unified recommender system.

In this paper, to solve the NCTR problem for B2B marketing campaign recommendations, we first propose the temporal graph as the temporal knowledge representation of the buying process of each business customer. The key idea is to extract and integrate the campaign order preferences of the customer using the temporal graph. We then develop the low-rank graph reconstruction framework to identify the common graph patterns and predict the unobserved edges in the temporal graphs. We show that the prediction of the unobserved graph edges is effective to recommend the marketing campaigns to the business customers during their buying processes. In addition, to exploit the community structure of the business customers for better graph edge prediction and marketing campaign recommendation, we define effective regularizers for the low-rank temporal graph reconstruction. Finally, we have performed extensive experiments on real-world B2B marketing data. The results show that the proposed method can effectively improve the quality of the campaign recommendations for challenging B2B marketing tasks.

II. Problem Formulation

We recommend the "Next Campaign To Run" (NCTR) for the business customers by modelling their historical behavior records and the community structure of the customers. Figure 1 shows some examples of the behavior records. Specifically, there are three campaign sequences for three customers $C1, C2$, and $C3$ from two companies. For the campaign sequences, we also have the event-happening time when the customer participated in the campaign, therefore we are able to compute the intervals between consecutive events. The intervals can reveal interesting relationship (e.g., the temporal correlations) between the dependent marketing campaigns. For example, both customer $C1$ and $C2$ downloaded the trial product two or three days after attending a webinar. If these similar patterns are followed by the majority of the customers, we could recommend to download trial products to the customers who attended webinars about 2-3 days ago. Intuitively, recommending the dependent campaigns according to the customer's current context in the decision-making process will expedite the buying cycle. Therefore, *our main objective is to exploit the temporal patterns in the behavior records of all customers for providing accurate marketing campaign recommendations*.

Moreover, we have the community structure of the customers. In the B2B markets, there are often multiple customers from the same company working together to make decisions in the buying processes. Accordingly, these customers often form a community by communicating with each other and behaving collaboratively. In Figure 1, $C2$ and $C3$ are from the same company and we can see these two customers followed similar behavior patterns (e.g., both of them first participated in a email campaign and then downloaded the trial products with similar temporal intervals). In this paper, we exploit such a community structure to improve the "next campaign to run" recommendations for all the business customers.

III. A NCTR Recommender System

This section presents our NCTR recommender system, including: 1) A novel graph-based representation to encode the temporal information in the customer behavior records; 2) A low-rank graph reconstruction approach to predict the unobserved graph edges which can be used for NCTR recommendations; 3) A regularization of the graph reconstruction to incorporate the community structure of the customers; 4) A stochastic gradient descent learning algorithm to optimize the regularized graph reconstructions.

A. Temporal Knowledge Representation

Our first step to develop the NCTR recommender system is to design the informative representation of the temporal knowledge hidden in the behavior records of each customer. Inspired by Liu et al. [11], we propose the personalized temporal graph which effectively encodes the temporal relationships of the campaigns participated in by each customer.

Suppose we have M campaigns under study. For one specific customer, e.g., the n-th customer, we have his/her behavior records which are represented as a sequence of campaigns $s^n = (s_1^n, s_2^n, \cdots, s_{L_n}^n)$, where $s_l^n \in \{1, 2, \cdots, M\}$ is the l-th campaign in the sequence. We also record the campaign-participating time t_l^n for s_l^n. With these notations, we define the personalized temporal graph G^n for the n-th customer, with all the M campaigns as graph nodes. The direct edge from the i-th node to the j-th node is weighted by:

$$R_{ij}^n = \frac{1}{L_n} \sum_{1 \le p \le q \le L_n} [s_p^n = i \wedge s_q^n = j] \kappa(t_q^n - t_p^n), \quad (1)$$

where $\kappa(\cdot)$ is a non-increasing function.

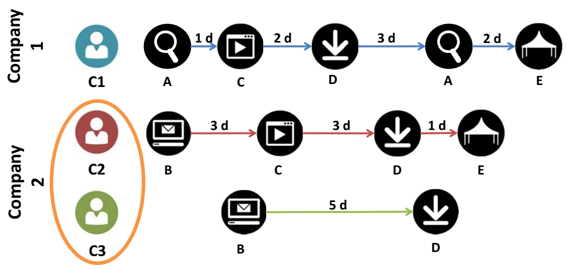
The non-increasing property of the function $\kappa(\cdot)$ enables us to compute a higher edge weight R_{ij}^n if the i-th and j-th campaigns appear close to each other in s^n. For example, we can use the simple Iverson bracket:

$$\kappa(\delta|\Delta) = [\delta \le \Delta], \quad (2)$$

where Δ is a threshold. In this way, we assume the marketing events happened within the temporal range of Δ are temporally related for this customer. An appropriate value of Δ can be thus determined with the domain knowledge in a particular application. More generally, we can also use a smooth function to further discriminate different temporal intervals between the events. In this paper, we use the truncated exceedance of the Exponential distribution:

$$\kappa(\delta|\Delta, r) = \begin{cases} \exp(-\delta/r) & \delta \le \Delta \\ 0 & \delta > \Delta \end{cases}. \quad (3)$$

Here, we exclude the weight computing between events with relatively large time interval, e.g., larger than Δ, and a scaling parameter r is used to compute the remaining weights. We use this definition for three reasons:

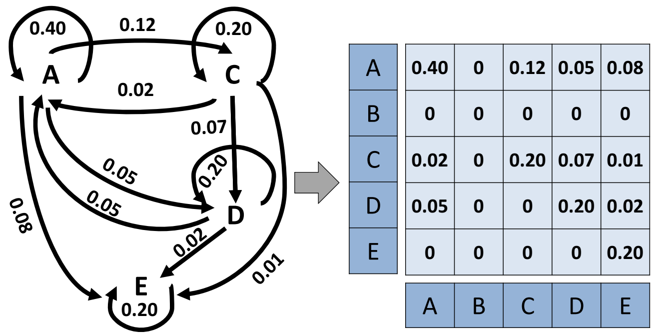


(a) The campaign sequences.

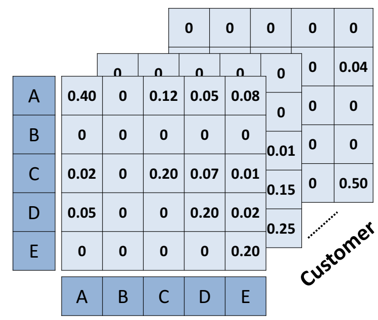
Code	Campaign Type	Icon
A	Global Search	
B	Email Campaign	
C	Webinar	
D	Trial Download	
E	Trade Show	

(b) The marketing campaigns.

Fig. 1: Examples of Customer Behavior Records.



(a) Example of transformation of the raw sequence into temporal graph.



(b) Personalized temporal graph.

Fig. 2: Examples of Personalized Temporal Knowledge Representation.

- According to the weight definition in Equation 1, the frequency of campaign i in the sequence s^n is included in R^n_{ii} (i.e., i-th diagonal entry), normalized by the sequence length L_n. These frequencies have been used in the design of conventional static recommender systems, while the temporal graphs extend the static frequencies with temporal correlations/dependencies.
- When $r \to +\infty$, Equation 3 and 2 are equivalent:

$$\lim_{r\to\infty} \kappa(\delta|\Delta, r) = \kappa(\delta|\Delta).$$

  The reason is that, when r is sufficiently large, each pair of events in the sequence s^n within the temporal range of Δ will be equally connected and weighted in the temporal graph G^n.
- When $r \to 0+$, we have:

$$\lim_{r\to 0+} \kappa(\delta|\Delta, r) = \begin{cases} 1 & \delta = 0, \\ 0 & \delta > 0. \end{cases}$$

  In this case, the graph weight matrix R^n with $\kappa(\cdot)$ defined in Equation 3 is almost diagonal, since very few distinct events happened at exactly the same time. Moreover, R^n_{ii} is exactly the normalized frequency of campaign i in the sequence s^n.

Therefore, our approach can be deemed a proper generalization of the conventional static recommender systems considering only the event frequencies as the implicit preferences/rating. For the same of simplicity, in the remaining of this paper, we let $\kappa(\delta) = \kappa(\delta|\Delta, r)$.

To provide an intuitive understanding, Example 1 shows the computation details of the personalized temporal graph. As can be seen, the graph-based representation translates the event sequences into the pairwise relationships, which captures the temporal closeness between any pair of campaigns. In the following, we utilize the personalized temporal graphs in our NCTR recommender system.

Example 1: We consider customer $C1$ in Figure 1. There are $L_n = 5$ behavior records. We let the scaling parameter $r = (1 + 2 + 3 + 2)/4 = 2$ and Δ is set to 90 days, then the personalized temporal graph R^{C1} can be constructed as shown in Figure 2a. The following are the calculations for the first row of the adjacency matrix:

$$\begin{aligned} R^{C1}_{AA} &= \frac{1}{5}(\exp(0) + \exp(0)) = 0.40, \\ R^{C1}_{AB} &= 0, \\ R^{C1}_{AC} &= \frac{1}{5}\exp(-\frac{1}{2}) = 0.12, \\ R^{C1}_{AD} &= \frac{1}{5}\exp(-\frac{3}{2}) = 0.05, \\ R^{C1}_{AE} &= \frac{1}{5}(\exp(-\frac{8}{2}) + \exp(-\frac{2}{2})) = 0.08. \end{aligned}$$

B. Recommendation with Temporal Graph

Suppose we have constructed the personalized temporal graphs for all customers with sufficient observations. Then for a specific customer with the last campaign i in his/her behavior records s^n, we can sort the campaigns $j = 1, 2, \cdots, M$ and

$j \neq i$ with respect to the values $R^n_{jj} \times R^n_{ij}$ in descending order. Here the two terms R^n_{jj} and R^n_{ij} computes the interest preference and the temporal preference, respectively. The campaigns ranked at the top will be recommended to the customer. However, it is expected that the constructed temporal graphs are very sparse with many edges unobserved, and the NCTR recommendation tasks rely on accurate prediction of the unobserved edges. In the next subsection, we develop the collaborative low-rank graph reconstruction approach to predict these unobserved graph edges.

C. Low-Rank Graph Reconstruction

Inspired by the popular matrix factorization [9] for predicting the unobserved customer-item ratings, we develop the low-rank graph reconstruction approach for predicting the unobserved edges in the personalized temporal graphs. The assumption is that, each observed temporal graph can be reconstructed by optimally combining a set of graph basis. To be specific, suppose we have N customers and constructed the temporal graph G^n for each $n = 1, 2, \cdots, N$. As introduced in Section III-A, each graph G^n is associated with the adjacency matrix $R^n \in \mathbb{R}^{M\times M}$, where M is the number of campaigns offered by the company. To reconstruct G^n, we assume there are K graph basis and each base graph is associated with a adjacency matrix $B^k \in \mathbb{R}^{M\times M}$ for $k = 1, 2, \ldots, K$. Then we use the graph basis to approximate the adjacency matrix R^n:

$$R^n = \sum_k A_{nk} B^k, \tag{4}$$

where A_{nk} is the reconstruction coefficients. Note that, the number of graph basis, K, can be deemed the rank of the graph reconstruction, which is set to be much smaller than the number of observed temporal graphs, N: $K \ll N$.

To compute the optimal graph basis B^k for $k = 1, 2, \cdots, K$, and at the same time the reconstruction coefficients in matrix A for all the observed temporal graphs G^n, $n = 1, 2, \cdots, N$, we can minimize the following reconstruction error:

$$\mathcal{J}(A, B) = \frac{1}{2} \sum_{n=1}^{N} \|R^n - \sum_{k=1}^{K} A_{nk} B^k\|_F^2, \tag{5}$$

where $\|\cdot\|_F$ denotes the Frobenius norm and we have constraints $A \geq 0$ and $B^k \geq 0$ for all k.

However, due to the sparsity of the temporal graphs, it is more efficient that we compute the reconstruction error with only the observed edges:

$$\mathcal{J}(A, B) = \frac{1}{2} \sum_{n=1}^{N} \|(R^n - \sum_{k=1}^{K} A_{nk} B^k) \odot I^n\|_F^2, \tag{6}$$

where $\odot$ is the Hadamard product operator, i.e., element-wise multiplication of matrices. The binary indicator $I^n_{ij} = 1$ if and only if we have the edge from i to j in graph G^n, i.e., $R^n_{ij} > 0$. Otherwise $I^n_{ij} = 0$.

An advantage of the low-rank graph reconstruction approach is that the formulation in Equation 6 can be conveniently extended. For example, inspired by the probabilistic matrix factorization [17], we can add regularization terms in the model to avoid overfitting:

$$\lambda_A \|A\|_F^2 + \lambda_B \sum_{k=1}^{K} \|B^k\|_F^2.$$

For the sake of simplicity, we do not include these terms since there are already extensive studies on this matter. Instead, in the following, we leverage the community network of the business customers to improve the performances of the low-rank graph reconstruction and the NCTR recommendation.

IV. Community Regularization

Indeed, in addition to modeling the temporal relationships of the marketing campaigns in the complicated decision-making process of the business customers, another important factor which can be leveraged to improve the NCTR recommendations is the community network of the customers. As we mentioned in Section I, in the B2B markets, it is often that multiple customers from the same company will make the business purchase decision together. These customers working on the same buying task or in the same company form a small community, where the members cooperate and communicate with each other. Therefore, during the reconstruction of their temporal graphs, these customers may share similar reconstruction coefficients. To integrate these constraints into our problem formulation, we adopt the so-called community regularization.

Suppose we have the community network encoded in the matrix H, where $H_{uv} = 1$ if and only if the two customers u and v are from the same company, and $H_{uv} = 0$ otherwise. Then our objective function is

$$\mathcal{J}(A, B) = \frac{1}{2} \sum_{n=1}^{N} \|(R^n - \sum_{k=1}^{K} A_{nk} B^k) \odot I^n\|_F^2 + \lambda \cdot \Omega(A), \tag{7}$$

where the community regularization $\Omega(A)$ encourages the customers from the same company to have similar reconstruction coefficients in A:

$$\begin{aligned}\Omega(A) =& \frac{1}{2} \sum_{u=1}^{N} \sum_{v=1}^{N} \frac{1}{2} H_{uv} \|A_u - A_v\|^2 \\ =& \frac{1}{2} \mathrm{tr}(A' L A),\end{aligned}$$

where $L = D - H$ and D is the diagonal graph degree matrix such that $D_{uu} = \sum_{v=1}^{N} H_{uv}$.

The level of the community regularization is controlled by the parameter λ. Specifically, a large λ will tend to make the campaign preferences of different customers to be the same in the same community. On the other hand, a small λ will tend to make the community network effects insignificant. In practice, the optimal λ is dependent on the actual data characteristics, and it can be realized by applying the cross validation procedure.

Moreover, the graph based regularization is quite flexible to encode different assumptions on the community networks. More generally, the graph structure in H can also be provided by domain experts or derived from external knowledge on the customer relationships. For example, we can derive a graph for B2B campaign recommender system based on the job title hierarchy of the companies.

V. Learning Algorithm

We present the implementation of the low-rank temporal graph reconstruction with regularizations, since the unregularized problem is a special case with $\lambda = 0$. We have two sets of parameters in A and B respectively, and we iteratively update them to minimize our objective function in Equation 7. We present the details in Algorithm 1.

Specifically, with B^k $(1 \leq k \leq K)$, fixed, we update A in a row-wise manner. For the n-th row, we have the subproblem:

$$\begin{aligned}
&\min_{A_{n*}} \frac{1}{2} \|(R^n - \sum_{k=1}^{K} A_{nk} B^k) \odot I^n\|_F^2 \\
&\quad + \lambda(\frac{1}{2} L_{nn} \|A_{n*}\|^2 + \sum_{n' \neq n} L_{nn'} \langle A_{n'*}, A_{n*} \rangle) \\
\Longleftrightarrow &\min_{A_{n*}} \frac{1}{2} \sum_{i,j:I_{ij}^n = 1} (R_{ij}^n - \langle A_{n*}, B_{ij}^* \rangle)^2 \\
&\quad + \lambda(\frac{1}{2} L_{nn} \|A_{n*}\|^2 + \sum_{n' \neq n} L_{nn'} \langle A_{n'*}, A_{n*} \rangle), \quad (8)
\end{aligned}$$

where A_{n*} is the n-th row in A, and we let vector $B_{ij}^* = (B_{ij}^1, B_{ij}^2, \cdots, B_{ij}^K)$. This subproblem is the so-called non-negative ridge regression fitting the data B_{ij}^* with the dependent variable R_{ij}^n for all campaign-pairs (i, j) such that $I_{ij}^n = 1$ or $R_{ij}^n > 0$. Since R is sparse, the problem scale (i.e., the number of campaign-pairs (i, j) with $R_{ij}^n > 0$) is small and the subproblem can be solved with little computing cost.

Then with A fixed, all the parameters in $B^k, 1 \leq k \leq K$ can also be updated simultaneously, but in an element-wise manner. To be specific, for a specific campaign-pair (i, j), we can update all $B_{ij}^k, 1 \leq k \leq K$ at the same time, with the following subproblem:

$$\begin{aligned}
&\min_{B_{ij}^*} \frac{1}{2} \sum_{n=1}^{N} ((R_{ij}^n - \sum_{k=1}^{K} A_{nk} B_{ij}^k) \times I_{ij}^n)^2 \\
\Longleftrightarrow &\min_{B_{ij}^*} \frac{1}{2} \sum_{n:I_{ij}^n = 1} (R_{ij}^n - \langle A_{n*}, B_{ij}^* \rangle)^2. \quad (9)
\end{aligned}$$

This subproblem is exactly non-negative regression fitting the data A_{n*} with dependent variable R_{ij}^n for all the customers with $I_{ij}^n = 1$ or $R_{ij}^n > 0$. Again, the problem scale (i.e., the number of customers with $R_{ij}^n > 0$) is small and the subproblem can be efficiently solved.

VI. Empirical Experiments

In this section, we evaluate the performances of our approach in comparison with several state-of-the-art methods. All the experiments are performed on a GNU/Linux system with 2 CPUs (AMD 2.4GHz) and 4G RAM.

Algorithm 1 NCTR with Community Regularization.

```
Initialize A, B with values uniformly drawn in (0, 1).
repeat
    for each customer n do
        update the n-th row A_{n*} by solving the subproblem
        in Equation 8.
    end for
    for 1 ≤ i ≤ M do
        for 1 ≤ j ≤ M do
            update B^k_{ij} for all 1 ≤ k ≤ K by solving the
            subproblem in Equation 9.
        end for
    end for
until Convergence
```

Characteristics	Data A	Data B	Data C
# of Customers	2,119	568	930
# of Companies	597	250	144
# of Distinct Campaign	116	90	125
Total Campaign Events	24,125	6,758	23,977
Average Time Interval (day)	17.41	19.49	16.88
Average Sequence Length	11.43	11.90	25.78

TABLE I: Characteristics of the B2B Marketing Data.

A. Data Description

We have collected the behavior records of the business customers interested in several products offered by a Fortune 500 company. For each customer, we collect the marketing campaigns participated in by the customer. The campaigns are ordered by the event-happening time as a marketing event sequence. A summary of all the data sets is shown in Table I with more statistics in Figure 3. One can see diverse characteristics in terms of data size (e.g., number of customers, number of companies, and number of campaigns) and event frequency (e.g., total campaign events, average time interval, and average sequence length).

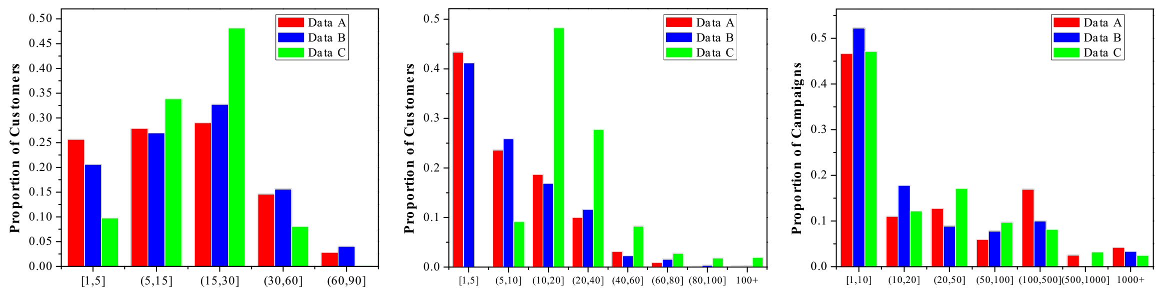
B. Evaluation Metrics

We use the following evaluation metrics to measure the recommendation performances:

- **Normalized Discounted Cumulative Gain (NDCG)**: The NDCG measures the ranking quality of the recommended list based on a graded relevance scale. It is widely used in researches on recommendation, information retrieval, search engine, etc. Specifically, for a ranking list with K items:

$$DCG@K = \sum_{k=1}^{K} \frac{2^{rel_k} - 1}{log(k+1)},$$
$$NDCG@K = \frac{DCG@K}{IDCG@K},$$

where $IDCG$ is the maximum possible DCG for the recommended items, and rel_i is the graded relevance of the list at position i. The range of NDCG is $[0, 1]$, with 1 representing the perfect ranking quality.



(a) Average time interval

(b) Sequence length

(c) Campaign frequency in all sequences

Fig. 3: Statistics of the B2B Marketing Data.

- **Precision and Recall**: For a ranking list with K items:

$$Precision@K = \frac{\#relevant\ recommendations}{K},$$
$$Recall@K = \frac{\#relevant\ recommendations}{\#all\ relevant\ items}.$$

The value of $Precision@K$ and $Recall@K$ closer to 1.0 means better recommendation performances.

In addition, we also compute the Root Mean Square Error (RMSE) to select parameters in our methods:

$$RMSE = \sqrt{\frac{1}{N}\sum_{n}\|(R^n - \widehat{R}^n)\odot I^n\|_F^2},$$

where R^n denotes the customer n's temporal graph, $\widehat{R}^n$ denotes the reconstructed approximation, and N is the number of customers in test set.

In our experiments, we use the first 65% of the behavior records of each customer for training and the remaining 35% for testing. All the metrics are computed for each customer and then the overall average is summarized to compare different methods or parameter settings.

C. Benchmark Methods

We compare our approaches with several benchmark methods, including both conventional static recommender systems (Customer/Item Mean, NMF) and the state-of-the-art methods based on temporal behavior patterns (FPMC, PIMF). We summarize all the methods as follows, where the last two methods (NCTR and NCTRC) are proposed in this paper.

1) **Customer Mean**: This method makes prediction based on the mean value of each customer:

$$\widehat{R}_{ij}^n = \frac{\sum_{ij} R_{ij}^n I_{ij}^n}{\sum_{ij} I_{ij}^n}.$$

2) **Item Mean**: This method makes prediction based on the mean value of each item:

$$\widehat{R}_{ij}^n = \frac{\sum_{n} R_{ij}^n I_{ij}^n}{\sum_{n} I_{ij}^n}.$$

3) **NMF**: NMF is a widely used collaborative filtering approach [7], which factorizes the customer-campaign binary matrix (1 means the customer participated in the campaign, 0 otherwise).
4) **FPMC**: The Factorized Personalized Markov Chains model [19] transforms the sequential data of each user into a transition matrix and then predict the users next action by factorizing the matrices.
5) **PIMF**: The Purchase Interval based Matrix Factorization model [28] incorporates purchase interval into the marginal utility model to predict the most relevant items for recommendation at a given time. With this method, we compare the recommended list of campaigns with the future event sequence of each customer.
6) **NCTR**: This baseline method is our low-rank temporal graph reconstruction approach without any additional regularization.
7) **NCTRC**: This method is our low-rank temporal graph reconstruction approach with the community regularization for customers from the same company.

D. Performance Comparison and Discussion

Table II, III, and IV summarizes the comparison results on three B2B marketing data sets, respectively. For all methods, the parameters (if any) are tuned by the cross-validation procedure. More details on parameter selection in our method are provided in Section VI-F. From the results, we have the following observations:

1) In general, our methods (*NCTR* and *NCTRC*) consistently outperform other baseline methods such as *Customer Mean*, *Item Mean*, and *NMF* on all data sets. This observation well affirms our idea that the performance of NCTR recommendation can be improved by considering the temporal information in customer behaviors and the community structure between customers.
2) *NCTR* and *NCTRC* also outperform *FPMC* for predicting the next possible campaigns. The reason is that, the temporal graphs in our methods are more robust in representing the event sequences. For example, in our data, the events happen irregularly with varying time intervals. Meanwhile, there are missing

Metrics		CustomerMean	ItemMean	NMF	FPMC	PIMF	NCTR	NCTRC
Running Time (sec.)		**1.949**	8.160	31.265	336.218	6308.192	187.382	1241.641
$NDCG$	@3	0.6353	0.1442	0.7114	0.2971	0.7192	0.7538	**0.7597**
	@5	0.5890	0.1251	0.6697	0.2711	0.6281	0.6937	**0.7000**
	@10	0.5333	0.0777	0.6401	0.2603	0.5135	0.6549	**0.6562**
$Precision$	@3	0.1884	0.0104	0.2503	0.1408	0.1829	0.2543	**0.3037**
	@5	0.1339	0.0077	0.1872	0.1135	0.1256	0.1858	**0.2242**
	@10	0.0926	0.0118	0.1096	0.0744	0.0769	0.1117	**0.1324**
$Recall$	@3	0.2933	0.0091	0.3618	0.2871	0.3043	0.3672	**0.4179**
	@5	0.3308	0.0104	0.4267	0.3878	0.3928	0.4331	**0.4949**
	@10	0.4261	0.0326	0.4872	0.4992	0.4987	0.4973	**0.5506**

TABLE II: Performance Comparison on Data A.

Metrics		CustomerMean	ItemMean	NMF	FPMC	PIMF	NCTR	NCTRC
Running Time (sec.)		**0.344**	4.243	8.537	54.636	533.39	19.064	131.214
$NDCG$	@3	0.4701	0.3292	0.7114	0.3748	0.7336	0.7262	**0.7797**
	@5	0.4403	0.1752	0.6697	0.3338	0.6203	0.6740	**0.8053**
	@10	0.4000	0.1305	0.6302	0.3144	0.5323	0.6329	**0.8154**
$Precision$	@3	0.2165	0.0117	0.2832	0.1356	0.1708	0.2806	**0.3337**
	@5	0.1594	0.0237	0.2000	0.1330	0.1146	0.2031	**0.2462**
	@10	0.1086	0.0194	0.1211	0.0832	0.0654	0.1225	**0.1423**
$Recall$	@3	0.3246	0.0083	0.3916	0.2775	0.3124	0.3928	**0.4569**
	@5	0.3800	0.0296	0.4561	0.4467	0.4531	0.4685	**0.5349**
	@10	0.4811	0.0579	0.5314	0.5412	0.5549	0.5387	**0.5937**

TABLE III: Performance Comparison on Data B.

events not recorded in the behavior logs. All of these make the simple transition probabilities of *FPMC* less meaningful. In contrast, our temporal graphs directly compute the temporal correlations between marketing events for recommendations. The irregular time intervals are counted with robust smooth functions and the missing events cannot affect the construction of graph edges between observed events.

3) Moreover, *NCTR* and *NCTRC* outperform *PIMF* which considers the diminishing product's utility and user's satisfaction. In our data, the diminishing marginal utility pattern does not fit into the B2B marketing events. For example, a customer might participate in the same campaign (such as 'Webinar') several times consecutively. Our temporal graphs can model not only this kind of repeatability using self-connecting edges but also temporal correlations between campaigns in the evolving buying processes.

4) *NCTRC* achieves higher recommendation quality than *NCTR*, which demonstrates the benefits gained by incorporating the community information when computing the low-rank graph reconstruction. Note that, currently we define the communities for customers from the same company. In the future, we may identify more fine-grained community structures among customers, e.g., customers from the same company can form different groups for different buying tasks.

Customer	Last Record	Recommended List
	Company A	
C1	Search	Email Campaign, Web Advertising, Webinar, Seminar, Trial Download
C2	Webinar	Webinar, Seminar, Trade Show, Corporate Event, Conference
	Company B	
C3	Trade Show	Training, Outbound Telemarketing, Corporate Event, Trade Show, Seminar
C4	Trial Download	Trade Show, Training, Webinar, Seminar, Corporate Event
C5	Search	Webinar, Seminar, Trade Show, Trial Download, Training

TABLE V: Examples of Recommended Lists.

E. Case Study

To provide a better understanding, we demonstrate some detailed recommendation results in Table V for 5 customers from two companies. With different last behavior records, our approach recommends to each customer a list including distinct contents. For example, the last record of customer $C1$ is 'Search', which indicates that $C1$ is still in the primitive decision-making stage. Therefore, the recommended campaigns for $C1$ include first 'Email Campaign' and 'Web

Metrics		CustomerMean	ItemMean	NMF	FPMC	PIMF	NCTR	NCTRC
Running Time (sec.)		**0.936**	11.075	26.031	184.870	3098.882	96.312	522.005
$NDCG$	@3	0.1211	0.2887	0.1321	0.4174	0.7332	0.7238	**0.7380**
	@5	0.1695	0.2504	0.1794	0.3938	0.6621	0.6662	**0.7015**
	@10	0.1849	0.2271	0.1967	0.3697	0.5661	0.6385	**0.6721**
$Precision$	@3	0.0025	0.1446	0.1064	0.2082	0.1787	0.3077	**0.3304**
	@5	0.0202	0.1243	0.0025	0.1463	0.1226	0.2209	**0.2537**
	@10	0.0124	0.0943	0.0202	0.0902	0.0673	0.1393	**0.1463**
$Recall$	@3	0.0017	0.2158	0.0123	0.3939	0.3783	0.4204	**0.4547**
	@5	0.0315	0.3029	0.0017	0.4594	0.4658	0.4779	**0.5423**
	@10	0.0387	0.4307	0.0315	0.5652	0.5673	0.5762	**0.6131**

TABLE IV: Performance Comparison on Data C.

Advertising' to enhance the product awareness, and later 'Webinar', 'Seminar', and 'Trial Download' to boost the interest level of the customer. In comparison, with 'Trade Show' as the last record, $C3$ is currently in a more mature status toward purchase. Thus, this customer is provided with some late-stage marketing campaigns such as 'Training', 'Outbound Telemarketing', and 'Corporate Event', which can accelerate the buying process for final purchase decision.

Another interesting observation is that, although customer $C5$ has the same last record with $C1$, the customer $C5$ is provided with very different recommendations. This becomes natural with the consideration of the community relationships. Specifically, we investigated the historical records of other community members and found that the customers from this company are more proactive. They are already well aware of the products without those advertisement related campaigns. Therefore, some late-stage campaigns such as 'Trade Show' and 'Training' are recommended to $C5$.

These observations clearly show that, for recommending the next campaign to run, our approach can model not only the customer-specific behavior preferences but also the community-related behavior commonalities. Both these two perspectives are effective to improve the recommendation performances in terms of various measures.

F. Parameter Selection

Now we discuss the impacts of four parameters (Δ, r, K, λ) in our method. For the sake of simplicity, we only show the parameter tuning results for Data B.

First, the two parameters (Δ and r) for constructing the temporal graphs can be chosen according to the domain knowledge. For example, the thresholding parameter Δ in Equation 3 is set to be 90 days in our data sets. The reason is that the customer will make decisions hardly based on actions taken three months ago. Moreover, to make the numerical computing stable in the function $\exp(\cdot)$, the scaling parameter r for computing the temporal correlation in Equation 3 can be chosen as the average time interval between all the consecutive marketing events.

Second, we show how to decide the optimal number of graph basis K for reconstructing the temporal graphs with Equation 6. As shown in Figure 4, we plot the recommendation performances with increasing number of graph basis. As can be seen, the performances in terms of different measures vary significantly with the different numbers of graph bases. It is worthy to note that the performances might not increase with more bases. The reason is that, more bases imply higher modelling complexity and may lead to overfitting in the training data and decreasing generality of the identified graph bases. Based on Figure 4, we see that $K = 100$ is a feasible trade off between the modelling complexity and the empirical accuracy, with which we achieve the optimal performance consistently in terms of all measures. Therefore we choose $K = 100$ for this data set.

Finally, the last parameter λ, which controls the degree of the regularization using the community network in Equation 7, can also be chosen according to the recommendation performance. Intuitively, if we use a small value of λ, then we only employ the temporal graphs encoding the customer behavior preferences for making recommendations. On the other hand, if λ is larger, the community network information will have a stronger impact on the reconstruction of the temporal graphs. To choose the optimal λ, Figure 5 shows the recommendation performances with increasing λ, where $\lambda = 0.001$ gives the optimal results in terms of Precision, Recall and NDCG.

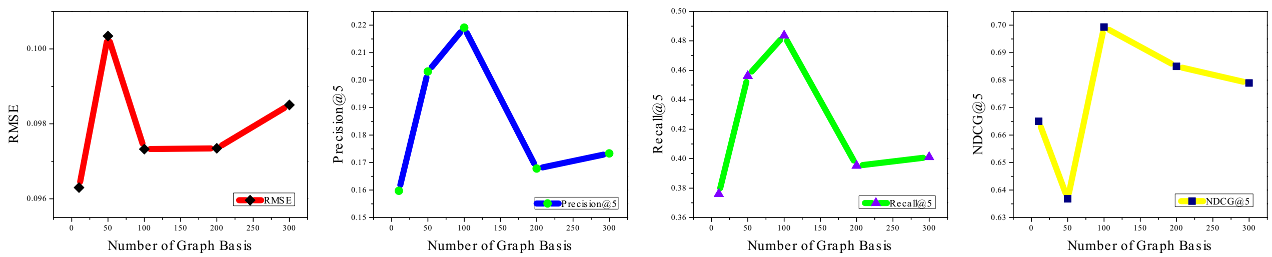
G. Time Complexity

As shown in Algorithm 1, during each iteration, we solve N subproblem in Equation 8 (with complexity $O(M^2K^2)$) and M^2 subproblem in Equation 9 (with complexity $O(NK^2)$). Therefore, the overall time complexity is $O(NM^2K^2T)$ where T is the number of iterations. At the top of Table II, III, and IV, we report the running time of all methods. The results show that in terms of the running time:

$$\begin{aligned} &Customer/Item\ Mean < NMF < NCTR \\ &< FPMC < NCTRC < PIMF. \end{aligned}$$

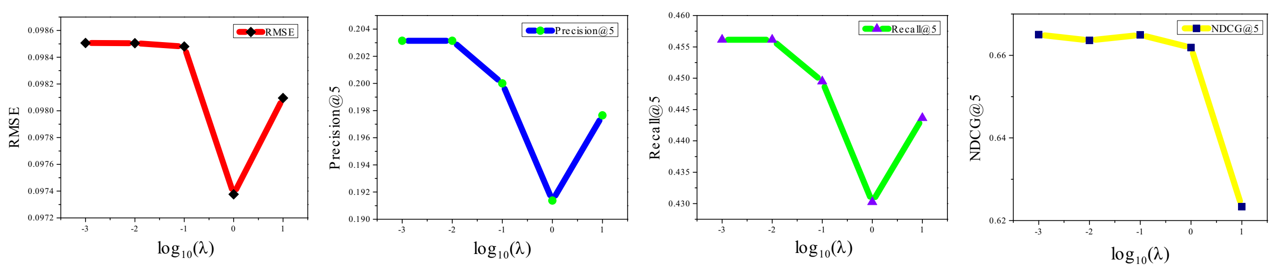
VII. Related Work

This section reviews several categories of the existing work that are closely related to the research proposed in this paper.



(a) RMSE (b) Precision@5 (c) Recall@5 (d) NDCG@5

Fig. 4: Impacts of The Number of Graph Basis on Model Performances.



(a) RMSE (b) Precision@5 (c) Recall@5 (d) NDCG@5

Fig. 5: Impacts of λ on Model Performances.

A. Traditional Recommender Systems

The research of recommender systems have been an active topic in recent years. In general, there are three major recommendation approaches including content-based methods, collaborative filtering (CF), and hybrid methods [2, 1]. One of the most widely used approaches is the collaborative filtering, which provides recommendations by predicting what users will like based on their similarity to other users. One way to define the similarity is to compute statistics in the user-item rating data. Another way is to define the similarity implicitly by fitting the observed data with latent models which can be used to make unobserved predictions. The model-based approaches received a great attention especially in the Netflix movie recommendation competition [9], where the low-rank matrix factorization was shown effective and efficient with sparse observations [18, 17, 24].

B. Temporal Recommender Systems

Another type of useful information in the design of recommender systems is the temporal pattern, which is still underexplored yet. For example, Koren [8], Lathia et al. [10], Xiong et al. [24], Chen et al. [4] showed that the temporal dynamics can be considered in the collaborative filtering model to learn the dynamic characteristics of the users and items. Also, Tang and Zhou [21] proposed to extract dynamic features using time-series analysis and apply the adaptively weighting algorithm to make recommendations. Liu et al. [12] proposed to combine explicit and implicit user feedbacks to learn the seasonality or short-term preferences for movie recommendations. Xiang et al. [23] proposed to combine the user similarity as the long-term preference and the product similarity as the short-term bias to make recommendations.

Another research direction of the temporal recommender system is to exploit the time factors for short-term 'next-item' recommendations. To this end, Rendle et al. [19] integrates the latent factor model and Markov chain model for next-basket recommendation. Wang and Zhang [22] proposed an opportunity model to estimate the follow-up purchase probability of a user at a specific time. Yap et al. [26] proposed to learn user-specific 'sequence important knowledge' through personalized sequential pattern mining. Recently, to improve the next-product recommendations in e-commerce, the time interval between purchase behaviors has been modeled by [28, 29]. Similar ideas have also been used in other applications [3], such as the recommendation of the next-POI (Point-of-Interest) to check-in in location-based services [5, 27]. We implemented the closely related and applicable methods in these researches and discussed more details in Section VI.

C. Recommender Systems with Social Information

Finally, the social information has been used to improve the recommendation performances. In particular, Ma et al. [14, 15, 16] proposed to integrate the social network with the matrix factorization method to learn the latent factors for both users and items with different applications. To improve review quality prediction, Lu et al. [13] developed a generic framework for incorporating social context information by adding regularization constraints in the text-based predictor. To better utilize user's social trust information, Yang et al. [25] developed a category-specific social trust circle based model with the user-item rating data combined with social network data. Moreover, in [6], the social correlation is used with Latent Dirichlet Allocation (LDA) to model the users' adoption of items. The authors further devised a hybrid model that combines a user's own latent factors with her friends' for adoption prediction. In [20], a joint personal and social latent factor (PSLF) model is used for social recommendation by explicitly expressing the varieties of the social relationships for each user. In this paper, we have a simple community structure of the business customers. We observed that the individuals in the same community collaborate with each other and thus we incorporate a community regularization to reflect the collaborations in our low-rank graph reconstruction approach to improve the marketing campaign recommendations.

VIII. Conclusion

In this paper, we developed a novel recommender system to combine the temporal and social factors captured in customer behavior records and the customer community networks for B2B marketing campaign recommendations. The goal is to provide the marketers with a better marketing strategy to expedite the customer conversion cycle and boost the customer conversion ratio. Specifically, we first represented the rich temporal content in customer behavior records using the temporal graphs. Next, with the personalized temporal graphs, we computed the low-rank graph reconstruction to predict the unobserved graph edges. Moreover, we regularized the graph reconstruction with the community network of the business customers. Finally, we developed efficient algorithms to compute the optimal solutions, which we have applied on several real-world B2B marketing data sets. The experiments clearly validated the effectiveness of the proposed approach in comparison with the state-of-the-art methods.

IX. Acknowledgments

This research was partially supported by the Rutgers 2015 Chancellor's Seed Grant. Also, it was supported in part by Natural Science Foundation of China (71028002).

References

[1] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *TKDE*, 17(6):734–749, 2005.

[2] John S Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pages 43–52. Morgan Kaufmann Publishers Inc., 1998.

[3] Pedro G Campos, Fernando Díez, and Iván Cantador. Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. *User Modeling and User-Adapted Interaction*, 24(1-2):67–119, 2014.

[4] Wei Chen, Wynne Hsu, and Mong Li Lee. Modeling user's receptiveness over time for recommendation. In *SIGIR*, pages 373–382. ACM, 2013.

[5] Chen Cheng, Haiqin Yang, Michael R Lyu, and Irwin King. Where you like to go next: Successive point-of-interest recommendation. In *Proceedings of the 23rd international joint conference on Artificial Intelligence*, pages 2605–2611. AAAI Press, 2013.

[6] Freddy Chong Tat Chua, Hady Wirawan Lauw, and Ee-Peng Lim. Predicting item adoption using social correlation. In *SDM*, volume 11, pages 367–378. SIAM, 2011.

[7] Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *The Journal of Machine Learning Research*, 5:1457–1469, 2004.

[8] Yehuda Koren. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89–97, 2010.

[9] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8): 30–37, 2009.

[10] Neal Lathia, Stephen Hailes, Licia Capra, and Xavier Amatriain. Temporal diversity in recommender systems. In *SIGIR*, pages 210–217. ACM, 2010.

[11] Chuanren Liu, Kai Zhang, Hui Xiong, Geoff Jiang, and Qiang Yang. Temporal skeletonization on sequential data: Patterns, categorization, and visualization. *TKDE*, 2015.

[12] Nathan N Liu, Luheng He, and Min Zhao. Social temporal collaborative ranking for context aware movie recommendation. *TIST*, 4(1):15, 2013.

[13] Yue Lu, Panayiotis Tsaparas, Alexandros Ntoulas, and Livia Polanyi. Exploiting social context for review quality prediction. In *WWW*, pages 691–700. ACM, 2010.

[14] Hao Ma, Irwin King, and Michael R Lyu. Learning to recommend with social trust ensemble. In *SIGIR*, pages 203–210. ACM, 2009.

[15] Hao Ma, Michael R Lyu, and Irwin King. Learning to recommend with trust and distrust relationships. In *Proceedings of the third ACM conference on Recommender systems*, pages 189–196. ACM, 2009.

[16] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. Recommender systems with social regularization. In *WSDM*, pages 287–296. ACM, 2011.

[17] Andriy Mnih and Ruslan Salakhutdinov. Probabilistic matrix factorization. In *Advances in neural information processing systems*, pages 1257–1264, 2007.

[18] Arkadiusz Paterek. Improving regularized singular value decomposition for collaborative filtering. In *Proceedings of KDD cup and workshop*, volume 2007, pages 5–8, 2007.

[19] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *WWW*, pages 811–820. ACM, 2010.

[20] Yelong Shen and Ruoming Jin. Learning personal+ social latent factor model for social recommendation. In *SIGKDD*, pages 1303–1311. ACM, 2012.

[21] Xiangyu Tang and Jie Zhou. Dynamic personalized recommendation on sparse data. *TKDE*, 25(12):2895–2899, 2013.

[22] Jian Wang and Yi Zhang. Opportunity model for e-commerce recommendation: right product; right time. In *SIGIR*, pages 303–312. ACM, 2013.

[23] Liang Xiang, Quan Yuan, Shiwan Zhao, Li Chen, Xiatian Zhang, Qing Yang, and Jimeng Sun. Temporal recommendation on graphs via long-and short-term preference fusion. In *SIGKDD*, pages 723–732. ACM, 2010.

[24] Liang Xiong, Xi Chen, Tzu-Kuo Huang, Jeff G Schneider, and Jaime G Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *SDM*, volume 10, pages 211–222. SIAM, 2010.

[25] Xiwang Yang, Harald Steck, and Yong Liu. Circle-based recommendation in online social networks. In *SIGKDD*, pages 1267–1275. ACM, 2012.

[26] Ghim-Eng Yap, Xiao-Li Li, and S Yu Philip. Effective next-items recommendation via personalized sequential pattern mining. In *Database Systems for Advanced Applications*, pages 48–64. Springer, 2012.

[27] Jia-Dong Zhang, Chi-Yin Chow, and Yanhua Li. Lore: Exploiting sequential influence for location recommendations. In *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Dallas, TX*, 2014.

[28] Gang Zhao, Mong Li Lee, Wynne Hsu, and Wei Chen. Increasing temporal diversity with purchase intervals. In *SIGIR*, pages 165–174. ACM, 2012.

[29] Gang Zhao, Mong LI Lee, and Hsu Wynne. Utilizing purchase intervals in latent clusters for product recommendation. In *SNAKDD*, page 4. ACM, 2014.