A Dynamic Recurrent Model for Next Basket Recommendation

Feng Yu, Qiang Liu*, Shu Wu, Liang Wang, Tieniu Tan
Center for Research on Intelligent Perception and Computing
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences
{feng.yu, qiang.liu, shu.wu, wangliang, tnt}@nlpr.ia.ac.cn

ABSTRACT

Next basket recommendation becomes an increasing concern. Most conventional models explore either sequential transaction features or general interests of users. Further, some works treat users' general interests and sequential behaviors as two totally divided matters, and then combine them in some way for next basket recommendation. Moreover, the state-of-the-art models are based on the assumption of Markov Chains (MC), which only capture local sequential features between two adjacent baskets. In this work, we propose a novel model, Dynamic REcurrent bAsket Model (DREAM), based on Recurrent Neural Network (RNN). DREAM not only learns a dynamic representation of a user but also captures global sequential features among baskets. The dynamic representation of a specific user can reveal user's dynamic interests at different time, and the global sequential features reflect interactions of all baskets of the user over time. Experiment results on two public datasets indicate that DREAM is more effective than the state-of-the-art models for next basket recommendation.

Keywords

Next basket recommendation; recurrent neural network

1. INTRODUCTION

In real-world scenarios, a customer always purchases a series of baskets of items at different time. This recommendation task in e-commerce sites is formulated as the next basket recommendation, which has received much attention recently [1, 3].

In general, there are two distinct approaches for next basket recommendation. One perspective is the collaborative filtering (CF) models, which capture users' general interests but have difficulty in considering sequential features of historical transactions. Matrix factorization (MF) is a successful CF model [4]. Through factorizing a user-item matrix constructed by the whole historical transaction data, users' general interests can be represented by latent vectors. For instance, a sportsman always buy various athletic equipment, the latent vector may tell that he is interested in sport and we can recommend sports items. Moreover, some sequential recommendation models mainly based on MC [2], which extract sequential features from historical transactions and then predict next purchase based on these sequential behaviors.

As a consequence, a more appropriate way to next basket recommendation is to capture above sequential behaviors and user general interests in a hybrid model. Factorizing Personalized Markov Chains (FPMC) can model sequential behaviors between every two adjacent baskets, and user general interests is shaped by items in baskets [7]. Actually there are multiple interacting factors influencing users' next purchase in real commercial scenarios. FPMC just utilizes a linear operation on multiple factors, and cannot depict the interactions among multiple factors. Hierarchical Representation Model (HRM) seems to partially solve the problem of how to summarize multiple interacting factors through a nonlinear *max pooling* operation [9]. Nevertheless, all the MC based methods (including above FPMC and HRM) have the same deficiency that these recommenders can only model local sequential behaviors between every two adjacent baskets, and some of which may be irrelevant sometimes. For instance, a user u bought a ultrabook in basket $B^u_{t_1}$ (a basket of user u at time t_1, similarly hereinafter), some food in $B^u_{t_2}$ and accessories of the ultrabook in $B^u_{t_3}$, there does not exist any relevance between every two adjacent baskets. Hence, we need to model global sequential behaviors to make the best of all relations among sequential baskets like the above $B^u_{t_1}$ and $B^u_{t_3}$. For this reason, we plan to model global sequential features among all sequential baskets of a user.

In order to mine global sequential features in complex commercial scenarios and reveal dynamic representations of users' interests, deep neural network is employed in this work. As stated above, local sequential features extracted by HRM is not capable enough to model relations among apart baskets, while a recurrent operation of a deep RNN architecture can capture global sequential features from all baskets of a user. Recently, RNN approaches to word embedding for sentence modeling [5], sequential click prediction [10] have achieved much success in respective fields. We propose a dynamic recurrent model, i.e., DREAM, for next bas-

*The first two authors contributed equally to this paper.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGIR '16, July 17–21, 2016, Pisa, Italy
© 2016 ACM. ISBN 978-1-4503-4069-4/16/07...$15.00
DOI: http://dx.doi.org/10.1145/2911451.2914683

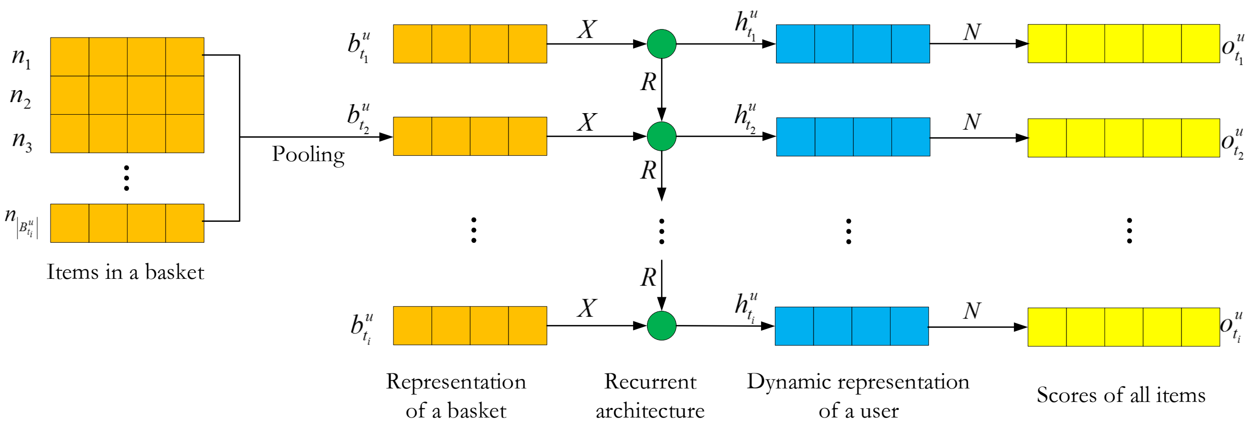


Figure 1: The framework of DREAM. Pooling operation on the items in a basket to get the representation of the basket. The input layer comprises a series of basket representations of a user. Dynamic representation of the user can be obtained in the hidden layer. Finally the output layer shows scores of this user towards all items.

ket recommendation. An input instance of DREAM model consists a series of baskets of items, which are sequential transactions of a specific user. Pooling and matrix operations can offer each user a dynamic representation with different baskets over time. Moreover, the recurrent structure can obtain some global sequential features for all users from overall historical transaction data. Our experiment results on two real-world datasets reveal that the DREAM model achieves great improvement for next basket recommendation comparing with the state-of-the-art models such as FPMC, HRM.

In this work, we take advantage of the whole historical sequential transaction data to gain comprehensive understanding of users' purchase interests and consequently recommend items that each user most probably purchase in the next visit. The main contributions of this work are as follows. We investigate the dynamic representation of each user and the global sequential behaviors of item-purchase history. Experiments on two datasets are conducted to validate the effectiveness of DREAM model. To the best of our knowledge, DREAM is the first approach that attempts to incorporate dynamic representation and global sequential behaviors for enhancing the performance of next basket recommendation.

2. THE PROPOSED APPROACH

In this section, we formulate the task of next basket recommendation and then introduce the proposed DREAM model in detail.

2.1 Problem Formulation

In the scenario of next basket recommendation, there are a mass of users, and each user purchases a series of baskets of items. Let $\boldsymbol{N}$ be the representations of items, and $\boldsymbol{n}_v \in \mathbb{R}^d$ indicates the latent representations of item v. For a user u, the historical transactions B^u are composed of a collection of baskets $\{B^u_{t_1}, B^u_{t_2}, ...\}$ in time order, where $B^u_{t_i}$ is a basket of items purchased by user u at time t_i. For next basket recommendation with historical transaction data, we formalize the problem as predicting a ranking list of items for each user at a specific time t_i.

2.2 DREAM

The general framework of DREAM is illustrated in Figure 1. An input instance of the proposed model are a sequence of baskets. For one basket $B^u_{t_i}$ of the user u, there are a variety of items, $B^u_{t_i} = \left\{\boldsymbol{n}^u_{t_i,j} \in \mathbb{R}^d | j = 1, 2, ..., \left|B^u_{t_i}\right|\right\}$. $\boldsymbol{n}^u_{t_i,j}$ is the latent representation of the j-th item in basket $B^u_{t_i}$ and $\left|B^u_{t_i}\right|$ means the number of items in basket $B^u_{t_i}$. Now, we can generate the latent vector representation $\boldsymbol{b}^u_{t_i}$ for a basket $B^u_{t_i}$ by aggregating representation vectors of these items. In this work, we adopt two kinds of aggregation operation, i.e., *max pooling* and *average pooling*.

For the *max pooling* operation, we aggregate a group of vectors through taking the maximum value of every dimension among all those vectors. Then each dimension of $\boldsymbol{b}^u_{t_i}$ is formulated as

$$b^u_{t_i,k} = max\left(n^u_{t_i,1,k}, n^u_{t_i,2,k}, ...\right), \qquad (1)$$

where $b^u_{t_i,k}$ is the k-th dimension of a basket-representing vector $\boldsymbol{b}^u_{t_i}$, $n^u_{t_i,j,k}$ means the value of k-th dimension of the vector representation of the j-th item $(\boldsymbol{n}^u_{t_i,j})$ in basket $B^u_{t_i}$.

The *average pooling* is a similar operation but replaces maximum with average. In other words, the *average pooling* is to aggregate a group of vectors through taking the average value of every dimension of all those vectors, which can be formulated in a similar way as

$$\boldsymbol{b}^u_{t_i} = \frac{1}{\left|B^u_{t_i}\right|}\sum\nolimits_{j=1}^{\left|B^u_{t_i}\right|} \boldsymbol{n}^u_{t_i,j}. \qquad (2)$$

These above representations of baskets can form the input layer of a recurrent architecture.

As is shown in Figure 1, the vector representation of a hidden layer $\boldsymbol{h}^u_{t_i}$ is the dynamic representation of user u at time t_i. The recurrent connection weight matrix $\boldsymbol{R}$ helps to propagate sequential signals between every two adjacent hidden state $\boldsymbol{h}^u_{t_{i-1}}$ and $\boldsymbol{h}^u_{t_i}$[10]. $\boldsymbol{X}$ is a transition matrix between latent vector representations of baskets and a user's interests. Then, the vector representation of the hidden layer can be computed as:

$$\boldsymbol{h}^u_{t_i} = f\left(\boldsymbol{X}\boldsymbol{b}^u_{t_i} + \boldsymbol{R}\boldsymbol{h}^u_{t_{i-1}}\right), \qquad (3)$$

where $\boldsymbol{b}^u_{t_i}$ is a latent vector representation of the user's basket at time t_i, and $\boldsymbol{h}^u_{t_{i-1}}$ is the dynamic representation of the previous time t_{i-1}. $f(x)$ is a activation function, here we choose a *sigmoid* function $f(x) = \frac{1}{1+e^{-x}}$. Finally the model can output a user's scores $\boldsymbol{o}_{u,t_i}$ towards all items at time t_i. The output $\boldsymbol{o}_{u,t_i}$ can be calculated through multiplication of item matrix $\boldsymbol{N}$ and a user's dynamic representation $\boldsymbol{h}^u_{t_i}$, which is formulated as follows:

$$\boldsymbol{o}_{u,t_i} = \boldsymbol{N}^T \boldsymbol{h}^u_{t_i}. \tag{4}$$

Therefore $o_{u,t_i,v}$, i.e., an element of $\boldsymbol{o}_{u,t_i}$, represents the score of a transaction between a user u and an item v at time t_i. A higher score indicates that the user is more likely to purchase the corresponding item.

2.3 Objective Function

In the learning process of DREAM, we adopt Bayesian Personalized Ranking (BPR) [6]. BPR is a state-of-the-art pairwise ranking framework for the implicit feedback data. The basic assumption is that a user prefers an item in basket at a specific time than a negative item sample. The negative items can be any other items apart from those in the basket. In this way, we need to maximize the following probability:

$$p\left(u, t, v \succ v'\right) = \sigma\left(o_{u,t,v} - o_{u,t,v'}\right), \tag{5}$$

where v' denotes a negative item sample, and $\sigma(x)$ is a nonlinear function which is chosen as $\sigma(x) = \frac{1}{1+e^{-x}}$. Adding up all the log likelihood and the regularization term, the objective function can be written as follows:

$$J = \sum \ln\left(1 + e^{-\left(o_{u,t,v} - o_{u,t,v'}\right)}\right) + \frac{\lambda}{2}\|\Theta\|^2, \tag{6}$$

where $\Theta = \{\boldsymbol{N}, \boldsymbol{R}, \boldsymbol{X}\}$ denotes all the parameters to be learnt, λ is a parameter to control the power of regularization. Furthermore, the objective function can be optimized by Back Propagation Through Time (BPTT) [8]. BPTT is to iteratively repeat the calculation of derivations of J with respect to different parameters and obtain these gradients of all the parameters in the end. Then we update parameters utilizing Stochastic Gradient Descent (SGD) until converge.

Notice that the DREAM model utilize an iterative method in learning users' representation vectors. That is to say, for any new transactions, we can update users' representation vectors based current ones. Some state-of-the-art models, such as HRM, need to factorize a new built user-item matrix to get users' representation vectors. Therefore this iterative learning method may be more practical in real-world applications.

3. EXPERIMENTS

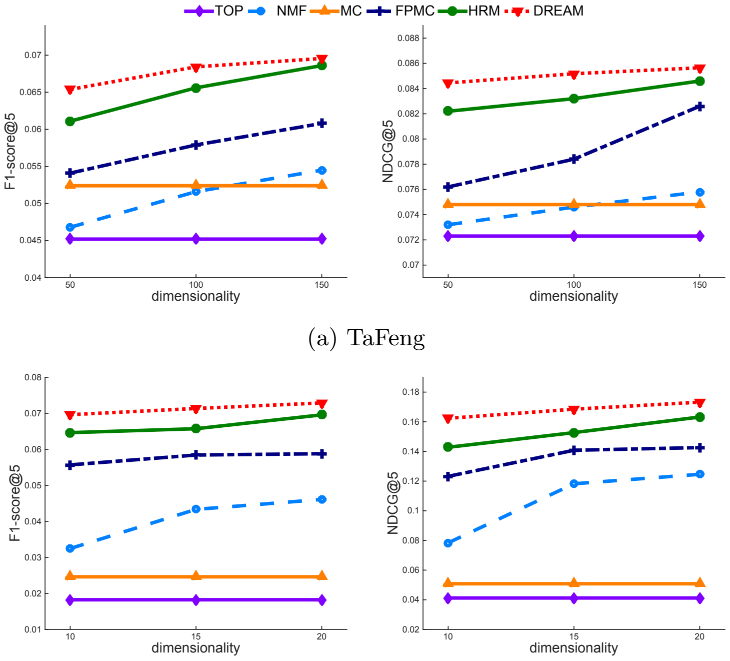
3.1 Datasets and Baselines

To evaluate the performance of our method on the task of next basket recommendation, we perform experiments on two real-world datasets, i.e., Ta-Feng[1] and T-mall[2]. The Ta-Feng dataset contains numerous baskets of purchased items from a grocery store, where each basket encapsulates the items purchased by one user in a period of time. This dataset is a public dataset which contains 817,741 transactions belonging to 32,266 users and 23,812 items. The T-mall dataset is a public online e-commerce dataset released by Alibaba group[3], which contains 4,298 transactions of 884 users and 9,531 brands. The slight difference between these two datasets is that the T-mall dataset records the transactions based on brands and each brand may covers a series of items. The above datasets are preprocessed to obtain k-*core* subsets [7], i.e. each user u purchased in total at least k items $\left(\sum_{t_i} \left|B^u_{t_i}\right| \geqslant k\right)$ and vice versa each item was purchased by at least k users. We set $k = 10$ for the Ta-Feng dataset, and $k = 3$ for the relatively smaller T-Mall dataset.

Several baseline and state-of-the-art methods on next-basket recommendation are used for empirical comparison. (1) **TOP** recommends the top popular items to each user. (2) **MC** is a Markov chain model based on sequential transaction information of a user. The prediction function is as follows:

$$p\left(i \in B^u_{t_i} | B^u_{t_{i-1}}\right) := \frac{1}{\left|B^u_{t_{i-1}}\right|} \sum_{j \in B^u_{t_{i-1}}} p\left(i \in B^u_{t_i} | j \in B^u_{t_{i-1}}\right)$$

(3) **NMF** is a collaborative filtering method, which applies Nonnegative Matrix Factorization over the user-item matrix. For implementation, we adopt the released codes from NMF: DTU Toolbox[4]. (4) **FPMC** [7] is a hybrid model combining MC and MF for next basket recommendation, which can capture both sequential effects and general interests of users. (5) **HRM** [9] is a state-of-the-art hierarchical representation model, which can capture general users' interests and sequential effects. Besides, with various nonlinear operations, HRM can capture all those factors more properly than previous models.



(a) TaFeng

(b) Tmall

Figure 2: Experiment Results of different methods on two datasets.

[1] http://recsyswiki.com/wiki/Grocery shopping datasets

[2] http://102.alibaba.com/competition/addDiscovery/index.htm

[3] http://www.alibabagroup.com/cn/global/home

[4] http://cogsys.imm.dtu.dk/toolbox/nmf/

Table 1: Performance comparison of two pooling operations on two datasets

Methods	Ta-Feng						Tmall					
	d=50		d=100		d=150		d=10		d=15		d=20	
	f1-score	NDCG	f1-score	NDCG	f1-score	NDCG	f1-score	NDCG	f1-score	NDCG	f1-score	NDCG
Avg-Pooling	0.061	0.082	0.064	0.081	0.067	0.083	0.058	0.141	0.063	0.154	0.066	0.160
Max-Pooling	**0.065**	**0.084**	**0.068**	**0.085**	**0.070**	**0.086**	**0.070**	**0.162**	**0.071**	**0.168**	**0.073**	**0.173**

3.2 Metrics and Setup

For recommendation, we generate a ranking list of K items ($K = 5$) for each user u. In order to measure the performance of next basket recommendation, we adopt two evaluation metrics, i.e., $F1$-*score* and Normalized Discounted Cumulative Gain ($NDCG$). $F1$-*score* calculates the harmonic mean of the precision and recall measurements. $NDCG$ is a cumulative measure of ranking quality, which is more sensitive to the relevance of higher ranked items. For both metrics, the larger the value, the better the performance.

On both datasets, we use the last transaction of each user as the testing data and all the rest transactions as the training data. The vector representations of items are randomly initialized. Moreover, performance results of different methods are compared along with varying dimensions d of the representation. We illustrate the results with dimensions $\{50, 100, 150\}$ for the Ta-Feng dataset, and $\{10, 15, 20\}$ for the relatively smaller T-Mall dataset.

3.3 Results and Analyses

First, the performance of DREAM model are compared with the state-of-the-art methods. As illustrated in Figure 2, in general, the performance ranking of next basket recommendation methods is as follows, DREAM, HRM, FPMC, NMF, MC and TOP. Since the baseline TOP just list the popular items and does not utilize the features of separate baskets, this method is the weakest one among all methods. Despite the fact that NMF and MC leverage only one kind of feature, either sequential behaviors or users' general interests, we can observe that the NMF model achieve better performance than that of the MC model, especially on the sparse T-mall data. It may be because that MC cannot reveal the collaborative information among users. On the sparse user-item matrix of T-mall, collaborative information is more important to generate the accurate interests of users than the sparse sequential behaviors. On both datasets, the HRM model outperforms the FPMC model. Though FPMC and HRM both utilize sequential behaviors, the nonlinear operations among multiple factors of HRM earn it a better performance, while the FPMC model's linear independence assumption of interaction relationship of items in a basket makes it inapplicable in complex commercial scenarios. The proposed DREAM model can consistently outperform all comparing models in terms of both metrics on two datasets. These results show that the dynamic representation of user with a recurrent architecture is effective in capturing sequential features and dynamic interests of users. Besides, richer nonlinear operations such as *pooling* and activation functions contribute to a better representations of baskets.

Then, we assess performances of the DREAM model with *max pooling* and *average pooling*. As illustrated in Table 1, DREAM with *max pooling* can outperform DREAM with *average pooling* on both datasets with $F1$-*score*@5 and $NDCG$@5. It demonstrates that *max pooling* gains advantage over *average pooling* in modeling interactions among multiple factors. Obviously, as a linear operation, *average pooling* takes an average representation of a basket, indicating that each item in a basket measures the basket representation in an independent way. In real-world scenario, many items we purchase are interactive, that is to say, one item influences whether we purchase another item, then the whole items we purchase can help shape our interests. Consequently a better solution is to learn the elaborate interaction relationship of a basket of items through a nonlinear operation. *Max pooling* is a nonlinear operation, which takes a key representation of a basket and is more capable to learn those complicated interactions than a linear operation.

4. CONCLUSIONS

In this paper, we have proposed a dynamic recurrent basket model based on RNN for next basket recommendation. Our model can merge users' current interests and global sequential features into users' recurrent and dynamic representation. Moreover, it shows that the nonlinear operation on learning the representation of a basket does well in capturing elaborate interactions among multiple factors of items. Extensive experiments on two public datasets demonstrated the effectiveness of the proposed model.

5. ACKNOWLEDGMENTS

This work is jointly supported by National Basic Research Program of China (2012CB316300), and National Natural Science Foundation of China (61403390, U1435221, 61525306).

6. REFERENCES

[1] G. Adomavicius and A. 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] S. Chen, J. L. Moore, D. Turnbull, and T. Joachims. Playlist prediction via metric embedding. In *SIGKDD*, pages 714–722, 2012.

[3] A. Gatzioura and M. Sanchez-Marre. A case-based recommendation approach for market basket data. *IEEE Intelligent Systems*, 30(1):20–27, 2015.

[4] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, (8):30–37, 2009.

[5] T. Mikolov, S. Kombrink, L. Burget, J. H. Černockỳ, and S. Khudanpur. Extensions of recurrent neural network language model. In *ICASSP*, pages 5528–5531, 2011.

[6] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *UAI*, pages 452–461, 2009.

[7] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *WWW*, pages 811–820, 2010.

[8] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5:3, 1988.

[9] P. Wang, J. Guo, Y. Lan, J. Xu, S. Wan, and X. Cheng. Learning hierarchical representation model for nextbasket recommendation. In *SIGIR*, pages 403–412, 2015.

[10] Y. Zhang, H. Dai, C. Xu, J. Feng, T. Wang, J. Bian, B. Wang, and T.-Y. Liu. Sequential click prediction for sponsored search with recurrent neural networks. In *AAAI*, pages 1369–1376, 2014.