IEICE TRANS. INF. & SYST., VOL.E100–D, NO.9 SEPTEMBER 2017

LETTER

Enhancing Purchase Behavior Prediction with Temporally Popular Items

Chen CHEN[†], ***Member***, **Chunyan HOU**[††a)], **Jiakun XIAO**[†], **Yanlong WEN**[†], ***and*** **Xiaojie YUAN**[†], ***Nonmembers***

SUMMARY In the era of e-commerce, purchase behavior prediction is one of the most important issues to promote both online companies' sales and the consumers' experience. The previous researches usually use traditional features based on the statistics and temporal dynamics of items. Those features lead to the loss of detailed items' information. In this study, we propose a novel kind of features based on temporally popular items to improve the prediction. Experiments on the real-world dataset have demonstrated the effectiveness and the efficiency of our proposed method. Features based on temporally popular items are compared with traditional features which are associated with statistics, temporal dynamics and collaborative filter of items. We find that temporally popular items are an effective and irreplaceable supplement of traditional features. Our study shed light on the effectiveness of the combination of popularity and temporal dynamics of items which can widely used for a variety of recommendations in e-commerce sites.

key words: *recommender system, behavior analysis, temporal dynamics, session*

1. Introduction

As the e-commerce becomes popular and integrates into the daily life, customers spend more time to purchase online, and the analysis of their purchase behavior has become an increasingly important business tool for promoting sales. With the rise of e-commerce companies like Amazon in USA and Taobao in China, they are looking for all possible approaches to detect customers' intent and predict their purchase behaviors. Both the consumers and e-commerce companies benefit from the prediction technique. However, the purchase prediction becomes difficult when there is only the http session which consists of a list of clicks in a short time. The prediction with the short-term history becomes an area of growing research and commercial interest in recent years. Especially, RecSys Challenge 2015 [1] is associated with such problem. A history of user's click behavior during a browsing session at a website of online retailer is given, and the goal is to predict whether a user will purchase at the end of this session.

Session-based recommendation has been researched in the music recommendation [4], [8], [14]. Although there have been some studies in developing recommendation techniques based on the session, few work has been done to research purchase behavior prediction based on e-commerce sessions. Koren [7] studied the collaborative filtering with temporal dynamics and proposed TimeSVD++ model. Our method differs from his work because the temporal dynamics of user preference in TimeSVD++ model are based on users, rather than sessions. In addition, temporal dynamics of all items for a user in TimeSVD++ is taken into account while we only consider a part of items (i.e., temporally popular items) in a session.

With respect to purchase behavior prediction, previous studies are based on the demographic information [6], social media profiles [13], or media advertisements [3], [10]. Compared with those researches, our work is based on a list of clicks during a browsing session at an e-commerce website. The most related works come from researchers [9], [11], [12] who participate RecSys 2015 Challenge. The challenge is based on large number of sessions from an online e-commerce retailer [1]. Our study is inspired by their idea. Especially, we propose a novel kind of features based on temporal popular items, which are the effective supplement of traditional features and can significantly improve the prediction.

Our study reveals that it is more appropriate to include temporally popular items than all items. If temporally popular items are used, both the effectiveness and efficiency can be enhanced for purchase behavior prediction. In addition, we compare the effectiveness of four groups of features to understand their relations by the add-one-in and leave-one-out experiments. The results reveal that temporally popular items are the most effective and irreplaceable features.

The rest of the paper is organized as follows. Section 2 introduces temporally popular items. We give the problem definition and machine learning algorithms in Sect. 3. Experiment setting and results are discussed in Sect. 4. In Sect. 5, we conclude our paper.

2. Temporally Popular Items

We give two important definitions, and then introduce features based on temporally popular items.

Definition 1 (Temporal Popularity): Given an item i and a period of time t, the temporal popularity of item i in t is denoted by $TP(i, t)$ which is the measurement of users' behaviors on the item i in t.

In this paper, we regard the number of sessions which

Manuscript received March 7, 2017.
Manuscript revised May 7, 2017.
Manuscript publicized May 30, 2017.

[†] The authors are with the College of Computer and Control Engineering, Nankai University, Tianjin, China.

[††] The author is with the School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin, China.

a) E-mail: chunyanhou@163.com (Corresponding author)

DOI: 10.1587/transinf.2017EDL8057

consist of clicks on the item i in t as $TP(i,t)$. For example, if there are 500 sessions which include clicks on the item i on April 1st, 2015, $TP(i,t) = 500$ (i.e., t is the April 1st, 2015.).

Definition 2 (Temporally Popular Items): Given a period of time t, the temporally popular items is denoted by $TPI(t,k)$ which is a set of items with the top-k temporal popularity in t.

For example, if the $TP(i,t)$ of item i is the largest and item j the second largest among items clicked in t, $TPI(t,2) = \{i, j\}$. One day is used as the duration of t.

Given the day t, the training dataset is used to compute the temporal popularity $TP(i,t)$ of each item i and then we can determine $TPI(t,k)$. Suppose there are N unique items in the training dataset, and we select $TPI(t,k)$ in t. For a session s, $t(s)$ denotes the time when the session s occurs, $p(s)$ means the set of clicked items in the session s, $v(s)$ is a $1 \times N$ vector and the m-th element of $v(s)$ is defined as

$$v_m(s) = \begin{cases} 1 & I_m \in TPI(t(s),k) \cap p(s) \\ 0 & o.w. \end{cases} \tag{1}$$

where I_m denotes the item which corresponds to the m-th element of $v(s)$.

By the Eq. (1), we can create a $1 \times N$ feature vector for each session, which is called TPI features. The TPI features enable the machine learning algorithms to leverage the popularity and temporal dynamics of items.

3. Purchase Behavior Prediction

3.1 Problem Definition

The dataset includes train and test dataset. Train and test dataset consist of sets of sessions S_{train} and S_{test} respectively. Each session s is represented as a click stream

$$c(s) = (c_1(s), c_2(s), \cdots, c_{n(s)}(s)) \tag{2}$$

$$c_j(s) = (i_j(s), t_j(s), y_j(s)), j \in \{1, \cdots, n(s)\} \tag{3}$$

where $n(s)$ is the number of clicks in session s, $i_j(s)$ denotes the j-th clicked item in session s, $t_j(s)$ is the time when the item $i_j(s)$ is clicked, $y_j(s)$ is one when item $i_j(s)$ is purchased at least once, and zero otherwise. A session s has the label $y(s)$ defined as

$$y(s) = \begin{cases} 1 & \exists j : y_j(s) = 1 \\ 0 & \forall j : y_j(s) = 0 \end{cases} \tag{4}$$

If $y(s) = 1$, session s is a buy session. We are given sets of purchased items for session $s \in S_{train}$, and are required to predict $y(s)$ for session $s \in S_{test}$.

3.2 Machine Learning Algorithms

Gradient Boosting decision tree (GB) [5] is a powerful model for both regression and classification problems. The model consists of a number of individual decision trees and each tree is trained by the residual of previous trees. The GB model is formulated as:

$$y(x) = \sum_{i=1}^{M} T_i(x) \tag{5}$$

where M is the number of trees and T_i is the i-th decision tree which maps x to the corresponding leaf with a weight.

The reasons of selecting GB are as follows. Firstly, it has strong capacity to model interactions among features. Secondly, compared with SVM etc., its algorithm is appropriate for the large-scale dataset and usually used by the teams in RecSys 2015 Challenge.

4. Experiments

4.1 Dataset

RecSys 2015 Challenge provides a large number of sessions from an online e-commerce retailer [1]. Our experiment is based on this dataset. As shown in Table 1, the dataset is large-scale and extremely imbalanced. There are 33 million sessions and only 5.5% of those session is buy sessions. The average number of clicks per session is about 4. It means that the click information per session is less and just 4 clicks in average is used to predict whether a session is a buy session. About 38% of all unique items are bought at least once. More than a half of all items are not purchased at all, so it is not necessary to include all items in the prediction. In all experiments, we use the train-part dataset to learn a model and tune all parameters by the valid-part dataset. The experimental results are reported on testing dataset.

4.2 Experimental Setting

We use four groups of features: Basic (B), Temporal Dynamics (TD), Collaborative Filter (CF) and Temporally Popular Items (TPI) features. We list B and TD features in Table 2. We use CF features by following the study [11]. B, TD and CF are usually used as traditional features. There are 52,739 unique items in training dataset, so up to 52,739 items are included in TPI features. We tune parameters with the valid-part dataset.

We use the precision, recall and F1-score to evaluate the prediction models. As an average of the precision and recall, F1-score is used as the evaluation metric to tune parameters on the valid-part dataset. Normalization is used to scale the range of features' value and make the converge faster. We use two-sided paired approximate randomization tests to assess the statistical significance of the difference between two F1-scores with a significance level of $p = 0.05$.

Table 1 Statistics of dataset.

dataset	click	session	item
training	33,003,944	9,249,729	52,739
testing	8,251,791	2,312,432	42,155
buy	1,150,753	509,696	19,949
train-part	24,754,609	6,937,297	50,678
valid-part	8,249,335	2,312,432	41,994

Table 2 Session features. B denotes basic features and TD is features of temporal dynamics.

group	feature	description
B	click	#(clicks in the session)
	Max Click	max #(clicks on any item)
	Time Span	time span of the session
	Max Span	max time span on any item
	Item	#(unique items clicked)
	category S	#(unique items in category S)
	category 0	#(unique items in category 0)
	category 12	#(unique items in category from 1 to 12)
TD	day of year	day when the last item is clicked
	day of week	day of week when the last item is clicked
	month	month when the last item is clicked
	hour	hour when the last item is clicked

To compare the efficiency of TPI, we also use LR and FM in experiments. For Logistic Regression(LR), we use LibLinear[†]which is a library for the linear classification. The standard *L*2 regularization is used in the experiments. For Factorization Machines(FM), we use LibFM[††] which is an implementation of factorization machines. Markov Chain Monte Carlo is used to learn the model. For Gradient Boosting decision tree(GB), we use the XGBoost[†††] library. We concentrate on the binary classification with the logistic loss function, and use tree boosters and *L*2 regularization term on weights.

4.3 Result and Discussion

Our first baseline is the Collaborative Filter (CF) model based on matrix factorization. The purchase is predicted by:

$$f(s) = \mu + \sum\nolimits_{k \in s} b_k + \sum\nolimits_{i,j \in s : i \neq j} \left\langle v_i, v_j \right\rangle \qquad (6)$$

where μ denotes the global bias, b_k is the bias of the item k in the session s, v_i is a vector of the item i in the latent factor space.

We also experiment with the trend model leveraging the temporal characteristics of both the session and the items clicked in that session [2]. The trendiness of the session, the number of clicks in the session, session dwell time and session start hour are features. We use the train-part dataset to model the trendiness of each item and compute the session trendiness feature. The gradient boosting decision tree is used to train the predictive model on the valid-part dataset and we evaluate the model on the testing dataset. Note that we do not use over-sampling and under-sampling techniques to balance the dataset.

As shown in Table 3, collaborative filter model is based on matrix factorization technique which takes into account items in a session and interactions between items, but ignore the temporal dynamics of the session. Trend model computers the trendiness of each of all items as the temporal feature and can get the better precision. TPI consider only temporally popular items, and lack of items, which are not temporally popular, leads to the decrease of the precision. Our approach can improve the recall efficiently by bringing temporally popular items, which consider both the temporal characteristics and popularity of items clicked in that session. Our model is significantly better than those baselines.

Table 3 Performance comparisons with baselines. dim is the dimensionality of the feature. ALL indicate that all items are included as features. The more stars denote a significant improvement.

model	feature	dim	precision	recall	F1-score
CF	ALL	52739	0.1323	0.3009	0.1837
GB	Trend+Other	27	0.2540	0.2281	0.2403*
GB	TPI	2500	0.2290	0.3142	0.2649**

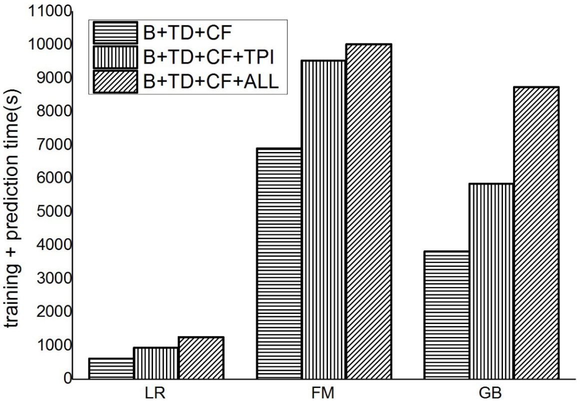

Fig. 1 Efficiency comparisons. 0 indicates that none of items is used. TPI means TPI are used as features. All means that all items are used as features.

Table 4 TPI performance comparisons with different machine learning algorithms. ALL features indicate that all items are included as features. The star means a significant improvement than B+TD+CF.

model	feature	dim	precision	recall	F1-score
LR	B+TD+CF	233	0.1937	0.3702	0.2543
	B+TD+CF+TPI	2733	0.2234	0.4397	0.2963
	B+TD+CF+ALL	52972	0.2308	0.4341	0.3013
FM	B+TD+CF	233	0.2135	0.3414	0.2627
	B+TD+CF+TPI	2733	0.2767	0.3958	0.3257*
	B+TD+CF+ALL	52972	0.2758	0.3834	0.3208*
GB	B+TD+CF	233	0.2180	0.3620	0.2722
	B+TD+CF+TPI	2733	0.2762	0.4441	0.3406*
	B+TD+CF+ALL	52972	0.2955	0.4014	0.3404*

In Fig. 1, we study the efficiency using the total time of training and prediction. The training time does not include the time used to extract features. We draw the consistent conclusion that the larger the number of dimension of feature is, the more time is spent for training the model and prediction. As shown in Table 4, we conduct experiments with the different algorithms. TPI can achieve comparable performance of all items. The TPI features are a better choice while considering both the effectiveness and the efficiency.

4.4 Feature Contribution

In this subsection, we study the contribution of each group of features to understand the relation of our proposed TPI

[†] https://www.csie.ntu.edu.tw/~cjlin/liblinear/

[††] http://www.libfm.org/

[†††] https://github.com/dmlc/xgboost/

Table 5 Experimental results of feature groups.

	model	feature	precision	recall	F1-score
LOO	GB	B+TD+CF+TPI	0.2655	0.4406	0.3314**
	GB	TD+CF+TPI	0.2432	0.3458	0.2856*
	GB	B+CF+TPI	0.2632	0.3694	0.3074*
	GB	B+TD+CF	0.2414	0.3906	0.2984*
	GB	B+TD+TPI	0.2676	0.4001	0.3207*
AOI	GB	B	0.1952	0.3416	0.2484*
	GB	TD	0.0716	0.5029	0.1253*
	GB	TPI	0.2290	0.3142	0.2649**
	GB	CF	0.1659	0.3509	0.2252*

features with other groups of features. The leave-one-out and add-one-in methods are used to evaluate the contribution of a group of features. Leave-one-out method means removing one group of features from all features each time while the add-one-in method denotes only using one group of feature each time. Gradient Boosting decision tree model is used in the experiments.

As Table 5 shown, TPI get the highest precision and lowest recall in add-one-in experiments. The possible reason is the sale promotion in E-commerce. In other words, if some items are sold as a bundle at a low price during a certain period, this kind of sale promotion can be captured more exactly by the TPI features than other groups of features while those promotions account for the fewer part of all purchase behaviors. With the evaluation of F1-score, TPI achieve the best performance while removing TPI in leave-one-out leads to the second largest decline. It proves that TPI are the most effective features and irreplaceable by other groups of features. B group of features is the second most effective in add-one-in and bring the biggest drops of F1-score in leave-one-out. It indicates that basic features have an independent effect on the prediction. Compared with TPI and B group, although CF is effective in add-one-in, they can be replaced by other groups. TD features have the worst performance in add-one-in experiments, but they are indispensable. Thus, compared with B, TD and CF features, TPI features are novel and effective. They are able to combine the advantages of the popularity and temporal dynamics of items and play an irreplaceable role in the prediction.

5. Conclusions

In this paper, we explore to use temporally popular items for purchase behavior prediction. Experimental results show that our method is significantly better than baselines. The features based on temporally popular items are a supplement of traditional features and a promising way to improve the prediction. We also study the contribution of each group of features to understand the relation of our proposed TPI features with other groups of features.

Although our work use a real-world dataset, our method is limited to the characteristics of the dataset. The few dataset is available, but it is necessary to conduct experiments in more datasets of different domains. In the future, it is interesting to explore other features in different datasets for the purchase prediction.

Acknowledgments

Research was partially supported by National Natural Science Foundation of China (No. 61402242, 61402333, 61402243), the Research Foundation of Ministry of Education and China Mobile Communications Corporation (No. MCM20150507), the Tianjin Municipal Science and Technology Commission (No. 14JCQNJC00200).

References

[1] D. Ben-Shimon, A. Tsikinovsky, M. Friedmann, B. Shapira, L. Rokach, and J. Hoerle, "RecSys Challenge 2015 and the YOOCHOOSE Dataset," Proc. RecSys Challenge, pp.357–358, 2015.

[2] V. Bogina, T. Kuflik, and O. Mokryn, "Learning Item Temporal Dynamics for Predicting Buying Sessions," Proc. 21st International Conference on Intelligent User Interfaces, pp.251–255, 2016.

[3] C. Chen, C. Hou, J. Xiao, and X. Yuan, "Purchase Behavior Prediction in E-Commerce with Factorization Machines," IEICE Transactions on Information and Systems, vol.E99-D, no.1, pp.270–274, 2016.

[4] R. Dias, and M.J. Fonseca, "Improving Music Recommendation in Session-Based Collaborative Filtering by Using Temporal Context," Proc. IEEE 25th International Conference on Tools with Artificial Intelligence, pp.783–788, 2013.

[5] J.H. Friedman, "Greedy function approximation: A gradient boosting machine," Annals of Statistics, vol.29, no.5, pp.1189–1232, 2001.

[6] F. Kooti, K. Lerman, L.M. Aiello, M. Grbovic, N. Djuric, and V. Radosavljevic, "Portrait of an Online Shopper: Understanding and Predicting Consumer Behavior," Proc. WSDM, pp.205–214, 2016.

[7] Y. Koren, "Collaborative Filtering with Temporal Dynamics," Proc. SIGKDD, pp.447–455, 2009.

[8] S.E. Park, S. Lee, and S.-G. Lee, "Session-Based Collaborative Filtering for Predicting the Next Song," Proc. 1st ACIS/JNU International Conference on Computers, Networks, Systems and Industrial Engineering, pp.353–358, 2011.

[9] P. Romov and E. Sokolov, "RecSys Challenge 2015: Ensemble Learning with Categorical Features," Proc. RecSys Challenge, Article no.1, 2015.

[10] Y. Tanaka, T. Kurashima, Y. Fujiwara, T. Iwata, and H. Sawada, "Inferring Latent Triggers of Purchases with Consideration of Social Effects and Media Advertisements," Proc. WSDM, pp.543–552, 2016.

[11] M. Volkovs, "Two-Stage Approach to Item Recommendation from User Sessions," Proc. RecSys Challenge, Article no.3, 2015.

[12] P. Yan, X. Zhou, and Y. Duan, "E-Commerce Item Recommendation Based on Field-aware Factorization Machine," Proc. RecSys Challenge, Article no.2, 2015.

[13] Y. Zhang and M. Pennacchiotti, "Predicting Purchase Behaviors from Social Media," Proc. WWW, pp.1521–1532, 2013.

[14] E. Zheleva, J. Guiver, E. Mendes Rodrigues, and N. Milić-Frayling, "Statistical Models of Music-listening Sessions in Social Media," Proc. WWW, pp.1019–1028, 2010.