Personalized Feature based Re-Ranking Method for E-commerce Search at CIKM Cup 2016

Deqiang Kong
Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing
Jiaotong University
15120335@bjtu.edu.cn

Jingyuan Tang
Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing
Jiaotong University
15120349@bjtu.edu.cn

Yao Zhao
Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing
Jiaotong University
yzhao@bjtu.edu.cn

Zhenfeng Zhu
Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing
Jiaotong University
zhfzhu@bjtu.edu.cn

ABSTRACT

Web search engines often return the same result pages for different people which they search the same session but for different needs, or return the different pages for people which they search the same needs in different ways. So it is crucial for the users to personalized re-rank the feedback of the search, especially in e-commerce, it helps to improve conversion rates of the items. Although there are many algorithms are proposed, this problem is still a big challenge. The CIKM Cup committee hosts an international competition for predicting the priority among the items based on the large scale search and browsing logs, transaction records and item catalog. Our solution is mainly focus on the deep feature engineering including user feature, item feature and user-item interaction feature. Finally, as published by official, our solution won the $3rd$ place in the whole competition.

Keywords

personalized feature; re-ranking; e-commerce

1. INTRODUCTION

With the fast development of the web, millions of users are experiencing a flood of information by interacting with search engines daily. They browse the web pages, click on ads or issue queries, and then they follow some of the links of the results. For the personalized search challenge, if we could match the result pages with the user's queries, we will optimize the retrieval function [7], in the long term, it can effectively increase the users stickiness [1].

Unfortunately, The majority of people are not willing to give explicit feedback, it becomes more difficult to match them with the result pages, so it's a hot research field [8, 3] and a challenge for academia and industry researchers to work up a new method. In order to attract more people to get involved into the personalized e-commerce search, the CIKM Cup committee[1] provides a unique opportunity for them to come up with a new idea to test, and the participants could implement and consolidate the approach that are existing and described. The CIKM Cup committee hold the competition named Personalized E-commerce Search Challege[2].

The competition was continued for two months and we won the $3rd$ place at last. Our solution focus on the feature engineering for two sessions, the one session is search engine result pages (SERPs) returned in response to a query called query-full, and the other one is SERPs returned in response to the user click on some item category called query-less. According to analysis of the dataset provided by organizer, we extract 7 types of features in total to describe the users, categories, items and some interactions from different aspects. In order to consider the characters in the two sessions, we need prioritize the 7 features to re-rank the personalized search.

The rest of the paper is organized as follows. Section 2 gives the problem description. Section 3 describes the 7 feature entities we have generated in detail. The results of the experiments are listed in the Section 4. Finally, Section 5 concludes the paper.

Table 1: Statistics of action types

#views	#clicks	#purchases
2,451,565(1.83%)	1,877,542(1.40%)	68,818(0.05%)

2. PROBLEM DESCRIPTION

For the personalized e-commerce, the CIKM committee provides a dataset contains a lot of information about user, item and category. We should predict relevance labels and re-rank items by an e-commerce search engine on the SERP.

[1]http://cikmcup.org/
[2]https://competitions.codalab.org/competitions/11161

CIKM Cup '16 October 24–28, 2016, Indianapolis, USA

DOI: 10.475/123_4

Table 2: Statistics of log activity data

#real user	#anonymous	#item	#cate	#session
232,817	333,097	185,047	1,217	573,935

Table 3: Statistics of sessions and queries

data	1-sess	1-sess/1-query	query-full	query-less
train	115,631	93,387	15,615	600,545
test	113,459	92,227	16,273	270,694

The following data is provided: for items, there are names, prices and categories; for user, there are searching tokens and time of browse; for user-item interaction, there are number of clicks, views and purchases and so on.

The competition is carried out in two phases, the organizer random select 50% test data in every phase, and the final submissions are evaluated in Normalized Discounted Cumulative Gain (NDCG) measure.

Table 1 shows the statistics of the total number of different type of actions, and the items that interact with actions are accounting for the percentage of the total items which are shown in brackets. Table 2 shows the statistics of the user information and the item information, the number of the anonymous is largely outweighed the number of users who have real name ID, so it's increasing the difficulty for competition. Because the anonymous have no historical data to trace, and it's also a hard problem in recommendation system [2]. Table 3 contains the query-less and the query-full in sessions, apparently the total number of query-less in all sessions occupied over 90%, so we can infer why the weight distribution of the query-less is more heavily in the Eq.(4). And there are a large number of users who only have 1 query in the whole time period, it is a well know problem called cold start problem which is a hot issue and many works [4] are dedicated to study this problem. In the next section, we will describe how we take advantage of this dataset and extract a huge amount of features.

3. FEATURE ENGINEERING

The characteristics of 5 entities in Table 2 and their interactions are useful to compare the relative importance of the items, that we could base on the importance to re-rank the items. For example, if a user have viewed, clicked or even purchased a item, it shows that the user is more likely search this item than others, so we hold that the item have higher priority and put it in the front. To describe the characteristics of the 5 entities and their pairwise interactions [6, 5], we generated 7 personalized features which could be grouped into 3 feature groups: user feature, item feature and user-item interaction feature. In the rest of this section, we will introduce the 3 feature groups in detail.

3.1 User Feature

User feature is the statistics analysis of the user historical data, it reflects the preference of different users, and we can generate personalized results according to this preference.

- **Category Period.** Category period is the time that spend on the browsing item categories by a user. This is a simple idea that if a user spend more time in this item, the item is more likely to be purchased by user than others. For example, a user spend 1 hour in category *phone* and 5 minutes in category *car*, so we can guess this user prefer *phone* than *car*. According to this preference, we can recommend more items in category *phone*.
- **Price Preference.** Price preference is the average price of items which the users often browse. This feature is mainly to help merchants to distinguish the high-end customers and the low-end customers. According to the different level of users, the merchants can recommend the different level of items for customers. For example, the price range of a particular category is $[0, 1000]$, and the average purchase price for two users are 200 and 800 respectively. We guess in another category which price range is $[0, 2000]$, these two users perfer to purchase items which price are around 400 and 1600. According to this preference, we can recommend to the users with the appropriate price of items.

3.2 Item Feature

Item feature reflects intrinsic properties of the items, and it directly represents the popularities of the items based on the statistical method. We conclude three item features which are global popularity, local popularity and category number to judge the priority of the items.

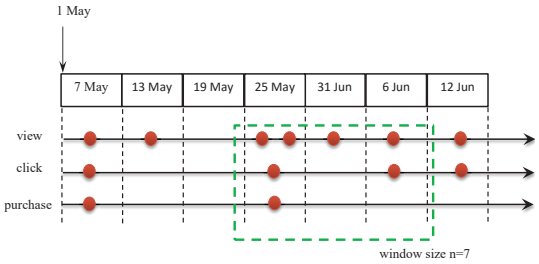


Figure 1: Action history of an example entity

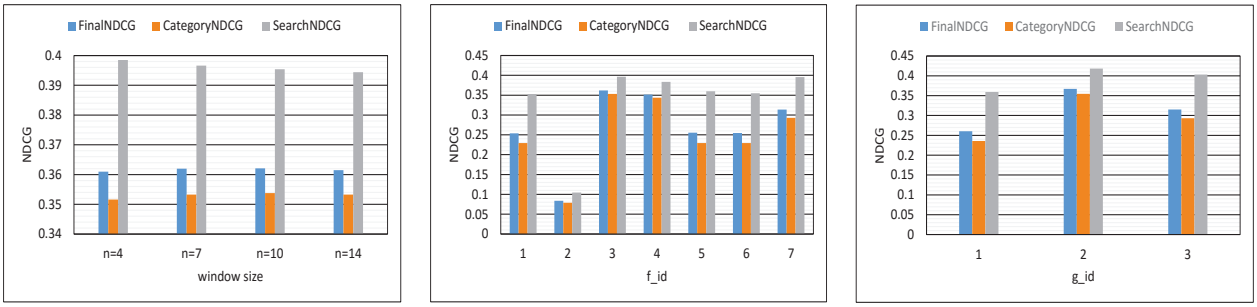
- **Global Popularity.** Global popularity is calculated by the historical data of the total items during six months, and the aim of this feature is to solve the cold start problem. According to Table 1, there are three item actions including view, click and purchase. We give different weights for these actions and then sum together to calculate the popularity. The weight allocation is on the basis of the importance of the actions, the experience shows that purchase behavior is the most important, and view behavior is the least important. The *global popularity* is the weighted sum of click counts, view count and purchase counts, so this feature is computed as:

$$GP_i = w_1 \times v_i + w_2 \times c_i + w_3 \times b_i \quad (1)$$

where GP_i is *global popularity* of item i, and v_i, c_i and b_i represent number of view, click and purchase respectively. For the entity shown in Figure 1, the weekly view counts are $(1, 1, 0, 2, 1, 1, 1)$, the weekly click counts are $(1, 0, 0, 1, 0, 1, 1)$, and the weekly purchase counts are$(1, 0, 0, 1, 0, 0, 0)$. The overall counts of view, click and purchase are 7, 4, 2 respectively. So the *global popularity* is $w_1 \times 7 + w_2 \times 4 + w_3 \times 2$.

Table 4: Summary of features

g_id	feature group	f_id	feature entity	description
1	user feature	1	category period	the duration of querying items by a user
		2	price preference	average price of items which were clicked, viewed and purchased by a user
2	item feature	3	global popularity	the global popularity of items
		4	local popularity	a window size of global popularity
		5	category number	number of items in different categories return by the search engine
3	user-item feature	6	search token	number of matches between search tokens and name of items
		7	user-item score	weighted sum for number of clicks, views and purchases by a user



(a) Results of different windows size n (b) Results of feature entity (c) Results of feature group

Figure 2: Results of different feature entities and groups

- **Local Popularity.** Local popularity is a window over the whole time period, the goal of this feature is also to solve the cold start problem. For the entity shown in Figure 1, as this feature, we choose a windows size n, in this case, $n = 7$. The view counts are $(2, 1, 1)$ in the windows, the click counts are $(1, 0, 1)$, and the purchase counts are $(1, 0, 0)$. The overall counts of view, click and purchase in the windows are 4,2,1 respectively. The *local popularity* is computed as Eq.(1) shows.

- **Category Number.** Category number is the statistics of the item categories return by the search engine. We think the original results of SERPs make some sense. So, the goal of this feature is to calculate the original information for search engine. In this way, we are able to count the number of the item categories and the number of items in each category, we use C_i to represents the number of the category i. The priority of each item is justified on the basis of the number of its category. For a simple example, the default feedback of the items are $item\ list = (1, 2, 3, 4)$, and the $item\ list$ belong to $category = (1, 1, 2, 3)$ respectively. So the $C_1 = 2$, $C_2 = 1$ and $C_3 = 1$. It is obvious that the priority of the items belong to category 2 is highest.

3.3 User-item Interaction Feature

User-item interaction feature is the impact of user behavior on items, it does not focus on accurately predicting the relevance degree of each aspect, instead, it cares about the relation between the two aspects [6]. So it includes item properties and user properties.

- **Searching Token.** Searching token is only useful for the test queries of the query-full. If a user search one item more times than others, it represents this item is more reasonable and satisfactory for this user. And we simply calculate the *searching token* by counting the matches between the tokens that user searched and the name of items.

- **User-Item Score.** User-Item score is mainly targeted to personalization recommendation by analyzing the personal preference. The concrete methods are as follows: For the popular items which were viewed, clicked or even purchased, we calculate the item popularities for users by the method of weight sum. This weight sum is called *user-item score*, and we also call these items *preference items*. For the items without user interactions, we generate the session-item matrix, and the value of matrix is computed as overall counts of click, view and purchase. Then we take advantage of the column vectors of two items to calculate the cosine similarity as Eq.(2) shows.

$$sim(i, j) = \frac{\vec{v}_i \cdot \vec{v}_j}{|\vec{v}_i| \times |\vec{v}_j|} \tag{2}$$

where $sim(i, j)$ is the cosine similarity between item i and item j according to the session-item matrix, and $\vec{v}$ is column vector of matrix. Then we can calculate the *user-item score* of those items without user interaction as Eq.(3) shows.

$$S_{i,j} = \frac{\sum_{k \in PI} sim(j, k) \times S_{i,k}}{\sum_{k \in PI} sim(j, k)} \tag{3}$$

where $S_{i,j}$ means *user-item score* for user i in item j, and PI means *preference items* as we have described above.

4. EXPERIMENTS

In this section, we evaluate the single feature and their combinations, and then we pick out the best combination of

Table 5: Results of different feature order

order(g_id)	NDCG	NDCG search	NDCG category
1-2-3	0.3622	0.3472	0.4222
1-3-2	0.3829	0.3686	0.4397
2-1-3	0.3672	0.3545	0.4181
2-3-1	0.3673	0.3545	0.4181
3-1-2	0.4015	0.3880	0.4558
3-2-1	**0.4056**	**0.3928**	**0.4570**

feature groups. The experiments have 3 scenarios including single feature, feature group and order of feature groups.

According to empirical observation, the action of purchase is the most important in these three actions (click, view and purchase). Based on several simple experiments, we assign the weight of view, click and purchase as $w_1 = 1, w_2 = 2, w_3 = 4$ in Eq.(1).

First experiment is to evaluate our single feature entity, and we use feature entities one by one to re-rank the default items. For evaluating the best windows size, we choose the windows size $n = 4, 7, 10, 14$, as the Figure 2(a) shows, we can figure out that windows size $n = 7$ has the best performance. As Figure 2(b) shows, the feature *price preference* has lowest performance among 7 feature entities, and the score of *global popularity* and *local popularity* achieves the highest performance. This may because the large number of cold start users and the number of purchase actions is only about 70k as shows in Table 1.

Second experiment is to evaluate the feature groups one as shown in Figure 2(c). We can find that the result of 3 feature entities in item feature is very well, and the combination of these 3 features gets highest performance among 3 feature groups. On the contrary, the result of feature entities in user-item interaction feature is not good, and then the combination of 2 feature entities in user-item interaction feature group gets lowest performance.

Last experiment is to evaluate feature groups with various permutations and combinations. We don't change the order in the every feature group, but exchange the order between the features groups. The scores of NDCG are shown in Table 5, and the order of *user-item interaction feature, item feature and user feature* works best.

According to the competition rule, the final NDCG score is weighted sum of search NDCG and category NDCG as Eq.(4).

$$Score = 0.8 \times NDCG_c + 0.2 \times NDCG_s \quad (4)$$

where $NDCG_c$ and $NDCG_s$ represent the NDCG score of query less and query full. Finally, our approach achieves the $3rd$ rank.

5. CONCLUSION

In this paper, we presented our solution for the personal e-commerce search challenge. We generated many feature entities and combined them to capture factors which may influence the rank of items. Our results in Figure 2(b) shown that none of a single feature can obtain a good score, so we need to combine them together to achieve a relative higher NDCG score. As shown in Figure 2(c), the item feature is the most effective feature, we guess it is a possible outcome of the data bias, because the interactive information is relative small. What's more, the time interval is a little short. The Table 5 shows the best order of the feature groups. Our experiment results win the $3rd$ place in the final evaluation. However, the performance is not good enough, we will do more in-depth study and research in the future.

6. ACKNOWLEDGMENT

The authors would like to thank the CIKM Cup committee for hosting such a meaningful competition. This work was supported in part by the National Natural Science Foundation of China(No.61532005 and No.61572068), the Program for New Century Excellent Talents in University(No.13-0661).

7. REFERENCES

[1] E. Agichtein, E. Brill, and S. Dumais. Improving web search ranking by incorporating user behavior information. In *SIGIR 2006: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, Usa, August*, pages 19 – 26, 2006.

[2] Y. Almurtadha, M. N. B. Sulaiman, N. Mustapha, N. I. Udzir, and Z. Muda. Ars: web page recommendation system for anonymous users based on web usage mining. In *European Conference of Systems, and European Conference of Circuits Technology and Devices, and European Conference of Communications, and European Conference on Computer Science*, pages 115–120, 2010.

[3] W. Di, A. Bhardwaj, V. Jagadeesh, R. Piramuthu, and E. Churchill. When relevance is not enough: Promoting visual attractiveness for fashion e-commerce. *Computer Science*, 2014.

[4] M. Elahi, F. Ricci, and N. Rubens. *Active Learning in Collaborative Filtering Recommender Systems*. Springer International Publishing, 2014.

[5] G. Liu, T. T. Nguyen, G. Zhao, W. Zha, J. Yang, J. Cao, M. Wu, P. Zhao, and W. Chen. Repeat buyer prediction for e-commerce. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[6] T. Y. Liu. Learning to rank for information retrieval. *Foundations & Trends in Information Retrieval*, 3(3):225–331, 2011.

[7] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.

[8] H. Wang, X. He, M. W. Chang, Y. Song, R. W. White, and W. Chu. Personalized ranking model adaptation for web search. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 323–332, 2013.