IOP Conf. Series: Materials Science and Engineering **261** (2017) 012019 doi:10.1088/1757-899X/261/1/012019

# Research on Long Tail Recommendation Algorithm

**Xuezhi Hu[1,a], Chuang Zhang[1,b], Ming Wu[1,c] and Yang Zeng[1]**

School of Information and Communication Engineering, Beijing University of Posts and Telecommunications
Xitucheng Road No.10, Beijing, P.R.China

[a]huxuezhi@bupt.edu.cn; [b]zhangchuang@bupt.edu.cn; [c]wuming@bupt.edu.cn

**Abstract.** Most recommendation systems in the major electronic commerce platforms are influenced by the long tail effect more or less. There are sufficient researches of how to assess recommendation effect while no criteria to evaluate long tail recommendation rate. In this study, we first discussed the existing problems of recommending long tail products through specific experiments. Then we proposed a long tail evaluation criteria and compared the performance in long tail recommendation between different models.

## 1. Introduction
Recommender systems enable recommendations to be made to users of a system in reference to the items or elements. Currently, collaborative filtering [1] is the most commonly used and studied technology. One difficult, though common, problem for a recommender system is that most of the recommended information is concentrated in a small number of population items, which we called "Long tail phenomenon" [2][3]. Hence we proposed a detailed long tail evaluation criteria including long tail recommendation rate, long tail stability and depth which satisfies the quantitative evaluation of long tail recommendation.

In this paper, we first carried out a detailed feature engineering work. Then we compared GDBT model [4] and LR model to evaluate the performances in recommendation. The result shows that GBDT model achieves a better performance. In order to measure the utilization rate and overall performance of the long tail data, we proposed a new evaluation criteria system and also compared two traditional models with a long tail recommendation algorithm based on user-entropy LDA model [5][6][7], which improves the long tail recommendation rate and stability ensuring the traditional recommended performance not significantly reduced.

## 2. Data feature engineering

### *2.1. Feature extraction*
The dataset we use in this paper comes from the user behavior log of Tmall, an E-commerce platform in 200 days, which contains the basic information of users and items, also the user behaviors. The amount of the data is over 200,000 and the total size of data is 700MB. In the initial research data, the information mainly includes users(user_id), purchase goods, brands(brand_id), category information (cat_id), behavior date(date), and types of user behaviors(click_cat) and so on.

According to the characteristics of Tmall data, we first divide the features into three categories. The first category is information of user characteristics, including the number of active days, the frequency of purchases, the total number of purchases and the conversion rate, which mainly reflects the influence of the user entity itself on E-commerce platform. The second category is information of behavior characteristics, including the average number of visits, the number of corresponding behaviors, and the days of these behaviors, which reflects the behaviors of users in using E-commerce platform. The last category is bag-of-words information, including browsing brands, categories of sellers and time, which mainly reflects the information of the platform itself.

After the preliminary analysis of the data features, as is shown in Table 1, we found that the amount of purchase goods is too much to cluster which is unable to extract the valid feature information. In the meantime, the user behaviors such as “Add to cart” can be ignored.

**Table 1.** F1 value of different models.

| Feature types | Amount | User behaviors | Amount |
|---|---|---|---|
| Purchase goods | 61553 | Click | 10928941 |
| Types of goods | 1226 | Add to cart | 41 |
| Sellers | 4275 | Purchase | 751502 |
| Brands | 6304 | Favorite | 682479 |

According to the data model given by the research object, we carried on detailed extraction including population preference and date of behaviors analysis. With the statistical characteristics of timing sequence, we can extract the characteristics of special festivals and cycle of crowd behaviors, inferring the basic cycle and time nodes of different people shopping.

Research on crowd preference is also an important part of feature extraction. We extracted some special brands and analyzed after normalization. The result shows a large number of brands do have the characteristics of the crowd preference. As is shown in Figure 1, the distribution of purchase or click are more concentrated on specific crowds. After screening the feature information, a total of 770 dimension features were extracted, including brand characteristics, seller behaviors, user behaviors, etc.

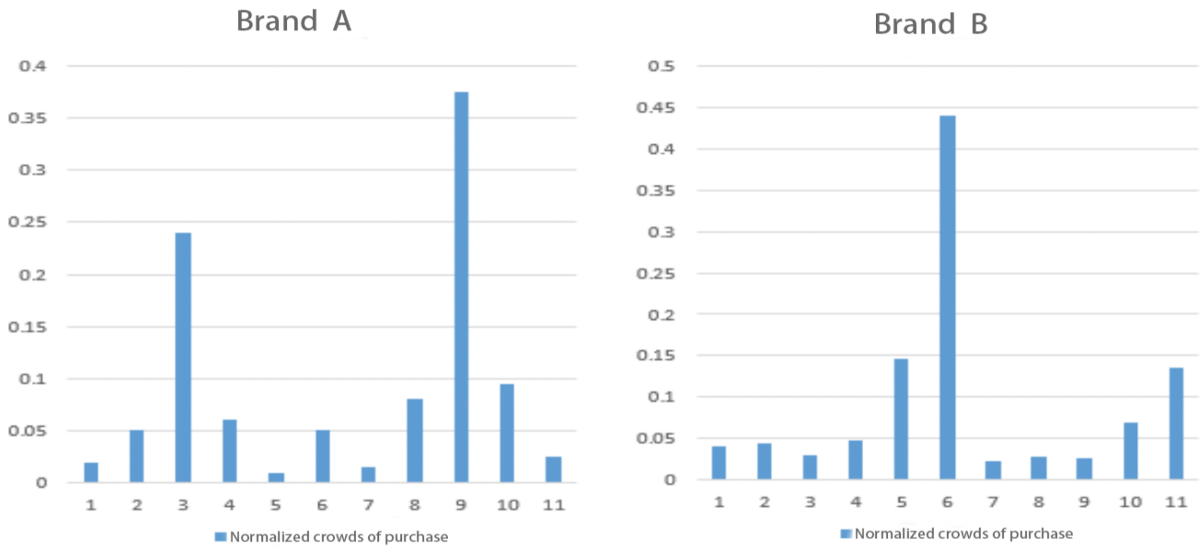


**Figure 1**. Statistics of purchase behavior in different crowds

*2.2. Feature selection*

The features are dimensioned by the calculation of similarity measure to achieve the purpose of dimensionality reduction, considering the correlation and redundancy between the features. We chose the Pearson correlation coefficient [8] as the relevance metrics. The specific approach is to use the Pearson correlation coefficient to estimate the correlation of each feature dimension. We reduce the dimension of the features by setting appropriate threshold, and finally select the relatively high correlation characteristics as the main features to train the model.

The Pearson correlation coefficient is defined as the ratio between the covariance and the standard deviation of random variables X and Y:

$$\rho_{X,Y} = \frac{Cov(X,Y)}{\sigma_X \sigma_Y} = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \quad (1)$$

We defined the user behaviors as variable X and the multidimensional features as variable Y. The user behaviors include click, favorite, add to cart and purchase. By calculating the correlation user behaviors and the features, the Pearson correlation coefficient is gained. Part of the experimental results are shown in Table 2. The features of the final input data set were reduced to 50 dimensions using Pearson correlation coefficient to evaluate the relativity of each dimension. They are divided into three categories to characterize the entire model, as is shown in Figure 2.

**Table 2.** Part of Pearson correlation coefficient results

| | Active days | Average pre-purchase clicks | Conversion rate | Browse categories | Purchase rates | Retention rates |
|---|---|---|---|---|---|---|
| **Click** | -0.148 | 0.074 | 0.011 | -0.103 | 0.112 | 0.078 |
| **Add to cart** | 0.012 | -0.108 | 0.092 | -0.067 | 0.103 | -0.131 |
| **Purchase** | 0.078 | 0.186 | -0.156 | 0.078 | -0.082 | 0.054 |
| **Favorite** | -0.053 | 0.153 | 0.12 | -0.065 | -0.197 | -0.03 |

## 3. Long tail evaluation criteria

### *3.1. Related works*

Considering the high solution speed of LR model and the comprehensiveness of GBDT model, we discussed two fusion programs. A LDA model based on user-entropy is also mentioned.

*3.1.1. Cascade model.* We use GBDT to build two types of trees, non-behavior features tree and behavior features tree [9]. The non-behavior features tree is base which means non-behavior features can be based on this model to get the corresponding information. The behavior features tree is used to identify the features and feature combinations corresponding to behaviors. The mapping relations are shown in Figure 3. When a sample X comes in, the resulting feature is input to LR after traversing two types of tree-to-leaf nodes, and the non-behavior feature tree can be used as a supplemental tree.

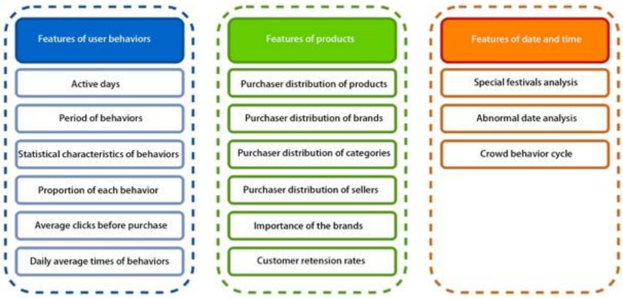


**Figure 2.** Feature classification results

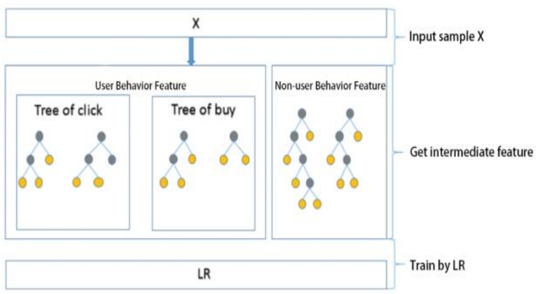


**Figure 3**. Cascade model processing

*3.1.2. Artificial fusion.* First of all, we extract N users from user set A to train the GBDT model, then split out M users to recommend using this model. We retrain the inaccurate part using LR model, the inaccurate part in the second part will be trained using GBDT model again. Repeat the above steps until all the users in the data set are removed.

The result is shown in Table 3. We use F1 value [10] to represent the performance of recommendation. It shows the cascade model performs better than any other models. However, according to statistics, only 30% of the recommended items come from the long tail part (We classified the items in proportion to 8:2 and the small part is long tail part), which means the existing evaluation indexes cannot evaluate the long tail characteristics.

**Table 3.** F1 value of different models.

| | LR | GBDT | Artificial Fusion | Cascade Model |
|---|---|---|---|---|
| **F1** | 0.312 | 0.321 | 0.319 | 0.336 |

*3.1.3. EN-LDA model.* The LDA model is a method of modeling the subject information in the text data [11]. It assumes that the document set can be divided into a number of implicit topics, and the topology of these topics is linear. Using the probabilistic inference algorithm, a single document can be represented as a mixture of these specific topics.

We use the product scores to measure the strength of the relationship between the user and the goods. A user's preferences for different items may not be the same, for example, he may give a high score to his favorite goods. We use $w(u, i)$ to represent the score of $\mathrm{item}_{u,i}$ ranks $i$. In the algorithm, as is shown in Figure 4 [2], $w(u, i)$ represents the frequency of appearance in goods set $S_u$, $\theta$ represents the theme distribution of each user, and $\psi$ represents the distribution of each good.

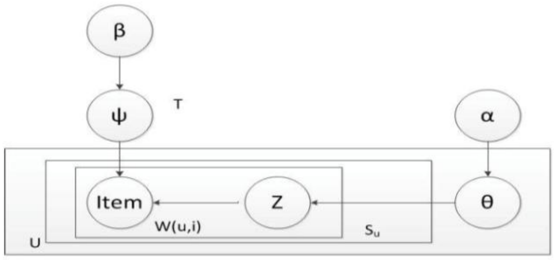


**Figure 4**. LDA model

The parameters of the EN-LDA [2] model are determined by using Gibbs sampling [12], and finally give the distribution of each user's theme. Use Gibbs sampling to modify the subject distribution iteratively until the model parameters converge.

*3.2. Long tail recommendation rate*

Long tail recommendation rate refers to the percentage of recommended products in the "long tail" part after classifying according to 8:2 way [13] in unit time.

$$\mathrm{Longtail} = \frac{item_l}{item_{all}} \tag{2}$$

where $item_l$ represents the number of products in long tail, and $item_{all}$ represents the total number of recommendation. Long tail recommendation rate describes the long tail data utilization in recommendation systems.

*3.3. Long tail stability*

In addition to considering the long tail performance, the main task of the recommended system is to recommend the most reliable and practical product for the user. Based on this, we present the recommendation stability:

$$\mathrm{Stability} = \frac{Longtail}{Accuracy} + Accuracy \tag{3}$$

where *Longtail* represents the long tail recommendation rate and *Accuracy* represents the accuracy of recommendation. The long tail recommended stability represents the balance between recommendation performance and long tail utilization, which comprehensively reflects the recommended metrics for the system.

### *3.4. Long tail depth and long tail level*

Long tail depth is mainly used to describe the state of current system using the long tail data. As shown in Figure 5, we divide the long tail depth into three stages: initial long tail, intermediate long tail and deep long tail. The long tail level is divided into three according to the value of long tail recommended rate. Table 4 shows the definition of long tail.

**Table 4.** Definition of long tail

| Conditions | Long tail level |
|---|---|
| Recommendation rate of deep long tail ≥ 10% | Third level |
| Not third level & Recommendation rate of intermediate long tail ≥ 10% | Second level |
| Not second level & Recommendation rate of initial long tail ≥ 10% | First level |
| Others | Non-Longtail |

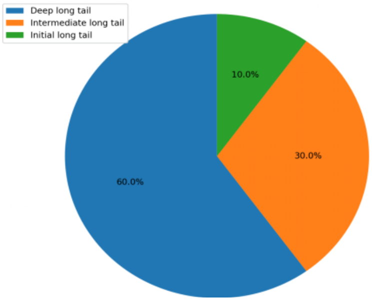


**Figure 5.** Long tail depth distribution

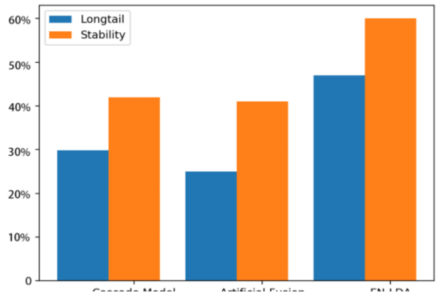


**Figure 6**. Long tail indicator between different models

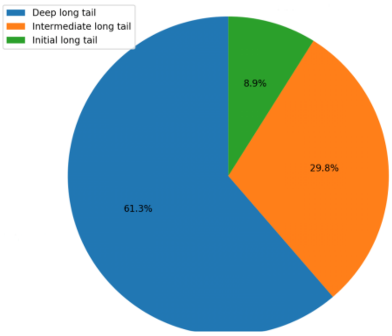


**Figure 7**. Long tail depth of recommended items

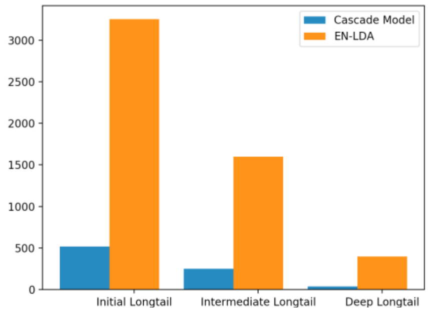


**Figure 8**. Long tail depth between different models

## 4. Experimental Results

In the previous chapter, we present two evaluation indicators of the long tail attributes: long tail recommendation rate (Longtail) and stability (Stability). The experiments in this chapter hope to extend the recommended range to the long tail by the subject model.

Figure 6 shows the comparison of the long tail indicator between the cascade model and the EN-LDA model. In the case of the same data set and other indicators are optimal, the long tail recommendation rate of the EN-LDA model is about 47.6% and the stability is about 60.1% which are both higher than other models. Ensuring that the recommended effect is not greatly affected, EN-LDA model can dig out the value of long tail products as much as possible, proving the feasibility of the subject model for long tail recommendation.

At the same time, we carefully divide the long tail stages of the recommended products, as is shown in Figure 7. It can be seen from the figure that long tail recommended goods are mainly concentrated in the initial long tail stage, accounting for more than 60%; intermediate long tail and deep long tail distribution is about 30% and 10%. While comparing to cascade model in Figure 8, the overall utilization of the cascade model is much smaller than the EN-LDA model, indicating that the EN-LDA model is more comprehensive in considering long tail data.

## 5. Conclusion

Our research is dedicated to solving the long tail problems of traditional recommendation algorithms. In order to ensure the research is based on real scene, we chose the user data of Tmall platform. We first divided the main features into behaviour features, product features and time features through extraction and selection. Then, we showed the long tail problem through experimental comparison of cascade models. In order to further quantify the problem, we proposed a detailed long tail evaluation criteria. And finally we validated the long tail criteria in evaluating the recommended diversity by comparing the performance of the EN-LDA model and the cascade model on this criteria.

On the basis of the traditional classification regression algorithm, the EN-LDA model uses the idea of the theme model to refine the mapping of the LDA algorithm, mapping the breadth of the theme to the long tail data diversity, and tries to solve the long tail problems. From the results of the experiment, it can be seen that the LDA model shows a very strong performance in the long tail evaluation criterion. The indicators have verified the model for long tail data hypersensitivity and high utilization, reflecting that recommendation and long tail phenomenon are not two irreconcilable opposition application scenarios.

## References

[1] Silva E Q D, Camilo-Junior C G, Pascoal L M L, et al. An evolutionary approach for combining results of recommender systems techniques based on collaborative filtering[J]. Expert Systems with Applications, 2016, 53:204-218.
[2] Yin H, Cui B, Li J, et al. Challenging the Long Tail Recommendation[J]. Proceedings of the Vldb Endowment, 2012, 5(9):896-907.
[3] Brynjolfsson E, Hu Y J, Smith M D. From Niches to Riches: Anatomy of the Long Tail[J]. Social Science Electronic Publishing, 2006, 47(4):67-71.
[4] Yao L, Sheng Q Z, Segev A, et al. Recommending Web Services via Combining Collaborative Filtering with Content-Based Features[C]// IEEE, International Conference on Web Services. IEEE, 2013:42-49.
[5] Wei X, Croft W B. LDA-based document models for ad-hoc retrieval[J]. 2006:178-185.
[6] She J, Chen L. TOMOHA: Topic model-based Hashtag recommendation on twitter[C]// International Conference on World Wide Web. ACM, 2014:371-372.
[7] Richardson M, Dominowska E, Ragno R. Predicting clicks:estimating the click-through rate for new ads[C]// International Conference on World Wide Web. ACM, 2007:521-530.
[8] Benesty J, Chen J, Huang Y, et al. Pearson correlation coefficient[M]//Noise reduction in speech processing. Springer Berlin Heidelberg, 2009: 1-4.
[9] Tang X, Yang C C. TUT: a statistical model for detecting trends, topics and user interests in social media[M]. ACM, 2012.
[10] Ma H, Yang H, Lyu M R, et al. SoRec:social recommendation using probabilistic matrix factorization[C]// ACM Conference on Information and Knowledge Management, CIKM 2008, Napa Valley, California, Usa, October. DBLP, 2008:931-940.
[11] Sun C, Gao B, Cao Z, et al. HTM: a topic model for hypertexts[C]// Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2008:514-522.
[12] Lambiotte R, Salnikov V. Effect of memory on the dynamics of random walks on networks[J]. Eprint Arxiv, 2014, 3(2):1373–1378.
[13] Brynjolfsson E, Simester D. Goodbye Pareto Principle, Hello Long Tail: The Effect of Search Costs on the Concentration of Product Sales[J]. Social Science Electronic Publishing, 2011, 57(8):1373-1386.