Comparison of attribute sets in Recommendation Prediction using Tencent Weibo Datasets

CSE 881 Machine Learning

Mini-Project 1

Mengling Hettinger

Abstract

Tencent Weibo is the largest microblog social network in China. The data provided on the KDD cup 2012 contains multiple features from different sources. We clean the data by removing all the users who never accept any recommendations and the users we believe didn't pay attention to their Weibo account during a short period of time. After the preprocessing, we used SVM light algorithm and performed four different approaches with different feature combinations. We found that the user-item keyword frequency summation and the category the item belongs to play important roles on the decision the user made, while the follow action, such as number of followees, number of followers, and number of tweets do not affect it too much. The highest F1 factor we obtain is 47.2% and highest accuracy is 55%.

I. Introduction

Social networking sites provide a tremendous amount of data in recent years. It is desirable to make predictions whether or not a user will follow an item that has been recommended to the user and thus a more efficient recommendation system can be built. In this paper, I use the training data provided by KDD-Cup 2012 and testing data are provided by data mining class.

Comparing with the traditional recommender problem, the datasets provided by Tencent Weibo have some unique features: 1. there are other features about the users and items are provided in other files. There are multiple heterogeneous data besides the recommendation records. 2. most of the decisions are -1, that implies that in most records, user did not accept the recommendation. Whether it is due to the noise or not need to be analyzed.

Here we briefly summarize the datasets we used. The training dataset is made up of 73,209,277 records which each record contains usersID, itemID, decision, timestamps. Item is a specific user in Weibo, which can be a person, an organization, or a group, that was selected and recommended to other users. Other than recommendation records, we also have the access to the

1. User profile data: UserID, year of birth, gender, number of tweet, tag_IDs

2. Item data: ItemID, Item category, Item keyword
3. User action data: UserID, action destination userID, number of at action, number of retweet, number of comment
4. Use sns data: follow history, in the form of follower_userID\followee_userID
5. User keyword data: userID, keywords. Keywords is in the form "kw1:weight1; kw2:weight2;...". Users keyword shares the same vocabulary as the item keword.

In this paper, we present 3 approaches to predict the recommendation using various features extracted from the training data.

II. Pre-processing

The training dataset consists of more than 73 million records spanning 31 days from 1,392,873 users on 4,710 items. Among all the records, there are more than 67 million negative (the user did not accept the recommendation), and 5 million positive (the user accepted the recommendation) records, which means 92.82% of the training data are negative samples. However, negative records can be either real rejections or noise due to various reasons.

In order to get relatively clean data, we firstly remove the users who did not post a tweet, did not have a follower, followee, or did not act during this period of time. Secondly, we remove the records which the users did not accept any recommendations during the whole period. We assume these users do not accept recommendations no matter how similar the items are relative to them. Thirdly, we notice that most of the time an user is given multiple recommendations at a same time, thus we group them into a same session. For user U, if all the recommendations in the same session are being rejected, there is a possibility that they are simply being ignored.

III. Experimental Evaluation

The data provided have multiple features, they can be divided into three categories: user's information, item's information and pair information between user and item.

Since each record in the training set only has a binary result (+1: accept or -1: reject), we view this task as a classification problem. We use y_{ui} to present the actual result that when recommend item I to user u and t_{ui} to present the corresponding predicted value.

Thus the loss function can be calculated in the following form:

$$L_{ui}=\delta(y_{ui}t_{ui}),\ \delta(x)=0(if\ x\geq 0)\vee 1(if\ x<0)$$

In this paper, SVM is used to build the basic model. The features are varied for specific informational utilization . We discuss each approach in detail in the following way:

1. Based on number of follower/followee/tweets

if one user has more followers, followees and number of tweets, it means that the user is more likely to be active when using Weibo. Thus we assume these features may affact the decision.

2. Keywords

if user u share the same keyword with item i, this keyword can be seen as a feature of the common interests between u and I. We also have the information about the frequency of each keyword in user_keyword file. As it has been renormalized, we may define

$$\sum_{kw}(W(u,kw))user_{kw} \wedge item_{kw}$$

as the measurement of how close the user's keyword relative to the item's keyword.

3. age and gender

The age and gender of a user are also the important factors of his/her interest. We split age into 7 bins with binsize 8 years old where 5 < bin1 < 13, 13 < bin2 < 2 1, 22 < bin3 < 30, 31 < bin4 < 39, 40 < bin5 < 38, 39 < bin6 < 47, bin7 >48, and we also divide the gender into female and male for each bin.

4. Item's category

The category of an item is in the form of a.b.c.d where d includes in c includes in b includes in a. This feature may affect users decision in the following ways: a. some categories may be more popular than others for a given time, for example, if some pop star just released a new album, then he/she may be popular at this period of time. b. certain users may be interested in certain type of items. For example, teenages are more likely to follow a music star than old people.

We tried four methods with different combinations of all the features using SVM light algorithm. The learning curve is set to 7 different sizes of training data : 100, 1000, 2000, 5000, 10000, 100000, 1000000.

Method 1 includes features: a. number of followee, b. number of follower, c. number of tweets, d. user_item pair summation of the keyword frequency.

Method 2 includes features: a. age range, b. gender, c.user_item pair summation of the keyword frequency, d. the category of the item.

Method 3 includes features: a. age range, b. gender, c. number of tweets d. category of the item.

Method 4 contains all the features we obtained: a. a. number of followee, b. number of follower, c. number of tweets, d. age range, e. gender, f. user_item pair summation of the keyword frequency, g. item category.

	Accuracy	Precision	Recall
Method 1	26.90%	26.90%	100%
Method 2	54.96%	37.56%	62.60%
Method 3	26.96%	26.96%	100%
Method 4	48.48%	26.32%	52.67%

F1 score = 2P*R/(P+R):

Method 1 42.4%

Method 2 47.2%

Method 3 42.4%

Method 4 35.1%

Since the data have the asymmetric nature, we use F1 score here to illustrate how well each method performs. From the above result, we believe that method 2 is the best so far and that means the frequency of the keyword pair and the category of the item is more important while the number of follower, number of followee and number of tweets have less impact on whether the user is going to accept or reject the recommended item.

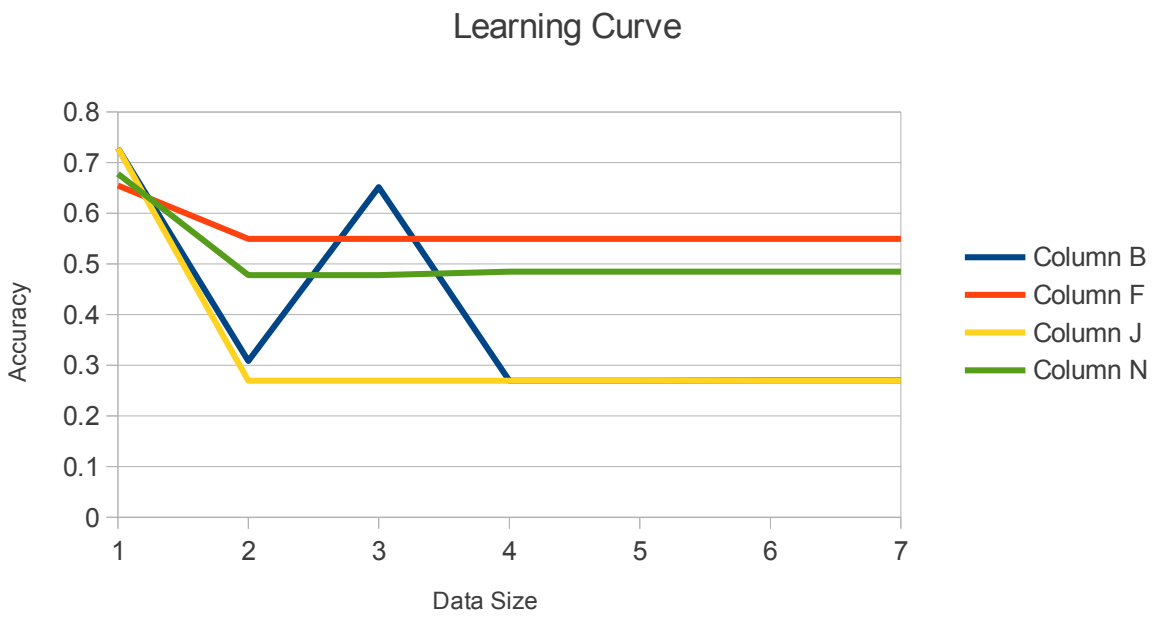

Fig 1. Learning curve: accuracy vs data size

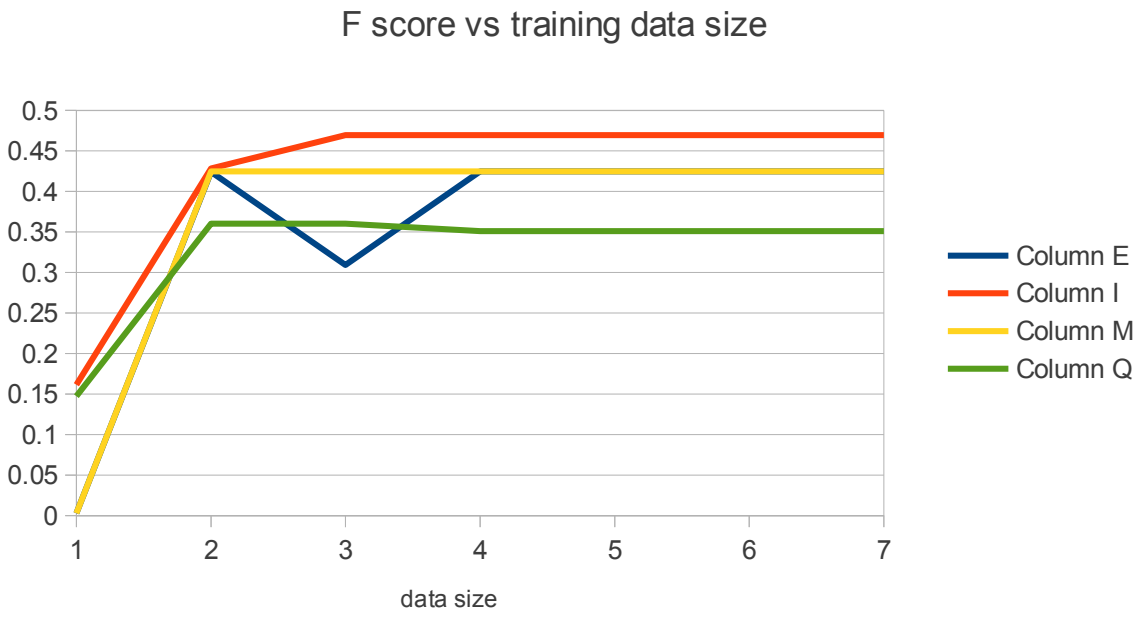

Fig 2: Learning curve: F score vs training data size

We also change the training data size and plot the learning curve (Fig 1 and Fig 2). Most of the methods converge very fast. The blue line is method 1, orange is method 2, yellow is method 3 and green is method 4. From Fig 2, it can be easily seen that method 2 is the best approach among all four.

IV. Conclusion

We tried four different combinations of features that extract from different datasets frm Tencent Weibo. The highest F1 score is 47.2% and the highest accuracy is 55% from method 2. We believe, from our experiments, the most important features are:1.whether there are common keywords between the user and item and how much is the weight; 2. the category of an item also shows the importance.