Effectiveness of the Data Generated on Different Time in Latent Factor Model

Qianru Zheng
Department of Computer Science
City University of Hong Kong
qrzheng2-c@my.cityu.edu.hk

Horace H S IP
Department of Computer Science
City University of Hong Kong
cship@cityu.edu.hk

ABSTRACT

User selection data accumulates as time goes by. Although the recent selections are usually assumed to have higher impact on the recommendation accuracy, empirical studies on this problem are limited. For old data, whether they can contribute to the recommendation accuracy is still to be determined. On one hand, changes in short-term user preference over time may limit their effectiveness in prediction, but on the other hand, one cannot rule out their potential in capturing long term user preferences. The result is important for the system owner to determine which data is useful to make the recommendation accurately. While there have been some related studies on the time dependency of data quality using neighbor-based CF methods (e.g., [4]), its effects remain unverified for other CF methods. In this paper, we study the effect of data generated over different time period on recommendation precision using several popular model-based CF algorithms (latent factor models). Experiment results show that while more recent data expectedly have larger impacts, the usefulness of older data cannot be ignored as long as there are sufficient old samples. However, the addition of insufficient amount of old data seems to have negative impacts.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: [Miscellaneous]

Keywords

Data Effectiveness, Latent Factor Model, Recommendation system

1. INTRODUCTION

Recommendation systems generally rely on previous user selections. Intuitively, the more user data is accumulated over time, the higher is the recommendation accuracy. In practice, however, the problem is not as trivial, particularly in cases where less recent data are involved. On one hand, it is reasonable to speculate that inclusion of older data would be ineffective, or would even have negative impacts on prediction of short-term preferences based selections. On the other hand, however, they nevertheless still contribute to the overall improvement in data density, thus should have positive impacts on the learning of long term preference. The overall picture is far from clear, and up to now there are few empirical studies that support or refute either claims.

One related study on the impact of such timeliness of data (and the effect of inclusion of old data in particular) is the work of Pessemier et al. [4]. Their work studied the impact of inclusion of older data on recommendation accuracy in neighbor-based CF algorithms. It was shown that, for certain dataset, old data is indeed useful for improving the resulting accuracy. However, it is still unknown whether similar results will hold if other CF algorithms, such as the model-based approaches, are in place. Moreover, the effectiveness of inclusion of recent user data, although generally believed to be helpful, still needs to be verified. The results would be important for determining the data that are effective in learning user preference, so that the overall recommendation accuracy can be improved. Cremonesi et al.[3] have conducted a research on the effect of data evolution on the accuracy, which concerns how the accuracy changes as more recent data is added. Our work is different from theirs: effectiveness of recent and old data are both considered.

In this paper, we study the effect of inclusion of both old and recent data in model-based collaborative filtering methods. More specifically, we focus on three different latent factor models because of their good performance and that they have recently become more prevalent in recommendation system [2, 7].

The remainder of this paper is organized as follows: Section 2 introduces the latent factor models. Section 3 describes our testing methodology. Section 4 shows the results obtained by three different latent factor models on two representative data sets. Finally, conclusion and discussion of the future work are given in Section 5.

2. LATENT FACTOR MODELS

Recently, latent factor models have gained popularity in recommender systems because of their effectiveness in the recommendation accuracy [2]. The majority of latent factor models are based on the factorization of the user-item rating matrix by Singular Value Decomposition (SVD) [7].

The main idea of SVD models is to factorize the user-item rating matrix into three low rank matrices. The idea is illustrated in Eq.1, where U is $n \times k$ orthonormal matrix, Q

RecSys'13, October 12–16, 2013, Hong Kong, China.

ACM 978-1-4503-2409-0/13/10 ...$15.00.
http://dx.doi.org/10.1145/2507157.2507202 .

is $m \times k$ orthonormal matrix and Σ is $k \times k$ diagonal matrix with the top k singular values. k is the number of latent factors. Alternatively, $\hat{R}$ is represented by $P \cdot Q^T$, see Eq.2. $\hat{R}$ is the estimated rating matrix.

$$\hat{R} = U \cdot \Sigma \cdot Q^T \tag{1}$$

$$\hat{R} = P \cdot Q^T \tag{2}$$

After factorization, each user is associated with a k-d vectors p_u, which represents the user u's preference for k factors. And each item is also associated with a k-d vectors q_i, which describes the item i's importance weight for k factors. In this work, we study three common SVD models, namely, Pure SVD, SVD (bias) and SVDpp [7]. The number of latent factors (k) is 50 for all cases. We first briefly explain the three models as follows. Pure SVD is the basic latent factor models. It measures the association between i (the item) and u (the user) by using the product of user-factor vector p_u and item-factor vector q_i. (Eq.3).

$$\hat{r}_{ui} = p_u \cdot {q_i}^T \tag{3}$$

In the SVD (bias) model, in order to predict the relevance of an item for a given user, a rating bias is also considered. The idea is shown in Eq.4, where b_{ui} is the bias. Refer to [7] for a discussion on the estimation of b_{ui}.

$$\hat{r}_{ui} = b_{ui} + p_u \cdot {q_i}^T \tag{4}$$

SVDpp (also noted as SVD++) is an extension of SVD (bias) by also considering the implicit feedback. The concept is shown in Eq.5, where $R(u)$ denotes the set of items rated by user u and $\sum_{j \in R(u)} y_j$ represents the implicit preference of user u. Interested readers may refer to [7] for more details.

$$\hat{r}_{ui} = b_{ui} + (p_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j) \cdot {q_i}^T \tag{5}$$

3. METHODOLOGY

In this paper, we adopt a testing methodology similar to the one adopted in [4]. For each dataset, all user historical selections are chronologically split into a training set and a testing set, with the most recent ones (10%) in testing set while the remaining (90%) as input data (Input data is not equivalent to the training set, as explained later on). To study the usefulness of both old data and recent data, one straightforward approach would be to first divide the data into an old data group and a recent data group, and then compute their recommendation accuracy accordingly. However, the drawback is that there is no objective rule for dividing the data this way. So instead, we divide the input data into 10 chronologically ordered segments, namely $\{D_1, D_2, \ldots, D_{10}\}$. Each segment contains 10% of the total data with D_1 being the oldest and D_{10} being the most recent. Initially, all segments are put into the training set, which is used as the benchmark for comparison. In each subsequent iteration, a new training set is obtained by depleting different number of segments. Recommendation accuracy based on each generated training set is then compared. This way, we can observe the changes in recommendation accuracy and determine which data segment are useful for prediction. The experiment proceeded as follows. To learn the usefulness of recent data, in the rth iteration, the r most recent segments, namely $\{D_{10-r+1}, \ldots, D_{10}\}$, are removed while the remaining segments $\{D_1, \ldots, D_{10-r}\}$ are used as training set. The experiment is then repeated for learning the usefulness of old data. This time, instead of removing the r most recent segments, the r oldest segments are removed in the rth iteration. Regarding evaluation metrics, Root Mean Square Error (RMSE) is a widely used metrics in recommender systems, which measures the difference between predicted rating and the true rating for a given pair of user and item. However, RMSE is not applicable in situations where only 'best bet' items are provided to the user while the predicted ratings are not provided (this is known as the find good items task [5]). So instead, we require a classification metric that measures how frequent the system makes correct or incorrect suggestions [5]. For this purpose, the primary classification metric top n precision is adopted in this work for recommendation accuracy evaluation.

Table 1: Basic statistical information of two employed data sets

	# of users	# of items	# of ratings	Density
Movielens	2,113	10,196	800k	3.976%
Netflix	1,087	1,948	215k	10.18%

3.1 Data Sets

For the dataset selection, we require datasets that are collected over a long period because such datasets contains plenty information for learning the effectiveness of data generated over time (i.e., with both recent data and the data collected long time ago). Two widely used datasets fit this requirements, namely Movielens [1] and Netflix [6]. The Movielens dataset [1] is logged from October 1997 to December 2008. The original Netflix dataset contains more than 480k users and 17K items. For the sake of scalability, a subset of the Netflix dataset is used instead, which was collected over a 2 years period from August 2002 to December 2005. Table 1 shows the basic statistical information of both datasets. The time window of each data segment in Netflix and Movielens is about 2.5 months and 12 months respectively. That is, the most recent data segment in Movielens and Netflix were collected in the last 12 months and last 2.5 months respectively, whereas the oldest data segment in Movielens was collected 10 years before and that of Netflix was collected 2.5 years before.

4. EXPERIMENTAL RESULTS

4.1 Effectiveness of the recent data

We first performed a series of experiments to verify the usefulness of recent data by removing the data segments from the training set starting from the most recent one (as described in Section 2). For instance, in the second iteration, the most recent 10% data (D_{10}) is removed, and in the following iteration, the most recent 20% data ($\{D_9, D_{10}\}$) is removed, and so on. Figure 1 shows the changes in precisions of the three latent factor models as the training data is being reduced in the successive iterations.

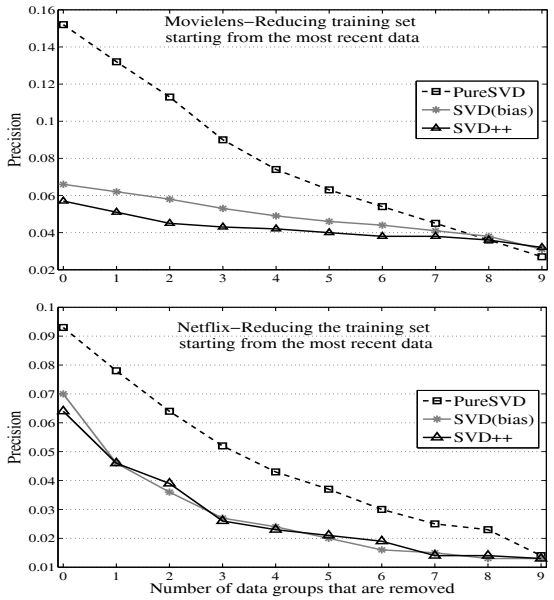

Figure 1: Change of the top n precision by removing the data starting from the most recent one.

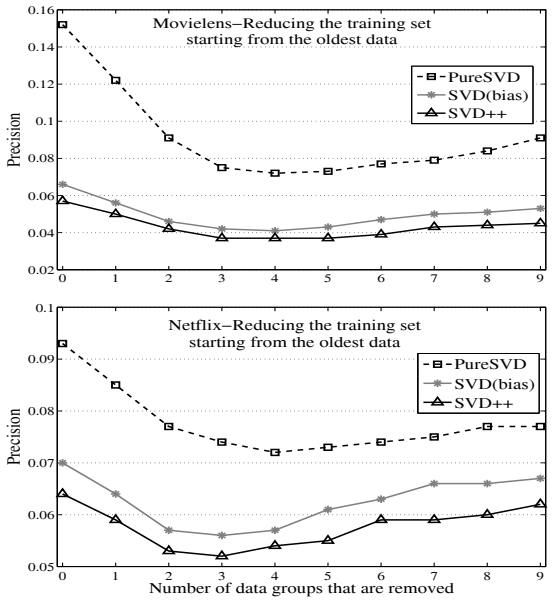

Figure 2: Change of the top n precision by removing the data starting from the oldest one.

As shown in Figure 1, we can observe that in both Movielens and Netflix datasets, the precision of Pure SVD, SVD (bias) and SVDpp decreases linearly as more data (starting from the most recent ones) is eliminated from the training set. The loss of precision may due to that the reduced amount of data leads to lower density of the user-item rating matrix, as a result the recommendation accuracy decreased. For instance, in the first 4 iterations, the precision has decreased no less than 50% after the most recent 40% of the data is removed. For example, in Netflix dataset, precision of PureSVD decreases 53.5% after the most recent 40% of the data is removed from the training set. These results confirm the assumption that the recent data contributes the majority of the recommendation accuracy.

4.2 Effectiveness of the old data

Similarly, another series of experiments were performed to study the effectiveness of the old data by removing the data segments from the training set starting from the oldest one. Experiment results using the three selected latent factor models on the two datasets are shown in Figure 2.

Figure 2 illustrates the decrease in precision of 3 latent factor models when the oldest data segment is removed from the training set. This result confirms the usefulness of oldest data in the recommendation accuracy. But after the oldest 30% of the data are depleted, precision of 3 latent factors model begin to grow. For example, in Movielens dataset, precision of SVD is getting raised after 30% or more percentage of the oldest data is removed. Although the precision increases, it is not as high as the baseline situation. This is an interesting phenomenon that the accuracy increases even though the density of the user-item rating matrix decreases. Compared to Figure 1, precisions decreased linearly without fluctuation. However, in Figure 2, obvious fluctuation can be observed. We try to explain this by two possibilities. One possibility is that some proportion of data is useless to contribute the recommendation accuracy. Hence, by removing this part of data, accuracy gets increased. The second possibility is that some proportion of data is ineffective to improve the accuracy with the absence of other older data. Hence, after this proportion of data is removed, the accuracy begins to increase.

4.3 Discussion

To distinguish the above discussed two possibilities, the effectiveness of the data generated on different time should be investigated respectively to check if there is some proportion of data which is useless for the recommendation accuracy. If such data exists, recommendation accuracy based on it should be zero. Since training set is getting reduced by a data segment, the usefulness of each segment should be verified individually. To fulfill this purpose, recommendations are generated based on each data segment and precisions of different set of recommendations are compared. The results are presented in Figure 3, with the oldest data segment on the left and the most recent one on the right.

From Figure 3, we can see that, data segment generated on different time all contributes to the recommendation accuracy, although the most recent data contributes more on it. Based on this observation, we conclude that each part of data contributes to the recommendation accuracy in latent factor models. Hence, the first possibility which assumes that some proportion of the data is useless to improve the recommendation accuracy is not true. In order to explain the fluctuation of precision in Figure 2, we come to think about the second possibility which relates to the constitution of the training set. Different constitution of data may influence the recommendation accuracy. For example, only the most recent data but without enough older data may lead to inaccurate recommendations.

To further explain this, we introduce the theory of user long-term and short-term preference in [8]. Xiang et al. [8] argued that user selection behavior was affected by long-term and short-term preference. Long-term preference is

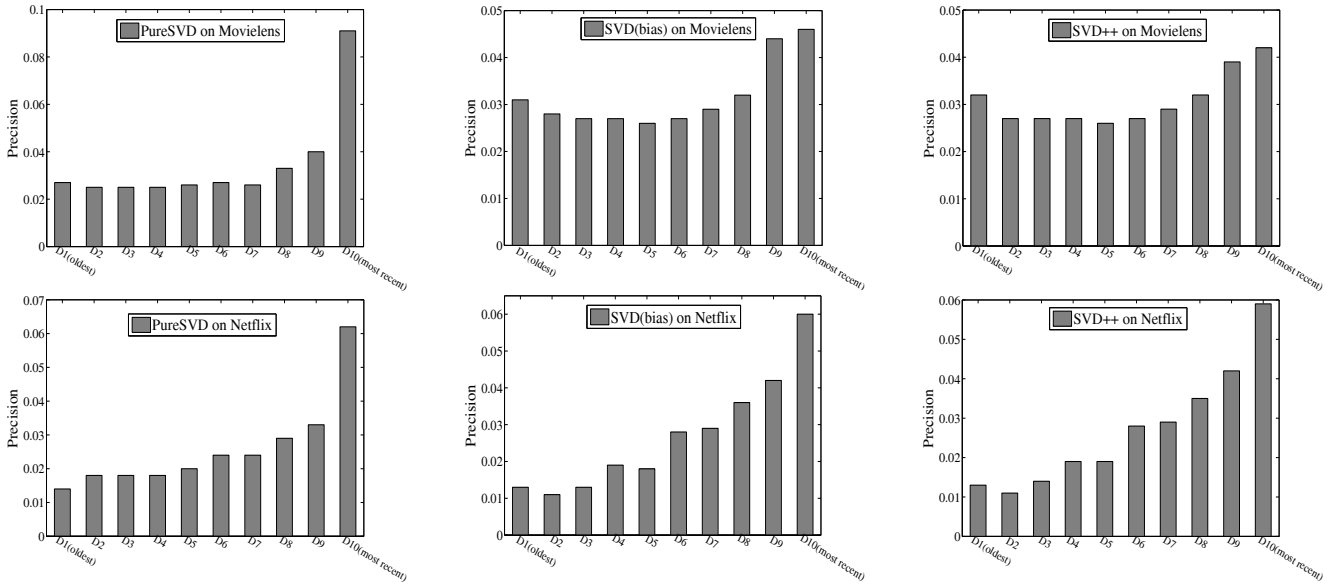

Figure 3: Effectiveness of the data generated on different time in three latent factor models on two datasets

inherent to determine the user behavior all the time, while the short-term preference is triggered by transient events, such as new product releases. Therefore, in order to make recommendation accurately, it is important to study both long-term and short-term preference correctly. According to the characteristics of long-term and short-term preference, short-term preference should be learned from the data generated within a specific time period, while the long-term preference should be learned thoroughly across all the data. Moreover, the long-term preference is time dependent, and hence should be learned starting from the oldest data to the most recent one. If the long-term preference is learned starting from the oldest data, it will be well learned after more selections (recent ones) are put in, as a result recommendation accuracy increased. Results in Figure 1 can illustrate this conclusion. By viewing the results in Figure 1 from right to left (which is equivalent to observe the variation of the precision as the size of the training set gets larger), we can see that recommendation accuracy increased linearly as more data (recent ones) is added in. In contrast, if long-term preference is learned starting from the most recent data but without enough old data, it may be learned incorrectly, as a result, decreasing the recommendation accuracy. Figure 2 can illustrate this conclusion by viewing the results from right to left. In the first several iterations, the recommendation accuracy decreases, mainly due to that based on a small proportion of the most recent data but lack of the older data, estimation of long-term preference is incorrect. In the subsequent iterations, as more data (older ones) is added for training, the long-term preference is well learned, hence the recommendation accuracy increases.

5. CONCLUSIONS

In this paper, we study the usefulness of the recent data and old data in three latent factor models in terms of recommendation accuracy. The experimental results show that data generated on different time all contributes to the recommendation accuracy, although the most recent data has larger contribution on it. Nevertheless, we find that old data is important to learn the user long-term preference. With insufficient old data, user long-term preference will be learned inaccurately, which may reduce the recommendation accuracy. These results are useful for the system owner: in order to make the recommendation accurately, all data should be put into training. Moreover, since the recent data is more effective, system owners can pay more attention on it, such as giving a higher weight to it. In this work, we only verify the usefulness of the old data and recent data in three latent factor models on two datasets. In the future, we hope to verify our conclusion with more datasets and other recommendation algorithms.

6. REFERENCES

[1] I. Cantador, P. Brusilovsky, and T. Kuflik. RecSys, 2011.

[2] P. Cremonesi, Y. Koren, and R. Turrin. Performance of recommender algorithms on top-n recommendation tasks. RecSys, pages 39–46, 2010.

[3] P. Cremonesi and R. Turrin. Time-evolution of iptv recommender systems. EuroITV, pages 105–114, 2010.

[4] T. De Pessemier, S. Dooms, T. Deryckere, and L. Martens. Time dependency of data quality for collaborative filtering algorithms. RecSys, pages 281–284, 2010.

[5] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, 2004.

[6] J.Bennett and S. Lanning. The netflix prize. In *Proceedings of KDD Cup and Workshop*, 2007.

[7] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[8] L. Xiang, Q. Yuan, S. Zhao, L. Chen, X. Zhang, Q. Yang, and J. Sun. Temporal recommendation on graphs via long- and short-term preference fusion. KDD, pages 723–732, 2010.