Evaluating Collaborative Filtering Over Time

Neal Lathia
Dept. of Computer Science
University College London
London, WC1E 6BT, UK
n.lathia@cs.ucl.ac.uk

Stephen Hailes
Dept. of Computer Science
University College London
London, WC1E 6BT, UK
s.hailes@cs.ucl.ac.uk

Licia Capra
Dept. of Computer Science
University College London
London, WC1E 6BT, UK
l.capra@cs.ucl.ac.uk

ABSTRACT

Collaborative Filtering (CF) evaluation centres on accuracy: researchers validate improvements over state of the art algorithms by showing that they reduce the mean error on predicted ratings. However, this evaluation method fails to reflect the reality of deployed recommender systems, which operate algorithms that have to be iteratively updated as new users join the system and more ratings are input. In this work we outline a method for evaluating CF over time, and elaborate on work done exploring the temporal qualities of CF algorithms and recommendations.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: :Information Filtering

General Terms

Algorithms

Keywords

Temporal Collaborative Filtering, Time-Averaged Error

1. INTRODUCTION

Collaborative filtering (CF) [1] fuels the success of online recommender systems; in fact, the benefits of filtering information collaboratively are so compelling that facets of CF are now making their way into search engines [2]. The crux of CF algorithm evaluation has become accuracy [3]: a plethora of research in this field focuses on methods that reduce the error between the *predictions* an algorithm makes of user-ratings and the ratings themselves. In other words, to measure the performance of a CF algorithm, a user-rating dataset is split into training/test sets and error is measured on test set predictions after the algorithm has been fed the training ratings. Improvements are then measured by repeating this process, with the same data and modified algorithms. This methodology is reflected in the ongoing Netflix prize[1]. The use of accuracy in and of itself has been questioned before [4]; however, more importantly, the *method* used to test CF algorithms fails to address an important aspect of recommender systems: time. Deployed recommender systems will be iteratively updated as users input ratings in order to update the recommendations that each user is offered [5]. The underlying rating dataset will grow, and any summary statistics derived from it will be subject to change. Experiments on an unchanging dataset do not reflect the reality of a deployed recommender system, and the effect that users will experience as a result of updated recommendations cannot be explored with any static method.

In this paper, we outline a method for evaluating collaborative filtering over time (Section 2), and elaborate on two aspects of CF: how user-similarity changes with time (Section 2.1) and how the system's time-averaged accuracy fluctuates (Section 2.2). We then argue that a broader range of characteristics of recommendations (beyond mere accuracy) are yet to be investigated, and briefly summarise our current work in this area.

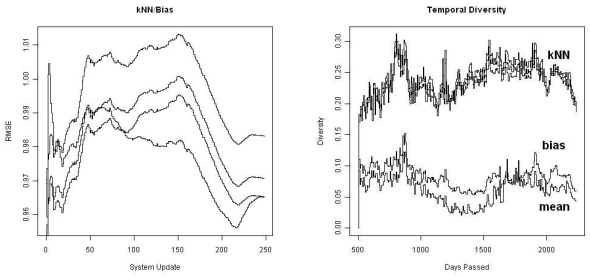
2. TEMPORAL CF

In order to incorporate time into CF experiments, we sort user ratings according to when they were input and then simulate a system that is iteratively updated (every μ days). Beginning at time ($t = \epsilon$), we use all ratings input before ϵ to train the algorithm and test on all ratings input before the next update, at time ($\epsilon + \mu$). We then repeat this process for each time t, incrementing by μ at each step. At each step, what was previously tested on becomes incorporated into the training set; we thus mimick the actual operation of deployed recommender systems by augmenting training sets with ratings in the order that users input them and only testing on ratings that users will make before the next round of recommendation updates. Altering CF experiments in this way highlights a number of hidden characteristics of recommender systems: in the next sections, we briefly summarise some key findings observed to date.

2.1 Similarity Over Time

The basic assumption of CF is that users who have been like-minded in the past are likely to be like-minded in the future. This assumption leads to the intuitive use of the k-Nearest Neighbour (kNN) algorithm for CF [1]: given a user (or item), the ratings of similar users (items) can be used to predict the former's ratings. The focus thus shifts toward

[1] http://www.netflixprize.com



(a) TA-RMSE (b) Temporal Diversity

Figure 1: Time-Averaged RMSE Of a Series of CF Algorithms and Temporal Top-N Diversity of Netflix Data

the problem of finding *like-minded* neighbours, by measuring the similarity between users or items. In this context, a range of similarity measures have been adopted, including the Pearson Correlation, Cosine Similarity, and many others. However, once similarity is examined on the temporal scale, there is no guarantee that users who were measurably similar at a previous update will continue to be deemed similar. In [6], we found that the similarity between pairs of users (and thus likelihood that they repeatedly be each other's kNN neighbours) highly fluctuates over time, and depends more on *how* similarity is being measured rather than *what* the users are rating. In other words, CF algorithms do not necessarily reflect their founding assumption in the way that they manipulate data over time.

2.2 Accuracy Over Time

To measure the temporal accuracy of a CF system that is iteratively updated, we applied the time-averaged root mean squared error (TA-RMSE) metric. If we define R_t as the set of predictions made up to time t, then the time-averaged error is simply the RMSE achieved between the predictions $\hat{r}_{u,i}$ and ratings $r_{u,i}$ made so far:

$$\text{TA-RMSE}_t = \sqrt{\frac{\sum^{N}_{\hat{r}_{u,i} \in R_t} (\hat{r}_{u,i} - r_{u,i})^2}{|R_t|}} \quad (1)$$

Figure 1(a) shows the TA-RMSE results of the kNN algorithm (with a variety of k values) and Potter's bias model [7] over a sequence of updates on Netflix data subsets. The results highlight that there is no single algorithm that dominates over all others over time. In fact, in [8] we explored how techniques that *improve* accuracy in static experiments actually *degrade* time-averaged accuracy during iterative experiments; furthermore, techniques that produce the best results at the *global* level do not produce similar results when analysing the per-user performance.

2.3 Temporal Recommendations

Observing how CF operates over a sequence of updates also paves the way for exploring a broader range of recommendation characteristics. Given a method of evaluating CF over time, let us focus on the metrics. Since minimal improvements to accuracy bear little meaning to the end user [4], other metrics are worth considering, like temporal diversity. While diversity has been explored in the static case [9], one may be interested in measuring the extent that users are recommended the same items repeatedly over time [10]. To explore this facet of recommendations, we defined the *diversity* between two top-N lists, $L_{u,a}$ and $L_{u,b}$, generated for user u at times a and b, by looking at the proportion of items that appear in both lists, using the Jaccard distance:

$$div(L_{u,a}, L_{u,b}) = 1 - \frac{|L_{u,a} \cap L_{u,b}|}{|L_{u,a} \cup L_{u,b}|} \quad (2)$$

Figure 1(b) plots the temporal diversity in the recommendation rankings when three different algorithms are applied to the Netflix data. From these, we observe that of the methods explored, those that are more *accurate* produce lower *diversity* over time: researchers must therefore question what characteristics they aim to achieve with their recommendations, and prioritise accordingly.

3. CONCLUSIONS

In this paper we have outlined a method for evaluating CF over time, and introduced a number of metrics that relate to temporal evaluations: the time-averaged RMSE measures how prediction accuracy varies over time, while the temporal diversity metric measures the extent that users are being recommended the same items over a number of updates. A number of further metrics are possible. For example, researchers may be interested in the *novelty* of recommendations: how quickly items are recommended after being first rated. More generally, evaluating any information system requires a notion of what *good* results are: in this work, we argue that an awareness of the temporal nature of recommender systems not only better reflects how CF algorithms are deployed online, but broadens the set of qualities that can be explored when examining the dynamics of recommendations.

4. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE TKDE*, 17(6), June 2005.

[2] J. Pujol, R. Sanguesa, and J. Bermudez. Porqpine: A Distributed And Collaborative Search Engine. In *Proc. 12th WWW*, Budapest, Hungaryt, 2003.

[3] J. Herlocker, J. Konstan, L. Terveen, and J. Riedl. Evaluating Collaborative Filtering Recommender Systems. *ACM TOIS*, 22(1):5–53, 2004.

[4] S. M. McNee, J. Riedl, and J. A. Konstan. Being Accurate Is Not Enough: How Accuracy Metrics Have Hurt Recommender Systems. In *Extended Abstracts of the ACM CHI Conference*, Montreal, Canada, 2006.

[5] M. Mull. Characteristics of High-Volume Recommender Systems. In *Proceedings of Recommenders '06*, Bilbao, Spain, September 2006.

[6] N. Lathia, S. Hailes, and L. Capra. kNN CF: A Temporal Social Network. In *Proceedings of Recommender Systems (ACM RecSys '08)*, Lausanne, Switzerland, 2008.

[7] G. Potter. Putting the Collaborator Back Into Collaborative Filtering. In *Proceedings of the* 2^{nd} *Netflix-KDD Workshop*, Las Vegas, USA, August 2008.

[8] N. Lathia, S. Hailes, and L. Capra. Temporal Collaborative Filtering With Adaptive Neighbourhoods. In *Proceedings of ACM SIGIR*, Boston, Massachusetts, 2009.

[9] J. A. Konstan G. Lausen C.N. Ziegler, S. M. McNee. Improving Recommendation Lists Through Topic Diversification. In *Proceedings of WWW 2005*, Chiba, Japan, May 2005.

[10] N. Lathia, S. Hailes, and L. Capra. Tuning Temporal Recommendations With Adaptive Neighbourhoods. In *Under Submission*, June 2009.