RC24371 (W0710-026) October 5, 2007
Computer Science

IBM Research Report

Active Collaborative Prediction with Maximum Margin Matrix Factorization

Irina Rish, Gerald Tesauro
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Active Collaborative Prediction with Maximum Margin Matrix Factorization

Irina Rish
IBM T.J. Watson Research Center
19 Skyline Drive
Hawthorne, NY 10532
rish@us.ibm.com

Gerald Tesauro
IBM T.J. Watson Research Center
19 Skyline Drive
Hawthorne, NY 10532
gtesauro@us.ibm.com

Abstract

Collaborative prediction (CP) is a problem of predicting unobserved entries in sparsely observed matrices, e.g. product ratings by different users in online recommender systems. However, the quality of prediction may be quite sensitive to the choice of available samples, which motivates active sampling approaches. In this paper, we suggest an active sampling method based on the recently proposed Maximum-Margin Matrix Factorization (MMMF) [7], a linear factor model that was shown to outperform state-of-art collaborative prediction techniques. MMMF is formulated as a semi-definite program (SDP) that finds a low-norm (rather than traditional low-rank) matrix factorization, and is also closely related to learning max-margin linear discriminants (SVMs). This relation to SVMs inspires several margin-based active sampling heuristics that augment MMMF and demonstrate promising results in a variety of practical domains, including both traditional recommender systems and novel systems-management applications such as predicting latency and bandwidth in computer networks.

1 Introduction

Given a large but sparsely sampled matrix, the *collaborative prediction (CP)* problem is to predict the unobserved entries from the observed samples, assuming the entries are dependent. Typical application include online recommendation systems that attempt to predict user's preferences towards different products (e.g., movies, books), based on previously obtained product ratings from different users. Collaborative prediction can be also applied to non-traditional domains such as distributed systems management applications considered in this paper. In such applications, we wish to predict the end-to-end performance, such as latency in computer networks or bandwidth in peer-to-peer content-distribution systems, based on a limited number of available measurements between pairs of nodes. Moreover, collaborative prediction tasks may arise in various other domains, e.g. in image processing, where we may want to reconstruct unobserved (occluded) parts of an image from the observed pieces.

A typical assumption that leads to various collaborative prediction techniques is a *factorial* model that assumes the presence of some hidden factors that affect user's preferences towards the products. For example, genre of a movie, its comic factor, and its violence factors may affect user's preferences. Similarly, two nodes that are located in same part of the network may share several "hidden factors" such as intermediate nodes on their path to a third node; moreover, even distant nodes can share some other hidden factors which determine a quality of service they provide: e.g., a high-bandwidth can be achieved by downloading from a remote but powerful server instead of local laptop with a wireless connection. In this paper, we will focus on *linear factor models* which result into a matrix-factorization approach to collaborative prediction.

The predictive accuracy of such models can improve dramatically when more samples become available; however, sampling can be costly: a user may become annoyed if she is asked to rate many products or a network may become congested if too many measurements are performed. Besides, suggesting a product to buy or a server to download from has a high cost if the user does not like the product, or the download bandwidth turns out to be low. Therefore, a cost-efficient active sampling becomes an important component of any successful collaborative prediction approach.

In this paper, we propose an active-learning extension of the recently proposed Maximum Margin Matrix Factorization (MMMF) approach to collaborative prediction that was shown to outperform state-of-art collaborative prediction methods and has some nice theoretical guarantees [7, 6]. MMMF is a matrix factorization approach formulated as a convex optimization problem that uses low-norm constraints, unlike previous non-convex approaches, such as low-rank (SVD-like) or non-negative matrix factorizations [4]. Besides, MMMF is closely related to maximum-margin linear discriminants (SVMs), i.e. it can be viewed as simultaneous learning of multiple SVMs and a set of features common to to all SVMs. This insight is directly exploited by our active learning approach that extends MMMF with margin-based active-learning heuristics, where the margin is used to estimate informativeness of a candidate sample, as suggested in [8]. Besides the straightforward "most-uncertain" (min-margin) sample selection, we also investigate alternatives that take into account the cost of sampling.

Previous work on active sampling for collaborative filtering includes a *value-of-information* approach of [1] and Bayesian model averaging method of [3]. Both approaches are based on probabilistic hidden-factor models and computationally expensive procedures for choosing next active sample that require minimization of expected cost (or uncertainty). On the contrary, our active sampling is quite simple and inexpensive as it only compares the margin values produced by MMMF. Another related work proposes an active-sampling method for low-rank matrix factorizations [2] that requires a small number of users to provide the ratings of ALL products – a clearly unrealistic assumption in any large enough, practical recommendation system. Although our heuristic active sampling lacks theoretical guarantees associated with the above approach, it is much more practical since it does not impose any unrealistic sampling assumptions. Empirical evaluation on several application domains, from recommender systems to computer networks and peer-to-peer files distribution systems, demonstrate the advantages of our active sampling methods.

In summary, this paper makes following contributions. It proposes a simple, computationally efficient active sampling extension of the state-of-art MMMMF method for collaborative prediction, compares several active-sampling strategies, both on traditional collaborative filtering domain (movie rating prediction) and on novel application domain – distributed computer systems management, and demonstrates a noticeable improvement in prediction accuracy over random sampling.

2 Collaborative Prediction as Matrix Factorization

Collaborative prediction problem can be stated as follows. Given a partially observed $n \times m$ matrix Y, let us find a matrix X of the same size that provides "best" approximation for unobserved entries of Y with respect to a particular *loss function*, such as sum-squared loss for real-valued matrices, 0/1 loss or its surrogates such as hinge loss for binary and ordinal matrices, and so on.

Linear factor models, a particular type of factor models for collaborative prediction, assume that each factor is a preference vector, and actual user's preferences correspond to a weighted linear combination of these factor vectors with user-specific weights. Let k be the number of such factors, then the matrix Y can be approximated by a *matrix factorization* $X = UV$, where U is a $n \times k$ *coefficient matrix* (where each row represents the extent to which each factor is used) and V is a $k \times m$ *factor matrix* where the rows represent the "expression level" of the factors in each of m "products". Since the rank of the approximation matrix X is clearly bounded by k, fixing k to some small value leads to a low-rank matrix factorization approaches.

For example, a standard matrix-factorization approach is singular value decomposition (SVD) which finds a low-rank approximation that minimizes the sum-squared distance between X and a *fully observed* Y. The problem is, when Y is not fully observed, as in collaborative prediction and particularly in end-to-end performance prediction, SVD is not directly applicable and finding a low-rank approximation to a partially observed function using a sum-squared loss becomes a difficult

non-convex optimization problem, for which no exact solution method is known. Also, even for completely known matrix Y, approximating it with respect to other losses that the sum-squared loss (e.g., expected classification error) is still a non-convex optimization problem with multiple local minima [7].

In order to overcome such limitations, a novel *Maximum Margin Matrix Factorization (MMMF)* approach was proposed by [7]. This approach replaces the bounded-rank with the *bounded norm* constraint on U and V and yields a convex optimization problem. Namely, Lemma 1 in [7] shows that finding the matrices U and V having low Frobenius norms $\|U\|_{Fro}$ and $\|V\|_{Fro}$ is equivalent to minimizing the *trace-norm* (the sum of singular values) $\|X\|_{\sum}$ of X, since

$$\|X\|_{\sum} = \min_{X=UV} \|U\|_{Fro}\|V\|_{Fro} = \min_{X=UV} \frac{1}{2}(\|U\|_{Fro}^2 + \|V\|_{Fro}^2) \tag{1}$$

Since the trace-norm is a convex function [7], minimizing it together with any convex loss function or constraint results into a convex problem.

For simplicity, we focus herein on binary-valued matrices $Y \in \{-1, 1\}^{n \times m}$, and thus use the MMMF with hinge-loss, as in max-margin linear discriminant (SVM) learning. The MMMF optimization problem can be then stated as:

$$\min_X \|X\|_{\sum} + c \sum_{ij \in S} h(Y_{ij} X_{ij}), \tag{2}$$

where c is a trade-off constraint and $h(z) = \max(0, 1 - z)$ is the hinge-loss, minimizing which is equivalent to minimizing slack variables $\xi_i j \geq 0$ in soft-margin constraints $Y_{ij} X_{ij} \geq 1 - \xi_{ij}$.

Matrix factorization can be also viewed as a simultaneous learning of feature vectors and linear classifiers. Assume a factorization $X = UV$ is found, the rows of the $n \times k$ matrix U can be viewed as a set of n *feature vectors*, while the columns of V can be viewed as *linear classifiers*, and the entries of the matrix X are the results of classification using these classifiers. The original entries in the matrix Y can be viewed as *labels* for the corresponding feature vectors, and the matrix factorization task can be interpreted as finding simultaneously a collection of feature vectors (rows in U) and a set of linear classifiers (columns in V), given a set of labeled samples (columns in the original matrix Y), such that a good prediction of unobserved entries can be made. Particulary, the MMMF formulation above can be viewed as learning a collection of maximum-margin classifiers (SVMs) simultaneously with learning a common set of features.

3 Active Learning with MMMF

Standard collaborative prediction approaches, including MMMF, assumed no control over the data collection process. However, we have a choice between different actions that provide us with new samples. For example, in online recommendation systems, we choose a product suggested to the current user; in network latency prediction, we can request a probe (e.g., ping) between a particular pair of nodes; in content distribution systems, we can suggest a mirror site for a file download, and so on. Such additional measurements can greatly improve the predictive accuracy of our model, but they also have a cost (e.g., potentially low bandwidth or high network latency if a server is not selected carefully). One one hand, we wish to choose the next sample which is most-informative and leads to greatest improvement in the predictive accuracy in the future (i.e., yields better exploration), while on the other hand we want to avoid choosing samples which might be too costly by exploiting our current predictions about the sample costs (i.e., the corresponding predicted performance). Such exploration vs exploitation trade-offs must be considered as a part of our decision-making.

As mentioned in the previous section, MMMF approach can be viewed as learning a collection of SVMs, which provides a natural way for combining MMMF with various *active learning* approaches developed for SVMs. In this paper, we a simple heuristic margin-based approach, that uses the margin as our confidence estimate in the predictions made, similarly to active learning approach of [8]. Namely, [8] suggest to choose next the the *minimum-margin* sample, i.e. the one which is

closest to the separating hyperplane, and can be viewed as the one we are least confident about. This heuristic was shown to be successful in practice, and is very efficient computationally[1].

The active sampling algorithm (active MMMF, or A-MMMF), works as follows:

```
A-MMMF
    1. Given a sparse matrix Y,
       learn approximation X = MMMF(Y)
    2. Using current predictions, actively
       select S minimum-margin samples and
       request their labels
    3. Add new samples to Y
    4. Repeat 1-3 until no significant improvement in prediction is likely
```

The idea of min-margin active sampling is also demonstrated in Figure 1.

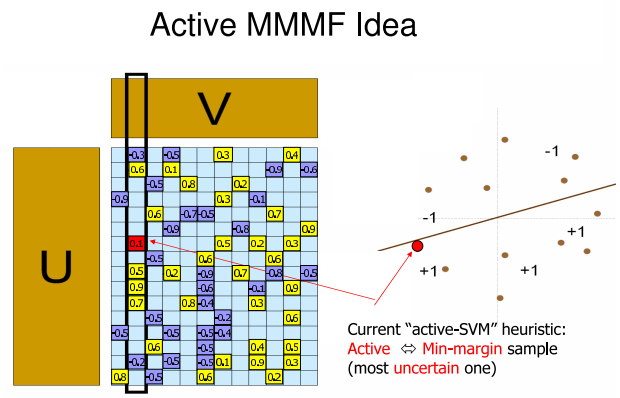

Figure 1: Main idea of active learning in MMMF: choose the most "uncertain" sample next, where the margin measures the confidence in the prediction (i.e. we are least confident in predictions made for the instances closest to separating line between positive and negative examples).

Besides the "aggressive" *most-uncertain* sampling we also tried several other active sampling approaches that take into account the cost of sampling and may decide to be more "conservative" about sample choice, e.g., when sampling also means providing a service such as file download, where besides improving the future accuracy we are also concerned with the immediate cost of sampling. We assume binary prediction problems (e.g., the performance over or under a specified threshold) and assume that positive samples (e.g., high bandwidth or product ratings) have less cost than the negative samples. We then explore several "cost-conscious" active learning heuristics, such as *most-uncertain-positive* heuristic that chooses positive min-margin sample, as well as *least-uncertain* (max-margin) and *least-uncertain-positive* heuristics, which which should corresponds to prediction we are most confident about. However, such sample selection may lead to a less accurate model, as we show in the empirical section where the different sampling heuristics are compared on several data sets.

4 Empirical Evaluation

We tested active learning approaches described above on the data from various practical applications. We select a subset of most populated rows and columns, to increase matrix density for testing purposes. We then split each dataset into a training, testing and active subsets, where active subset

[1]Although min-margin heuristic may be ineffective for problems with large label noise close to the separating hyperplane, as noticed by [**?**], in many collaborative prediction settings there is little or no noise in labeling: e.g., user's preferences for a movie typically do not change.

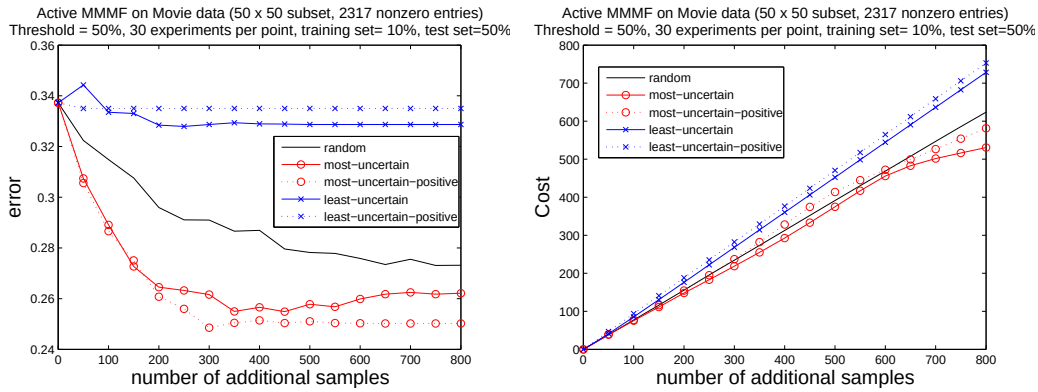

Figure 2: results on Movie dataset: (a) prediction accuracy and (b) total cost of sampling.

simulates the source of active samples. A training set is typically selected to be quite small (e.g., 5% of the whole dataset), to imitate learning "almost from scratch". We plot the prediction error on the training dataset, for each of the active strategies compared random sampling of the same number of instances.

The first dataset, called *Movies*, includes movie ratings collected through user interactions with the site www.movielens.org. This includes ratings on the scale of 1 (worst) to 5 (best) by 500 users of 1000 movies. We selected a subset of 50 users and 50 movies that correspond to most-populated rows and columns. We then impose a threshold to make the data binary, i.e. we assume that the rating larger than 3 is considered "good". The results are presented in Figure 2a. We can see that the most-uncertain sampling provides a significant improvement over the random sampling, while the max-margin sampling, as expected, is not very informative and practically does not improve the error. We also computed the actual cost of sampling, assuming no cost for positive samples selected and unit cost of the negative ones, and plotted it in Figure 2b. Clearly, random sampling would roughly have the slope of the cost curve equal to the proportion of negative samples in the data. Surprisingly, the alternative strategies did not deviate significantly from this random-sampling linear cost growth, although we can see some deviation for larger number of samples. We can see that the most-uncertain and most-uncertain-positive strategies are actually better not just in terms of future predictive error, but also in terms of total sampling cost.

Similar results were observed in multiple systems management applications. We used several network latency datasets obtained from PlanetLab pairwise ping project, the NLANR Active Measurement (AMP) project, and P2PSim project – the datasets used previously by [5]. We also used the data we obtained from an IBM-internal content distribution systems called *downloadGrid*. DownloadGrid architecture has some similarities with the Internet-based Gnutella, Napster and BitTorrent file distribution systems as it allows peer-to-peer file downloading; however, it combines the peer-to-peer approach with centralized decision-making architecture for matching "clients" and "servers". Centralized architecture is mainly motivated by security issues, but can also provide opportunities for optimization of the overall system's performance: for example, it allows to collect system-wide historic data about the previous file downloads which can be used later for predicting the end-to-end performance for previously unobserved client-server pairs, and for (nearly) optimal selection of a server(s) for a particular client file request.

All data sets were transformed to binary (discretized) by imposing a certain threshold on the performance, such as 50, 70 or 90% (e.g., 50% threshold corresponds to a median). We also used subsets of each dataset, including only most "active" nodes that yield most populated rows and columns (see for details the corresponding Figures). We started with initial training set which contained only 5% of non-zero entries in each of the matrices, and set aside a test set containing 50% of the non-zero entries. The rest was used as a pool for sampling. The results are presented in Figures 2b and **??**. The Y axis shows the prediction accuracy, while the X-axis shows the number of additional samples selected. As expected, we observe that active min-margin sampling results into consistently more accurate predictions than the random sampling using same number of samples. The active most-uncertain-positive heuristic comes close to the most-uncertain one, although a bit less accurate, and

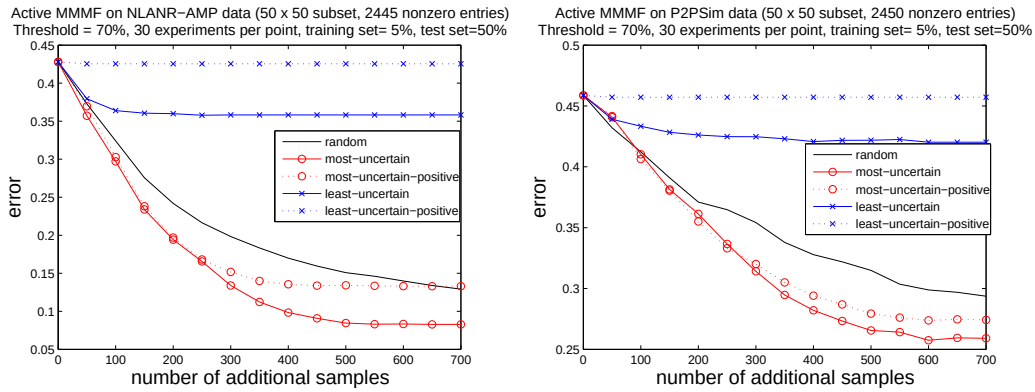

Figure 3: Prediction results on (a) NLANR-AMP and (b) P2PSim data

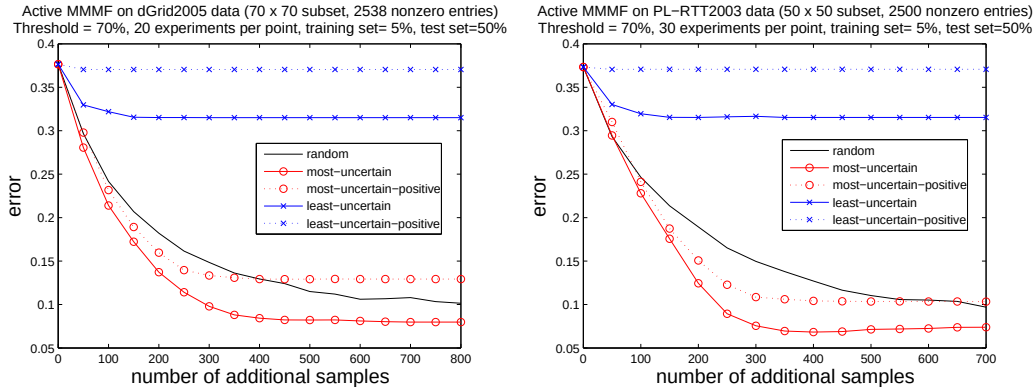

Figure 4: Prediction results on (a) dGrid2005 and (b) PL-RTT2003 data

may provide a safer alternative if the cost of sampling depends on the value of sample (positive being ”good”).

5 Conclusions and Future Work

We proposed a simple, computationally efficient active sampling extension of the state-of-art MMMMF method for collaborative prediction and compares several active-sampling strategies, both on traditional collaborative filtering domain (movie rating prediction) and on novel application domain – distributed computer systems management. Promising empirical results are demonstrated on all applications considered.

There are multiple directions for future work. One includes incorporating more advanced active sampling approaches into MMMF, that will come closer to more rigorous value-of-information (VOI) analysis, and can hopefully provide some theoretical guarantees. Another direction is extending the (cost-sensitive) active learning to exploration vs exploitation methods for sequential decision making that will trade active sampling versus choosing already known good actions. Finally, an important future direction is to further improve the computational efficiency of active MMMF by making it incremental, i.e. reusing the solution obtained on the previous sampling iteration without having to solve the MMMF optimization from scratch. Unfortunately, existing incremental approach to solving SVMs, such as, for example, the active set approach [Scheinberg], cannot be directly extended to the MMMF and require devising incremental optimization particularly tailored to MMMF formulation.

References

[1] C. Boutilier, R. Zemel, and B. Marlin. Active collaborative filtering. In *Proc. of UAI*, pages 98–106, 2003.

[2] P. Drineas, I. Kerenidis, and P. Raghavan. Competitive Recommendation Systems. In *Proc. of the 34th ACM Symposium on Theory of Computing (STOC)*, pages 82–90, 2002.

[3] Rong Jin and Luo Si. A Bayesian approach toward active learning for collaborative filtering. In *Proc. of UAI*, pages 278–285, 2004.

[4] Daniel D. Lee and H. Sebastian Seung. Algorithms for Non-negative Matrix Factorization. In *NIPS*, pages 556–562, 2000.

[5] Y. Mao and L. K. Saul. Modeling Distances in Large-Scale Networks by Matrix Factorization. In *Proc. of IMC 2004, Oct 2004*.

[6] Jason Rennie and Nathan Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proc. of ICML 2005*, pages 713–719, 2005.

[7] Nathan Srebro, Jason Rennie, and Tommi Jaakkola. Maximum Margin Matrix Factorizations. In *Advances in Neural Information Processing Systems (NIPS-04)*, 2004.

[8] S. Tong and D. Koller. Support Vector Machine Active Learning with Applications to Text Classification. In *Proc. of ICML 2000*, 2000.