Imputation-Boosted Collaborative Filtering Using Machine Learning Classifiers

Xiaoyuan Su
Computer Science and Engineering
Florida Atlantic University, USA
Boca Raton, FL 33431, USA
xsu@fau.edu

Taghi M. Khoshgoftaar
Xingquan Zhu
Computer Science and Engineering
Florida Atlantic University, USA
Boca Raton, FL 33431, USA
{taghi, xqzhu}@cse.fau.edu

Russell Greiner
Department of Computing Science
University of Alberta
Edmonton, AB T6G 2E8, Canada
greiner@cs.ualberta.ca

ABSTRACT

As data sparsity remains a significant challenge for collaborative filtering (*CF*), we conjecture that predicted ratings based on imputed data may be more accurate than those based on the originally very sparse rating data. In this paper, we propose a framework of *imputation-boosted collaborative filtering (IBCF)*, which first uses an imputation technique, or perhaps machine learned classifier, to fill-in the sparse user-item rating matrix, then runs a traditional *Pearson correlation-based CF* algorithm on this matrix to predict a novel rating. Empirical results show that *IBCF* using machine learning classifiers can improve predictive accuracy of *CF* tasks. In particular, *IBCF* using a classifier capable of dealing well with missing data, such as naïve Bayes, can outperform the *content-boosted CF* (a representative hybrid *CF* algorithm) and *IBCF using PMM* (predictive mean matching, a state-of-the-art imputation technique), without using external content information.

Categories and Subject Descriptors: H.2.8 [Database Management]: Database Applications - Data Mining

General Terms: Algorithms.

Keywords: Collaborative filtering, recommendation systems, imputation techniques, machine learning classifiers, incomplete data.

1. INTRODUCTION

Many of today's most effective recommender systems are based on *collaborative filtering (CF)*, which basically assumes that, if user U_1's ratings for several items are similar to those of U_2, then U_1's ratings for a novel item will likely resemble U_2's. This motivates collecting tables of user-item ratings (like Table 1(a)), and using them to guide future prediction. Unfortunately, these tables tend to be very sparse -- i.e., most users do not rate most items.

As imputation techniques are frequently used to deal with missing data, we consider first using some imputation method to fill in these tables, then making predictions based on this imputed data, anticipating this may yield more accurate predictions. In this paper, we propose a framework of imputation-boosted collaborative filtering (*IBCF*), which use an imputation technique to impute the missing data to create a *pseudo rating matrix* (i.e., transforming Table 1(a) to 1(b)), which is then used by a traditional *Pearson correlation-based CF* (*Pearson CF*) algorithm [4] to produce final recommendations.

We implement various *IBCF* systems, whose imputation techniques range from the state-of-the-art imputation technique, predictive mean matching (*PMM*) [2], to several commonly-used machine learning classifiers (from *WEKA* [5]) -- trained either on the pure rating data (Table 1(a)) or on content data -- as well as the *content-boosted CF* (*CBCF*) algorithm [3], (which is a representative hybrid *CF* algorithm), then comprehensively investigate their performance against one another.

We evaluate our systems on the real-world *MovieLens* [1] data, based on the mean absolute error (*MAE*), which computes the average of the absolute difference between the predictions and the true ratings.

Table 1: (a) original rating data, (b) imputed rating data

	I_1	I_2	I_3	I_4	I_5
U_1			**4**		
U_2	**2**		**4**	**3**	
U_3		**3**	**3**	**3**	**3**
U_4		**4**		**2**	

	I_1	I_2	I_3	I_4	I_5
U_1	*2*	*3*	**4**	*3*	*3*
U_2	**2**	*2*	**4**	**3**	*3*
U_3	*3*	**3**	**3**	**3**	**3**
U_4	*2*	**4**	*3*	**2**	*3*

2. FRAMEWORK

The main steps of the imputation-boosted *CF* using machine learning classifiers are:

(1) Divide the originally large ratings data into reasonably-sized subsets. Given the original *MovieLens* [1] data with 100,000 observed ratings (each from {1,2,...,5}) of 943 users on 1682 items (movies), we rank each item based on the number of users that have rated it and use this ranking to sort the items into 20 disjoint subsets, whose missing rates (sparsity) ranged from 64.5% to 99.3%. The first dataset had 943 users and the 65 most-rated movies, the 2nd dataset had 943 users and the next 65 movies, etc. We ignore the remaining 382 movies (out of 1682), which each had five or fewer user ratings; this involves a total of 958 ratings -- 0.958% of the original 100,000 ratings.

(2) Apply the machine learning classifiers to form *pseudo rating matrices*: for each item *i=1...n*, train a machine learner on

SAC'08, March 16-20, 2008, Fortaleza, Ceará, Brazil.

columns *{1, … i-1, i+1, … n}* of the pure rating matrix to produce a classifier for item *i*, then use this learned classifier to provide labels for the missing values of that *i*-th item. Alternatively, when trained on content data (which contains four demographic attributes: age, sex, occupation, and postal code), set the class label to be the value of the *i*-th item.

The machine learning algorithms applied in the *IBCF* framework include decision tree (*C4.5*), decision table (d*Table*), lazy Bayesian rules (*LBR*), logistic regression (*LR*), naïve Bayes (*NB*), neural networks (*NN*), one rule (*OneR*), decision list (*PART*), and support vector machine (*SVM*). We also devise an ensemble classifier, by using imputed data from 7 out of 9 classifiers that have the top performance, and used the threshold of 6 for the majority voting (this has the best result in comparison with other threshold values in our preliminary experiments). That is, if no class value receives at least 6 votes from classifiers, it will be left as missing.

(3) Apply a user-based *Pearson CF* to the imputed data to produce the prediction

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_u) \cdot w_{a,u}}{\sum_{u \in U} \left| w_{a,u} \right|}$$

of user *a* for item *i*, where $\bar{r}_a$ and $\bar{r}_u$ are the average ratings for user *a* and user *u* on all other items that both have rated, and $w_{a,u}$ is the *Pearson correlation* [4] between users *a* and *u*. The $u \in U$ summations are over all the users who have rated the item *i*.

3. EXPERIMENTAL RESULTS

Table 2 shows the overall *MAE* performance of *IBCF* using each classifier, which is aggregated from the *MAE* values of the 20 subsets (and incorporating the 0.958% ratings obtained based on the default voting, 3 in our work), weighted by the percentage of the total ratings of each dataset. In addition, *IBCF using PMM*, *content-boosted CF* and the traditional *Pearson CF* have *MAE* scores of 0.718, 0.726 and 0.792 respectively.

When trained on content data, the performance of *IBCF using machine learning classifiers* has the following ranking (here, ">>" means *significantly better than* (with 1-tailed t-test, $p<0.05$, or 95% confidence interval), ">" means *better than* (with $0.05 \le p < 0.2$), ">≈" means *slightly better than or equivalent with* (with $0.2 \le p < 0.5$)):

(ranking for IBCFs trained on content) (IBCF-) ensemble >≈ NB >≈ LBR >> SVM >> NN >> OneR >> dTable >> LR >> PART >> C4.5 >> Pearson CF.

When trained on pure rating data, the performance ranking is:

(ranking for IBCFs trained on pure ratings) (IBCF-) NB > ensemble >> SVM >> LBR >> C4.5 >> LR >> NN >> dTable >> PART >> OneR >> Pearson CF.

The overall performance ranking of selected *CF* algorithms:

(overall ranking) IBCF-NB (ratings) > IBCF-ensemble (ratings) > IBCF-PMM >> IBCF-ensemble (content) >≈ CBCF >> IBCF-SVM (ratings) >> Pearson CF.

These empirical results show that *IBCFs* using any of the machine learning classifiers and *IBCF* using *PMM* have better performance than the traditional *Pearson CF*. When trained on content information, *IBCF using an ensemble classifier*, *IBCF using NB*, and *IBCF using LBR* have better performance than the other *IBCF*s using machine learning classifiers. When trained on pure rating data, the best performer is the *IBCF using NB*.

IBCF using *NB* (trained on pure rating data) is better than *IBCF-PMM*, with 1-tailed t-test $p<0.039$. It is 5.5% better than *content-boosted CF*, and 13.4% better than the traditional *Pearson CF*, in terms of *MAE*.

As hybrid *CF* algorithms rely on external content information that is usually not available, the fact that our *IBCF*s (*IBCF-NB*, *IBCF using an ensemble classifier*, and *IBCF-PMM*) trained on pure rating data (i.e., without using content information) can outperform the *content-boosted CF*, has even more significance.

Table 2. MAE scores of the IBCF using machine learning classifiers on the MovieLens data (1st row, IBCFs trained on pure ratings; 2nd row, IBCFs trained on content data)

IBCF NB	IBCF ensem	IBCF SVM	IBCF LBR	IBCF C4.5	IBCF LR	IBCF NN	IBCF dTab	IBCF PART	IBCF OneR
0.686	0.712	0.743	0.746	0.754	0.756	0.76	0.761	0.764	0.768
0.726	**0.721**	0.73	0.726	0.754	0.751	0.733	0.748	0.752	0.738

4. CONCLUSION

As high data sparsity remains a challenge for *CF* algorithms, we propose the imputation-boosted collaborative filtering (*IBCF*) algorithms, which boost *CF* performance by making recommendations from imputed data instead of the original rating data. Besides implementing the *IBCF* using predictive mean matching (*PMM*), we implement and comprehensively investigate *IBCF using machine learning classifiers*, which respectively use nine commonly-used machine learning classifiers and an ensemble classifier to impute the missing rating data, trained either on the content data, or on the pure rating data. Empirical results show that *IBCF using naïve Bayes*, *IBCF using an ensemble classifier* (both trained on pure rating data), and *IBCF using PMM*, can outperform the *content-boosted CF*; and *IBCF* using any of the machine learning classifiers can achieve better performance than the traditional *Pearson CF*. In addition, we see that *IBCF* using a machine learning classifier capable of dealing well with missing data, such as *naïve Bayes*, can perform better than *IBCF* using a high quality imputation technique, such as *PMM*, in terms of *MAE*.

5. REFERENCES

[1] GroupLens. http://movielens.umn.edu.

[2] Little, R.J.A. Missing-Data Adjustments in Large Surveys. *Business & Economic Statistics*, 6(3), 287-296, 1988.

[3] Melville, P., Mooney, R.J. and Nagarajan, R. Content-Boosted Collaborative Filtering for Improved Recommendations, *AAAI*, 2002.

[4] Sarwar, B.M., Karypis, G., Konstan, J.A. and Riedl, J. Item-based Collaborative Filtering Recommendation Algorithms, *WWW*, pp. 285-295, 2001.

[5] Witten, I.H. and Frank, E. *Data Mining: Practical machine learning tools and techniques*, 2nd Edition, 2005.