Social network collaborative filtering

RONG ZHENG, DENNIS WILKINSON & FOSTER PROVOST[1]

This paper demonstrates that "social network collaborative filtering" (SNCF), wherein user-selected like-minded alters are used to make predictions, can rival traditional user-to-user collaborative filtering (CF) in predictive accuracy. Using a unique data set from an online community where users rated items and also created social networking links specifically intended to represent like-minded "allies," we use SNCF and traditional CF to predict ratings by networked users. We find that SNCF using generic "friend" alters is moderately worse than the better CF techniques, but outperforms benchmarks such as by-item or by-user average rating; generic friends often are not like-minded. However, SNCF using "ally" alters is competitive with CF. These results are significant because SNCF is tremendously more computationally efficient than traditional user-user CF and may be implemented in large-scale web commerce and social networking communities. It is notoriously difficult to distinguish the contributions of social influence (where allies influence users) and "social" selection (where users are simply effective at selecting like-minded people as their allies). Nonetheless, comparing similarity over time, we do show no evidence of strong social influence among allies or friends.

Categories and Subject Descriptors:
General Terms:
Additional Key Words and Phrases:

1. INTRODUCTION

One of the main challenges faced by providers of goods and information in the Internet age is to enable users to find relevant items, when the number of items is too large to support traditional browsing. This challenge arises in e-commerce websites as well as sites where the items are shared freely and the provider derives other benefits from increasing user visits and satisfaction. This latter category includes a range of online community, discussion, and social networking sites.

Recommender systems are thus relevant in a wide range of online settings. When the number of items and users is large, on the order of thousands or millions, collaborative filtering (CF) methods are a popular choice because they take advantage of the plentiful data on user activity without having to perform a detailed content-based analysis on the items (e.g. [Goldberg et al. 1992, Resnick et al. 1994, Adomavicius and Tuzhilin 2005]). Since web technologies can easily allow providers to present a different set of recommendations for each user, user-user CF is particularly relevant. This form of CF tailors its recommendations to spe-

[1] Authors' addresses: Zheng & Provost, NYU Stern School of Business; Wilkinson, HP Labs, Palo Alto, CA

Working Paper CeDER-8-08, Center for Digital Economy Research, Stern School of Business, NYU. New York, NY 10012.

cific users by examining the behavior of algorithmically identified like-minded "alters" (sometimes called "neighbors" in the literature) and basing its suggestions on the alters' past behavior.

However, computational cost remains a difficult hurdle in user-user CF because of the time required to identify the best alters from among the thousands or millions of candidates. Indeed, while CF is a popular choice for large Internet recommenders such as Amazon, Yahoo, and Netflix, these systems use item-to-item filtering only [Sarwar et al. 2000, Linden et al. 2003, Mull 2006, Bennett 2006] primarily because user-to-user algorithms are too computationally expensive (e.g., [Linden et al. 2003]). Although clustering methods designed to automatically identify groups of similar users have been combined with CF in a number of ways (e.g., [Anglade et al. 2007, Koren 2008], these methods do not significantly improve the cost problem and to our knowledge are not in use in industry.

The recent rise of online social networking provides a new source of similarity data for recommender systems, in the form of explicit user-generated connections linking pairs of users. Social networking has undergone a phenomenal rise in popularity since its inception in 2002 to the point that some 55% of U.S. teenagers, and as many as 90% of students at some U.S. colleges participate to some degree, most of them at least once per day [Lenhart and Madden 2007, Ellison et al. 2006]. Even among older people, social networking is gaining traction, and demographic and business trends indicate that its popularity and reach will only continue to increase (e.g., [Richtel 2007]).

Social networking links are used to facilitate communication and to allow users to stay updated on their alters' recent activities, ideas and opinions. The most common type of link by far is the "friend" link, a rather vague reciprocal connection that correlates only weakly to the traditional concept of friendship [Boyd 2006]. In this paper, we also examine “ally” links, which users established specifically when they shared opinions, preferences, or taste within a given context. These links are particularly relevant to recommender systems, as we show.

Since Internet social networking has attained mainstream popularity, it is now more than reasonable to consider how social networking ties might impact recommender systems. Indeed, retailers such as Amazon and Ebay have recently added relatively successful social networking features to their websites [Amazon Friends 2008, Ebay Neighborhoods 2008], providing evidence that Internet users will engage in social networking within a specific commercial setting. In these settings, the social network helps users select items to purchase. In other instances such as last.fm [BBC 2003], the goal is to enhance users’ enjoyment of the site’s content. Meanwhile, social networking giants Myspace and Facebook have ventured into the world of eCommerce with some success as well [Bustos 2007, Olsen 2006].

In spite of these developments, there has been little academic work on the subject to date. Previous work has centered on techniques for inferring a social network from existing copurchase or corating data (e.g. [Kautz et al. 1997, Perugini et al.

2005]). Applications of these include alleviation of the sparseness (cold-start) problem [Huang et al. 2004] and the study of the spread of recommendations over a network [Mirza et al. 2003, Leskovec et al. 2007]. The propagation of trust through implicit [Papagelis et al. 2005, Weng et al. 2006] and explicit [Massa and Avesani 2004, Golbeck 2006] connections has also been studied. The results show the better performance especially in cold-start situations. However, it remains unclear how to successfully elicit trust evaluation from users in large-scale online systems, and a trust network may not be as easily generated as modern online social networks.

Given an existing social network, the simplest way to apply it to collaborative filtering is to replace the algorithmically selected alters of traditional CF by user-selected social network alters. In doing this, the computationally intensive step of identifying like-minded alters is removed from the CF algorithm, thereby reducing the complexity from $O(N^2+NM)$ to $O(1)$ for a system of N users and M items (the number of social networking alters is generally bounded by a constant much smaller than N or M for realistic systems—very rarely exceeding 100 or 200 [Golder et al. 2007]). This procedure, called "social network collaborative filtering," was proposed by Zheng et al. [2007]. In this preliminary study, the data were rather sparse and SNCF on the few friends links was outperformed by traditional CF in predicting which user would purchase which item.

This paper presents an empirical study of social network collaborative filtering (SNCF) using a comprehensive data set of 1.2 million votes (ratings) and 25,000 social networking links from Essembly.com. This site invited users (i) to vote on each other's user-generated resolves, expressing four levels of agreement or disagreement, and (ii) to form social networking links. The Essembly social network is notable because users created both "friends" links and "allies" links, this latter a type of preference link implying like-mindedness. Other than this special link type, however, Essembly is strikingly similar to other online communities in its patterns of user activity [Wilkinson 2008] and social (friends) network structure [Hogg et al. 2008], suggesting that it is representative of online communities in general.

We show that SNCF using friends links improves on simple benchmarks such as by-user and by-item average, and performs as well as naïve forms of traditional CF, although not as well as other forms. This result is perhaps not surprising because friends tend to share similarities only to a limited degree [McPherson et al. 2001]. However, since SNCF is so computationally efficient, a friends approach represents a reasonable alternative to traditional CF in some situations.

We also show that when ally links are used in SNCF, it is competitive with even the best simple CF methods, and outperforms most CF. This is a significant result from a practical standpoint, for reasons of computational complexity and because of the popularity of online social networking.

We supplement these results with a comparison of the similarities of Essembly friends and allies, in terms of their voting histories. The results show that allies are significantly more similar than friends, particularly allies who are not also

friends (both links are allowed between any pair of users). Conversely, friends who are not allies are significantly less similar than friends in general. This suggests that the notion of friendship confounds like-mindedness with other compatibilities and that discerning the like-mindedness in particular can lead to improved predictions.

Our experimental results also raise interesting questions about why user-selected alters were equally or more predictive than algorithm-identified CF alters, which unfortunately we cannot answer with the data available. Previous work on the correlation of behavior between linked entities has identified two types of cause: social influence effects, where peers affect each others' decisions after they are linked, and selection effects, where users establish links based on homophily [Manski 1993]. In our context, selection effects might account for allies being more predictive than CF alters because of subtleties of human relationships that are not detectable in ratings data alone. We note that Essembly allows users to personalize their home pages and post comments, which might cast light on subtle aspects of their personalities. We perform a simple analysis of the Essembly data, based on how the similarity of connected pairs changed though time, which shows no evidence of social influence.

2. METHODS

This section describes the various collaborative filtering methods we employ in exploring the relationship between alters selected by algorithm and by user. CF and SNCF methods have "parameters," the setting of which can affect the results, and it is not clear a priori or based on prior research exactly which settings are optimal for a given domain. Therefore when drawing conclusions based on comparisons of CF methods, we assess several different variations in an effort to establish a more robust comparison.

The two main CF parameters we consider are: (i) which similarity measure to use in weighting (potential) alters' previous ratings, and (ii) how many alters to consider; we also consider (iii) the distance/error measure used to assess the result (see e.g. [Anh 2008, Sarwar et al. 2001]).

2.1. Collaborative filtering

CF is one of the most popular and most-studied techniques for recommender systems. Using only the past interactive information between users and items, such as ratings or purchases, CF can estimate ratings for the items that have not been seen by a user, or probabilities associated with a potential purchase. Based on the estimations, the system can present to users the items they are most likely to be interested in. As described in the introduction, CF of one sort or another is applied in many real world settings, such as Amazon, Netflix and Last.fm. The goal of these systems includes both increased sales and increased user satisfaction.

In our study, we focus on rating-based, user-to-user CF, where the goal is to predict the value for a rating variable r_{ij} quantifying user i's opinion of the item j; for example, how much user i likes item j, or agrees with item j. The rating variable r usually takes integer or real values within a certain range. Given a set of users

and items and rating records, the input of the problem can be formulated as an $M \times N$ *rating matrix* R associated with M users and N items. The cell r_{ij} corresponds to the rating cast by user i on item j. The rating recommendation problem then is predicting rating r in the unfilled cells in matrix R based on the values in the filled cells.

User-based CF algorithms first construct a $M \times M$ user similarity matrix W. The similarity score w_{st} between user s and user t is calculated based on associated the rating vectors, i.e. the s^{th} and t^{th} row vector in matrix R. There are various ways to measure similarity, discussed in the next section, which vary in effectiveness from system to system. A high similarity score w_{st} indicates that user s and user t may have similar preferences since they have previously rated products in a similar way. In the calculation, the multiplication of matrix W with R implements the above idea and generates the predicted rating scores in the rating matrix R. One modification to this procedure that we will return to is to modify the similarity matrix by zeroing out low scores; for example, in direct analogy to a k-nearest neighbor algorithm for non-parametric estimation, in CF one may set all but the top-k scores to zero.

2.1.1. Similarity measures:

As we describe above, the calculated similarity between users' rating vectors determines the weighting in aggregating other users' ratings on the item of interest. In the literature, various metrics have been proposed [Breese et al. 1998, Herlocker et al. 1999], including mean absolute distance (MAD), mean squared distance (MSD), Pearson's correlation (COR), which measures the linear correlation between two vectors of ratings, and cosine similarity (COS), which looks at the angle between two vectors of ratings, where a smaller angle is regarded as implying greater similarity. In a recent study, Ahn [2008] identified some anomalies of above measures when there are a small number of common ratings. He proposed a new measure called proximity-impact-popularity (PIP) in order to fix the identified problems, and showed it to be comparable with other in normal situations and better in difficult situations. Although the comparison of different similarity measures is not our focus, we present results for all in order that the conclusions are not based on a poor arbitrary choice.[2]

[2] Note that estimating the predictive performance of the best similarity measure is more complex than just looking in our results at the performance of the method that performed best, because of the bias inherent in such a multiple comparison procedure [Jensen & Cohen 2000]. Although it is not relevant for our comparisons, if one wanted to estimate the performance of the best method, the best method should be selected based on one set of data, and the performance estimated on a different set of data (as with *nested* cross-validation).

2.1.2. Number of alters

Alters are defined as a set of users who previously had ratings on an item which is the one the predicted rating is given to. It is sometimes called neighbor in the literature. With additional cost, a subset of alters could be selected to improve the prediction by setting a threshold or using the best *k* alters. Herlocker et al. [1999] observed that the performance increased initially and decreased later as more and more ratings are used.

In this paper, we intended to compare CF with SNCF in reasonably fair settings and the optimization of CF is not the focus of our study. Therefore, we implemented two variants of CF in terms of alter selection. One is the full set of alters. The other is a subset that contains the same number of alter as SNCF in a prediction, denoted as k-CF in the paper.

2.1.3. Significance weighting

Since most similarity measures don't account for the number of common ratings when calculating the distance between rating vectors, it is desirable to assign different confidence weights to them [Herlocker et al. 1999]. In our implementation, if two users have fewer than 10 commonly rated items, we applied a significance weight of n/10. If there were more than 10 co-rated items, then a significance weight of 1 is applied.

2.1.4. Producing and accessing predictions

There are two basic ways to combine alters' ratings into a prediction. One way is to compute the weighted average of the alters' ratings, using the similarity as the weight. This method assumes all users rate based on the same distribution. The other is to first compute the average deviation of an alter's rating from that alter's mean rating, where the mean rating is taken over all ratings by the alter. The prediction is then the predicting item's own mean rating adjusted by the average deviation-from-mean from alters. This second way is naturally suitable for the correlation based similarity measures. We mainly used the weighted average method in our study. The deviation method is only used for correlation similarity.

We adopted two standard error evaluation methods: unless otherwise indicated we compare based on the mean absolute error (MAE); we also present results for mean squared error (MSE).

2.2. Social network collaborative filtering

Instead of generating predictions based on algorithmically calculated "like-minded" users in CF, SNCF predicts ratings based on user self-selected friends. A weighted average of ratings from SNCF alters (a subset of friends who also rated the predicting item) becomes the predicted rating. We implemented the simplest the version of SNCF with equal weights, which avoids the expensive similarity calculation. We also implemented SNCF with unequal weights, denoted w-SNCF, in which the SNCF alters are weighted using the similarity measures described above for CF.

3. EXPERIMENTAL SETUP

In this section we describe the data and the basic structure of the analyses used for the results presented next.

3.1. Data

For this study we used a comprehensive data set of all activity on the Essembly website, www.essembly.com, from its inception in August 2005 until December 12, 2006 (the date of data capture). This website provides a forum in which users posted "resolves" and other users voted on them. Example resolves include statements such as "All speech -- even the most offensive -- should be protected under the First Amendment," or "Animals do not have rights. People who are mentally handicapped, and unborn babies, do." Voting is on a four point scale, with the options being "strong agree," "lean agree," "lean disagree," and "strong disagree."

Users also participate in social networking, forming both "friend" and "ally" links.[3] Link formation is by user choice only, and was reciprocal (i.e., both users had to consent to form the link). Links allow users to see their alters' votes and also facilitate communication.

As of our date of data collection, the website had 5,358 users who participated in both voting and the social networks. These users had created 24,953 resolves and cast 1.24 million votes on these resolves. In agreement with the system used internally in Essembly [Chan 2007] we denoted these votes by the integers 4, 3, 2, and 1, where 4 denoted "strong agree". The votes were slanted slightly toward "strong agree," with a mean of 2.32 and a standard deviation of 1.34.

The distribution of number of votes per user is strongly right skewed and well described by a power law [Wilkinson 2008]. This heavy tail form is almost ubiquitous in most forms of online activity, including peer production [Wilkinson 2008], media sharing [Ceyhan 2008], online discussion [Whittaker 1998], rating and commenting [Carlson and Doyle 2000], and open source contributions [Wilkinson 2008], among others. The heavy right skew means that a small fraction of very active participants are responsible for the large majority of the activity, an unfortunate reality for recommender systems attempting to provide accuracy for all users.

The Essembly friends social network had 4,873 nodes and 13,516 links. As shown by Hogg et al. [2008], the degree distribution was strongly right-skewed, well described by a power law with exponential cutoff (gamma distribution). The transitivity, modularity, diameter, average path length, and giant component fraction all fell well within the range observed for online friends networks. In other words, the Essembly friends network was very typical and representative of other online social networks.

[3] There was a third link type, "nemesis," for users who tended to disagree, which we do not include in our analysis.

The Essembly allies social network had 3,261 nodes and 15,593 links. Perhaps not surprisingly, it was quite different from the friends network in its network topology, having a smaller transitivity, modularity, diameter and average path length. These values for the allies network did not fall in the typical range of online social networks.

We finally note that Essembly exhibited an approximately linear growth rate in the number of users and the number of votes. Our data did not include time-stamps on the formation of links, but there is no reason to believe that the rate of link formation differed greatly from the growth rate of the system as a whole. Growth is typical of online systems and social networks, and the methodology of our measurements was designed with growth in mind, as described below.

3.2. Walk-forward analysis

In order to simulate the real-world prediction task, we evaluate CF and SNCF using a "walk-forward" analysis: for predicting a single data point v (a vote by a specific user on a specific resolve), the prediction can only be based on data points with time-stamps prior to the time-stamp of v. We will call the data points on which the prediction is based on the *training* data. As is standard, each prediction will be compared to actual, observed value for v. Errors will be averaged over a set V of test data, the selection of which is the subject of experimental design, as described in Section 3.3.

Importantly, each prediction on a test point $v \in V$ may generally be based on a different training set. This evaluation method is much more computationally expensive than a simple hold-out or cross-validation evaluation, because the similarity matrix in this case is not static. As each new vote is obtained, the similarities between all users must be updated. However, as this is the computation and prediction complexity encountered by a real-world recommender system, it is appropriate to simulate it here.

3.3. Testing set selection

To compare versions of CF and SNCF, we must select data points for comparison carefully. In order to minimize the possible pollution of the results by basing a prediction on a link that was formed after the vote was cast (recall that we do not have time stamps for the formation of social links), we first carefully selected three testing "supersets" as follows.

a) Testing superset **T1** consists of data that are time-stamped within the most recent few days (1 for allies and 4 for friends, in order to get roughly the same number of data points), and by a user whose first vote was cast three months earlier than the testing vote. Considering our data set spans 498 days, the chance is very small that a long-term user formed the social links in the last few days.

b) Testing superset **T2** consists of data that are cast by users, each of whose last vote is one month before the last day of the data capture, and whose first vote is three months earlier than the testing vote. Again, the chance

is very small that a long-term user formed the social links in the last few days before his last vote. However, because of the timing, the **T2** testing set covers an entirely different set of users as compared to **T1**.

c) Testing superset **T3** consists of each user's last vote. This testing set supplements **T1** and **T2,** and also it is not likely a user will make further social links after his/her last voting activity.

From these testing supersets we select testing sets for particular methods. In order to compare a pair of methods, we restrict the test set to include only data points for which there are sufficient alters (four in these experiments) for each method. This reflects our desire to compare the methods when they apply, rather than to assess how broadly they apply. So, for example, for SNCF using friends links, we would choose a testing superset T_i, and then from T_i select those user-resolve pairs for which there are at least four alters for prediction.

3.4. Benchmarks for comparison

We used three techniques as the benchmark predictions against which to compare the collaborative filtering techniques. The benchmarks comprise the global average voting score over the entire data set (BM-all), the average score for the resolve in question (BM-Item), and the average score by the user in question (BM-User). BM-item and BM-User are implemented consistently with the walk-forward analysis, in that the average is taken based only on the training data.

4. RESULTS

We now present our main results. Recall that our focus is the performance of collaborative filtering based on alters specially selected by users as being like-minded. We first compare traditional collaborative filtering to SNCF using friends. The friends are user-selected, but are not necessarily selected as being like-minded. Then we look directly at the allies. Finally, since the friends and allies sets generally overlap, we carefully separate them into several categories and show a clear ordering of predictive performance based on an intuitive interpretation of the categories. Finally, we present a brief investigation into whether there is evidence of social influence among the allies.

4.1. SNCF versus CF with friends links

Figure 1 and Table 1 compare SNCF using friends links (SNCF-friends) with the various alternative methods for estimating future votes. Table 1 also shows the results of one-tailed t-tests, with the null hypothesis that MAE(SNCF) is not greater than MAE(alternative). For convenience, if MAE(alternative)>MAE(SNCF), Table 1 shows the results of the corresponding t-test (the other tail), with a minus sign. In each case, the sample size is the size of the testing data set. Bold values show significance at alpha =0.05.

Ignoring the k-CF entries for the moment, Figure 1 and Table 1 show that SNCF-friends does quite well. SNCF-friends produces substantially lower error than even the best of the three benchmarks (BM-Item). Furthermore, SNCF-friends is competitive with full-matrix CF. Specifically, SNCF-friends has significantly

lower error in most cases, and there are no cases where CF has significantly lower error rate than SNCF-friends.

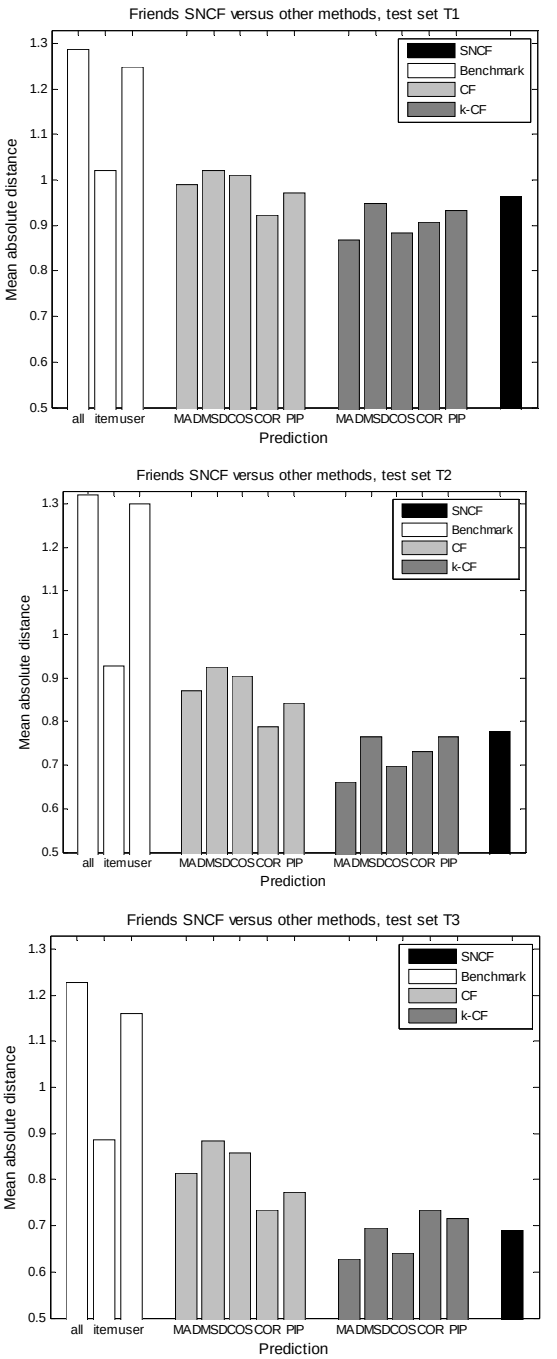

Figure 1. MAE comparison between SNCF and CF with friends links

Table 1. SNCF versus CF with friends links

Test Set		SNCF	BM-All	BM-Item	BM-User	CF-MAD	CF-MSD	CF-COS	CF-COR	CF-PIP
T1	MAE	0.963	1.287	1.021	1.248	0.991	1.021	1.01	0.924	0.973
	MSE	1.498	1.846	1.445	1.826	1.381	1.444	1.418	1.261	1.344
	t test		**0.000**	**0.017**	**0.000**	0.209	**0.049**	0.091	-0.132	0.389
T2	MAE	0.778	1.321	0.929	1.3	0.87	0.926	0.905	0.788	0.843
	MSE	1.165	1.921	1.208	1.905	1.094	1.2	1.157	0.97	1.041
	t test		**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.006**	0.358	0.013
T3	MAE	0.691	1.228	0.886	1.16	0.814	0.884	0.859	0.735	0.774
	MSE	1.075	1.701	1.11	1.632	0.968	1.105	1.052	0.847	0.9
	t test		**0.000**	**0.000**	**0.000**	**0.004**	**0.000**	**0.000**	0.167	**0.034**

As discussed above, in certain domains CF produces better predictions if the number of alters used is limited, rather than using similarity weighting across the full matrix. Therefore, we also compare SNCF with k-nearest-neighbor CF ("k-CF"). In order to produce an equal-footing comparison focused on our main question of the relative performance of user-selected alters and system-selected alters, for each comparison k-CF sets k equal to the number of friends used by the method to which it is being compared.

Figure 1 shows the comparison of all the CF methods with the basic SNCF, and Table 2 includes the detailed comparison of each CF-S methods (S being the similarity metric) with the corresponding wSNCF2-S method, which also weights the alters by similarity in its predictions. This table shows a pair of t-test rows for each testing set. The upper row is the same as in Table 1; the lower row compares the two columns it spans, in the same manner as the prior t-tests. The performance of the various wSNCF versions has very little variance and is very close to the unweighted SNCF. Therefore, although we show the individual values for completeness in the table, we ignore wSNCF for simplicity in the analysis, focusing on unweighted SNCF—which is only slightly worse in performance, much simpler, and strikingly efficient computationally.

As shown clearly in Figure 1, these results show that for this domain, k-CF indeed is more accurate than full-matrix CF, irrespective of the similarity metric employed. More importantly for the purpose of this paper, k-CF now outperforms SNCF-friends, significantly in some cases. Nevertheless, given the computational advantage of SNCF-friends, it is of note that one has to be careful to design CF well in order to achieve consistently lower error rate.

Table 2. SNCF versus k-CF with friends links.

Testing set		SNCF	wSNCF-MAD	k-CF-MAD	wSNCF-MSD	k-CF-MSD	wSNCF-COS	k-CF-COS	wSNCF-COR	k-CF-COR	wSNCF-PIP	k-CF-PIP
T1	MAE	0.963	0.946	0.869	0.963	0.948	0.955	0.884	0.853	0.906	0.941	0.932
	MSE	1.498	1.463	1.394	1.498	1.435	1.48	1.371	1.392	1.383	1.5	1.344
	t test		-0.332	**-0.008**	0.499	-0.349	-0.349	**-0.02**	-0.398	-0.068	-0.284	-0.206
	t test		**-0.023**		-0.351		**-0.032**		0.35		-0.409	
T2	MAE	0.778	0.753	0.662	0.778	0.765	0.768	0.698	0.751	0.731	0.751	0.765
	MSE	1.165	1.119	0.985	1.165	1.11	1.449	1.025	1.078	1.055	1.173	1.087
	t test		-0.229	**-0.000**	0.497	-0.35	-0.386	**-0.013**	-0.2	-0.074	0.33	-0.215
			-0.003		-0.348		**-0.016**		-0.267		0.33	
T3	MAE	0.691	0.69	0.629	0.704	0.695	0.697	0.642	0.789	0.735	0.691	0.715
	MSE	1.075	1.103	0.936	1.124	1.108	1.114	0.978	1.183	1.022	1.165	1.193
	t test		-0.49	-0.114	0.406	0.473	0.452	-0.173	**0.029**	0.192	0.5	0.328
	t test		-0.119		0.433		-0.147		-0.132		-0.333	

4.2. SNCF versus CF with allies links

This section presents results directly analogous to those of section 4.1, where instead of "friend" social network links in the SNCF prediction, we use "ally" links (SNCF-allies). As discussed previously, Essembly's user-created "ally" links were meant to be used to connect users who were like-minded in some sense.

More specifically, the data set used in this section is the same as that of section 4.1, except that the allies social network is used in for SNCF instead of the friends social network. We note that while the allies network was denser than the friends network, in terms of average number of neighbors per node, this should not affect our measurement since the experimental design considered only nodes with at least 4 alters in both cases, and because beyond that a higher number of alters was not correlated to higher accuracy of prediction.

As in section 4.1, we compare SNCF-allies with CF, both making use of a full set of alters (CF) and a matching number of alters (k-CF). Figure 2 and Table 3 show the clear dominance of SNCF-allies over full-matrix CF. Figure 2 and Table 4 show that SNCF-allies much more competitive than SNCF-friends (comparing with the results from section 4.1). SNCF-allies clearly dominates full-matrix CF, and is generally competitive even with k-NN CF (k-CF). The very best k-CF (generally, k-CF-MAD) still beats SNCF, but one has to choose the similarity measure for k-CF just right to beat SNCF-allies significantly. More importantly as shown by Figure 2, the magnitude of the difference between SNCF-allies and the best k-CF is relatively small. For some choices of similarity metric, SNCF-allies actually produces statistically significantly lower error as compared to k-CF.

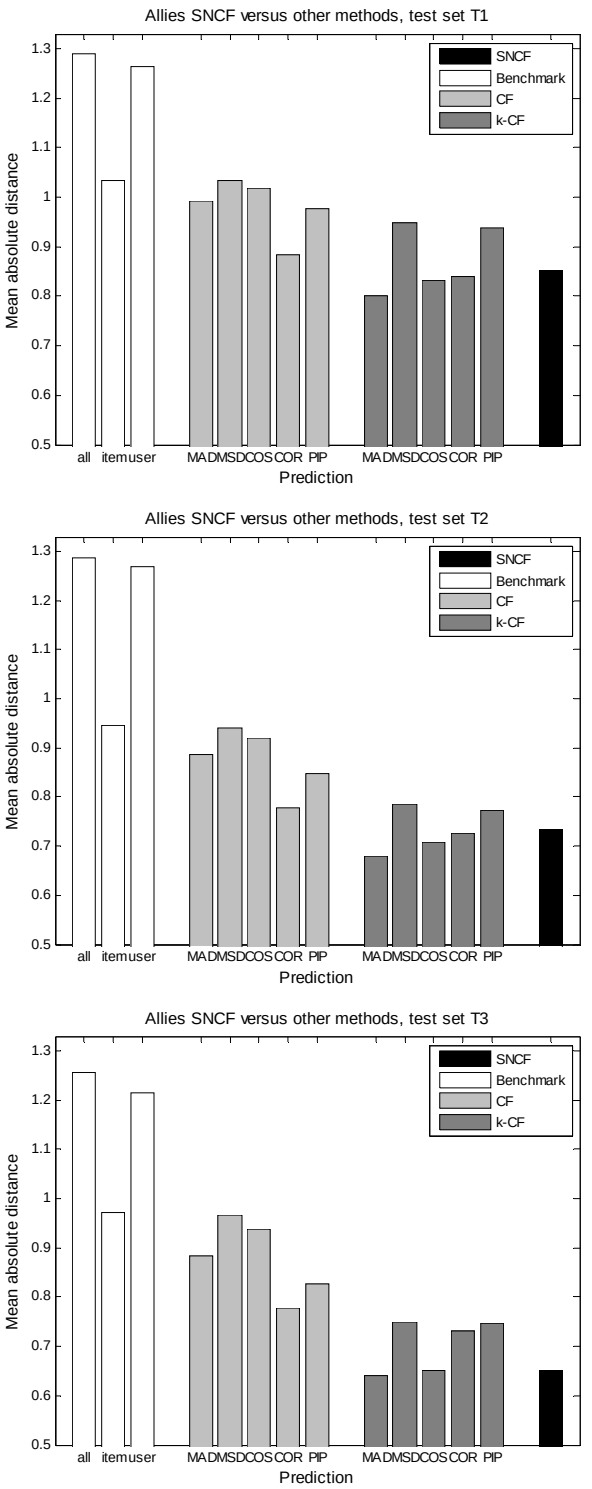

Figure 2. MAE comparison between SNCF and CF with allies links.

Table 3. MAE comparison between SNCF and CF with allies links

Test set		SNCF	BM-All	BM-Item	BM-User	CF-MAD	CF-MSD	CF-COS	CF-COR	CF-PIP
T1	MAE	0.852	1.29	1.034	1.265	0.993	1.034	1.018	0.885	0.978
	MSE	1.306	1.849	1.462	1.829	1.365	1.461	1.422	1.152	1.395
	t test		**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.001**	0.144	**0.000**
T2	MAE	0.735	1.288	0.945	1.27	0.886	0.941	0.921	0.777	0.848
	MSE	1.049	1.848	1.263	1.807	1.147	1.256	1.214	0.969	1.081
	t test		**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.042**	**0.000**
T3	MAE	0.652	1.256	0.971	1.214	0.884	0.968	0.938	0.778	0.827
	MSE	1.005	1.755	1.302	1.727	1.109	1.294	1.225	0.939	1.013
	t test		**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.001**	**0.000**

Table 4. SNCF versus k-CF with allies links.

Testing set		SNCF	wSNCF-MAD	k-CF-MAD	wSNCF-SD	k-CF-MSD	wSNCF-COS	k-CF-COS	wSNCF-COR	k-CF-COR	wSNCF-PIP	k-CF-PIP
T1	MAE	0.852	0.843	0.802	0.852	0.949	0.848	0.833	0.844	0.839	0.857	0.938
	MSE	1.306	1.291	1.259	1.306	1.426	1.299	1.268	1.255	1.249	1.38	1.395
	t test		-0.407	-0.08	0.5	**0.000**	0.463	-0.298	-0.414	-0.359	0.437	**0.006**
	t test		-0.12		**0.002**		-0.33		-0.442		**0.01**	
T2	MAE	0.735	0.726	0.679	0.735	0.787	0.731	0.708	0.734	0.727	0.724	0.773
	MSE	1.049	1.035	1.008	1.049	1.12	1.042	1.034	1.011	1.03	1.071	1.09
	t test		-0.327	**-0.017**	0.492	**0.023**	-0.433	-0.149	-0.48	-0.382	-0.338	0.073
	t test		-0.327		0.12		-0.446		-0.383		0.373	
T3	MAE	0.652	0.653	0.641	0.658	0.75	0.655	0.651	0.723	0.731	0.684	0.748
	MSE	1.005	1.014	0.988	1.02	1.212	1.015	0.989	1.049	1.03	1.148	1.241
	t test		0.492	-0.405	0.447	**0.017**	0.473	0.489	0.053	0.033	0.251	**0.02**
			-0.397		**0.023**		-0.462		0.422		0.09	

4.3. Allies, friends and taste similarity

Our driving hypothesis has been that collaborative filtering based on user-selected, like-minded alters might be competitive enough with standard collaborative filtering to be considered as a complement or an alternative, especially considering its advantages in computational complexity. The above results show that indeed SNCF is competitive with collaborative filtering, especially when the users select alters specifically as being like-minded (the allies).[4]

[4] The absolute numbers for the SNCF-friends and SNCF-allies experiments from the prior two sections are not directly comparable, because the predictions are on different test sets, depending on

Using the collaborative filtering framework, we can assess directly how similar the user-selected alters are comparatively. Figure 3 shows a cartoon of the space of different categories of user-selected alters; the sizes of the bubbles for Allies and Friends, and the intersections and differences, are in rough proportion to the actual numbers. The five categories are: allies, friends, ally-friends, non-friend-allies, non-ally-friends.

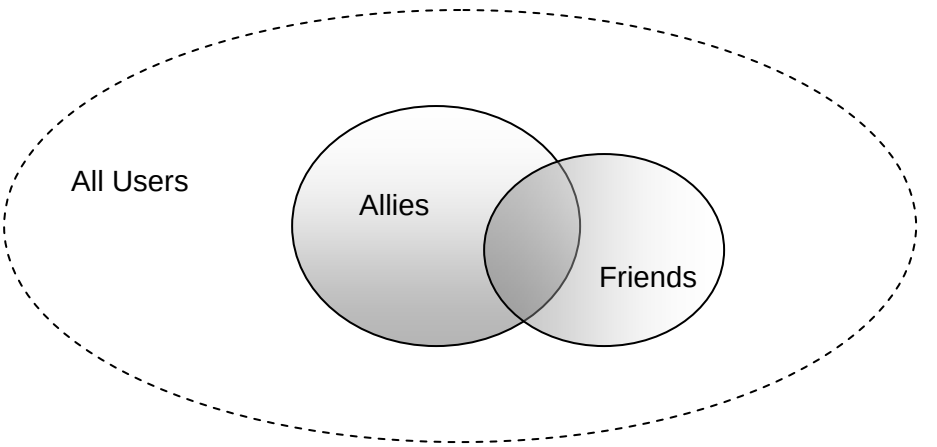

Figure 3. Different categories of alters correspond to different regions. Sizes of bubbles for Allies, Friends, overlap, and differences are in approximate proportion to actual numbers.

Table 5. Comparison of similarity among different categories of user-selected alters (minimum 10 votes on common resolves).

Category	mean	stdev	number
all pairs (sampled)	0.584	0.162	4.3 M
non-ally friends	0.682	0.146	7203
all friends	0.702	0.142	10671
ally and friend	0.746	0.123	3392
all allies	0.751	0.112	14717
allies only	0.754	0.107	11216

Table 5 shows the average similarity across the entire user base of five different categories of alters. More specifically, the table shows the mean similarity, using the MAD similarity described in section 2.1, across dyads in the corresponding category. Note that the similarity measure here is in fact 1-(MAD/3), so that users voting identically have a similarity of 1. A similarity of 0 would be obtained

---

the conditions described in section 3.3. For example, the benchmark accuracies suggest that the friends T3 data set is considerably easier than the allies T3 data set.

by a pair of users who always vote "strong agree" (4) or "strong disagree" (1) opposite each other. Only users with at least 10 common resolves are considered.

All of the differences in Table 5 are significant in mean, in the sense that we can completely rule out the hypothesis that the data in any two sets came from distributions with identical means ($p<0.001$, Wilcoxon rank sum test). The average similarity for random pairs of users is greater than 0.5 because the votes are slightly slanted towards 4 overall (strong agree).

Table 5 demonstrates that friends who are not selected as allies are strikingly less similar to the user than allies who are not among the user's friends. This shows clearly the advantage of user-selection of like-minded alters: users have significant numbers of friends who are not like-minded, which dilutes the set of alters for SNCF.

To illustrate this dilution in predictive performance, Table 6 presents a walk-forward predictive analysis between allies and non-ally friends. Specifically, we made predictions on each data item where there are at least 4 allies and 4 friends. To avoid there being more alters in one set than another, we randomly sampled a subset of the bigger set to match the size of the smaller one. The MAE values in Table 6 are the averaged scores over 10 repetitions of the sampling. The table shows clearly the dominance of SNCF using allies alters over SNCF using non-ally friends alters.

Table 6. Predictive performance of SNCF-allies Vs. SNCF-non-ally-friends

	SNCF-allies	SNCF-non-ally-friends	No. Of Test Data
MAE (Avg.)	0.892	1.062	1756
t test on the mean of absolute error: $p < 0.001$			

4.4. Social influence versus selection effects

The results so far in this section show clearly that users can find alters, allies, who are more like-minded than their friends in general, and specifically who can provide competitive accuracy in predicting future voting behavior. What we have not yet done, and cannot do definitively based on these data, is provide an exact explanation of why.

The most straightforward reason is simply that people can do a good job of selecting like-minded alters, based on comparing voting behavior, comparing statistics on voting similarity, looking at textual descriptions of others' interests, or some combination. In this case, it would make sense to design information systems that help users to find like-minded alters, which will be increasingly difficult without system aid as the size of the user base grows.

On the other hand, the reason for similarity and effective prediction may be due to social influence. Users who have established a linkage may affect the behavior (voting, rating, buying) of each other. However, if social influence is at play,

it is not clear that simple algorithmic techniques to help users find similar alters will be as effective; organically chosen alters may have greater social influence.

Unfortunately, distinguishing precisely between the social influence and other reasons for similarity among grouped or linked individuals is notoriously difficult [Manski 1993, Oestricher-Singer & Sundararajan 2008, Bramoulle et al. 2007, Anagnostopoulos et al. 2008] and requires a level of detail not present in these data. Previous work has demonstrated evidence of social influence [Hill et al. 2006, Birke 2008, Crandall et al. 2008], although these studies obtained somewhat ambiguous results and were not always able to account for exogenous selection effects.

Nonetheless, one simple analysis can be performed which could give evidence of strong social influence. Specifically, we are considering social influence as exhibited by increasing similarity of linked users over time. Thus, unless the links are all made at the very end of the data collection period (which for the Essembly data is highly unlikely), then strong social influence should lead to an increase in similarity over time.[5]

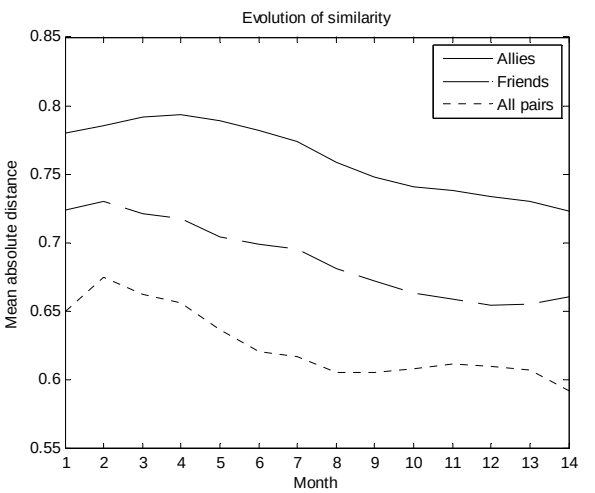

Figure 4. Similarity of allies, friends, and all pairs of users declines gradually over time.

Figure 4 plots similarity among allies, friends, and all pairs of users over time. The general downward trend across all users seems to indicate that Essembly resolves over time became more polarized; however, the similar decline in the similarity of allies and friends indicates that perhaps they just represented issues with greater random variance in opinion—although it is difficult to account precisely for simple reversion to the mean (e.g., after ally selection by perceived similarity).

[5] Notice that observing an increase over time does not necessarily prove social influence, because there could be other explanations.

In any case, Figure 4 shows no evidence of strong social influence—at least not strong enough to overcome declines in similarity for other reasons. This is in contrast to Crandall et al. [2008] who claim to have observed some social influence effects among Wikipedia editors; admittedly, Wikipedia is a different setting with a greater degree of personal interaction in some cases.

5. DISCUSSION AND LIMITATIONS

Our results suggest that online content providers should consider allowing users to choose "allies" whenever recommendation is desired, and even facilitating the choice. What allies represent depends on the domain of application and sort of recommendation desired. For example, e-retailers like Amazon.com would like to recommend products that their customers will like (and buy). Therefore, it may be useful for them to help users to identify "taste allies", thereby removing some of the guesswork inherent in statistical inference from sparse data. Prior work [Zheng et al. 2007] showed that SNCF-friends was not as accurate as traditional CF for predicting Amazon purchases (in agreement with the results above for Essembly); however, it may be SNCF-allies based on taste would be more accurate—or nevertheless useful as a complementary technique.

To examine taste-allies in a very different domain, we examined the subset of the Amazon friends network who revealed their purchases. More specifically, this data set consists of 1206 customers who made 13,494 purchases on 11,773 distinct items. These 1206 customers are also connected by the "Amazon Friends" social network. For these Amazon friends, we compared the purchases of specific products to those of a similar selection of 1206 "non-friend" customers, who also revealed their purchases, but were not in the friends network. For the top-10 book purchases of the Amazon friends, each was purchased by more than 40 customers in the network; the #1 purchase was purchased by over 100 customers. However, of these top-10 purchases, only 5 were purchased at all by the non-friends customers, and none by more than 3 non-friends.

Dead Ringer by Ken Douglas (Author)
Tangerine Dream by Ken Douglas (Author), Jack Stewart (Author)
Hurricane by Jack Stewart
Night Witch by Jack Priest
Scorpion by Jack Stewart
Ragged Man by Jack Priest
Gecko by Jack Priest
Harry Potter and the Deathly Hallows
Running Scared by Ken Douglas
Desperation Moon by Ken Douglas
Diamond Sky by Ken Douglas (Author), Jack Stewart (Author)

Figure 5. Top-10 books purchased by the subset of the Amazon friends network who revealed their purchases, a like-minded "thriller" social network.

This result provides evidence that these customer-selected friends indeed have similar taste. Examination of the purchases themselves strengthens the evidence. The top-10 purchases (shown in Figure 5) primarily were books from a particular genre (thrillers), and with the exception of the Harry Potter book, all by the same three authors.

One limitation of our study is the relatively small scale of Essembly vis-à-vis the scale of massive ecommerce systems. It could be that even with system-based aid in selecting allies, massive CF would out-perform SNCF-allies by a much larger margin. Unfortunately, we are aware of no data set that includes the information necessary to compare user-selected allies to CF-selected allies with a massive set of potential alters. In any case, the issue of computational performance still rears its head: given that user-to-user CF is such an elegant and intuitively satisfying concept, if SNCF is feasible where CF is not, it makes sense to consider whether it indeed improves user satisfaction or profit, even if in principle CF might perform somewhat better (if only it could be run).

Moreover, our result that SNCF-allies is competitive with k-CF, which considerably outperforms full-matrix CF, suggests that it is really the most-similar alters that drive the performance. Thus, even with a massive set of potential alters, as long as the system can provide help in selection, the task of selecting strong allies may not be onerous for individual users. As we have mentioned, the distribution of activity per person is heavily right skewed in most online settings which have been studied. This means that a few active people provide most of the ratings. So no matter how large the system, it always is difficult to make good recommendations for most people (a massive cold-start problem).

For comparison, in the Netflix challenge the by-item benchmark in MSE was 1.10, while the very best, highly optimized kNN CF methods reduce the error to about 0.83 MSE [Bell & Koren 2007]. This is a 25% improvement. In our case for T3, which is the closest in benchmark error rate to Netflix's test set, the by-item benchmark is 1.3 for MSE, and the error the "best" CF we tried (k-CF-cosine) is 0.99 MSE. This is an improvement of 24%. Even though the Netflix data set has 500,000 users, the best CF on that improved BM-item about the same as our straightforward k-CF improved over BM-item. This is only suggestive evidence, but qualitatively the effectiveness of CF on the Essembly problem is not very different than on a much larger, one.[6]

Self-selected alters, as opposed to algorithmically selected alters, might prove more accurate for people with few ratings because the algorithmically detected similarities could be chance. As future work, using the Essembly data we could assess the support (or lack thereof) for this conjecture. Consider an oversimplified model where user-selection clearly would be beneficial: each user has a latent taste distribution. His behavior is drawn from a mixture of this distribution and another distribution (e.g., representing present-buying behavior). Thus, for some individuals, especially with limited data, CF-selected alters would include

[6] Even if you adjust our errors by 5/4, since our scale was 4 and theirs was 5, the qualitative conclusion would be the same.

alters who are not so similar to the user's true taste distribution, but rather to the other distribution. However, the user may well be able to distinguish those alters who are similar in taste from the others.

A related limitation is that we have compared SNCF using generic friends to SNCF with taste-selected alters only on Essembly data. Thus, the results must be interpreted as a "proof of concept" that users can select links to allow better recommendations. Whether the results generalize to other settings, in particular product recommender systems, would be an important extension. Unfortunately, to our knowledge there are no other available data sets in a recommendation context that include with self-selected like-minded "allies", as opposed to other sorts of "friends." However, within Essembly, we have shown that the results are robust to different prediction settings.

6. CONCLUSION

This study shows that the extremely efficient collaborative-filtering technique consisting of choosing "recommenders" to be one's self-selected, like-minded social-network neighbors can be remarkably effective, particularly when users select the alters specifically as being like minded. For the domain studied, this social-network collaborative filtering was competitive with traditional user-to-user collaborative filtering systems. The selection of users specifically to be like-minded appears to be important, because the users have significant numbers of linked "friends" who are not like-minded, which reduces the accuracy of a system based on generic friends.

These results suggest that designers of web recommender systems should consider supplying users with facilities to self-select like-minded alters, in addition to generic friends. They can provide fast, effective recommendations.

ACKNOWLEDGEMENTS

Thanks to Brian Dalessandro for help analyzing the Amazon friends network, to Chris Chan and Jimmy Kittiyachavalit for help in accessing the data, to Tad Hogg for helpful discussions, and to NEC for a Faculty Fellowship.

REFERENCES

Adomavicius, G. and Tuzhilin, A."Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions", IEEE Transactions on Knowledge and Data Engineering 17 (6) June 2005 pp. 734-749

Ahn, Hyung Jun, (2008). A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem Volume 178 , Issue 1 Pages 37-51.

Amazon website social network info (2008), author unknown, http://www.amazon.com/gp/help/customer/display.html?nodeId=468600, accessed 9/14/2008

Anagnostopoulos, A., R. Kumar, and M. Mahdian (2008). Influence and correlation in social networks. KDD'08.

Anglade, A., M. Tiemann, F. Vignoli (2007). Complex-network theoretic clustering for identifying groups of similar listeners in p2p systems. To appear in Proceedings of the 2007 ACM conference on Recommender systems.

Badrul Sarwar, George Karypis, Joseph Konstan, John Riedl Item-based collaborative filtering recommendation algorithms (2001), in Proc. of the 10th International World Wide Web Conference (WWW10), Hong Kong, pp. 285-295.

BBC Online (2003), Website offers new view of music, published Thursday, 27 March, 2003, accessed 9/22/2008.

Birke, D. (2008). Who your are or whom you know? Consumption interdependences in social networks. SCECR-08. (and related white paper).

Bramoulle, Y. , H. Djebbari and B. Fortin (2007). Identification of Peer Effects through Social Networks. CIRPEE Working Paper No. 07-05. Available at SSRN: http://ssrn.com/abstract=965818.

Breese, J.S., D. Heckerman and C, Kadie (1998). Empirical analysis of predictive algorithms for collaborative filtering. in 14th Conference on Uncertainty in Artificial Intelligence, pp. 43-52.

Bennett, Jim. The Cinematch System: Operation, Scale Coverage, Accuracy, Impact. Presented at Recommenders 06 Summer School.

Bell, Robert M. and Yehuda Koren, Lessons from the Netflix Prize Challenge ACM SIGKDD Explorations Newsletter Volume 9 , Issue 2, pages 75-79.

Boyd, Danah (2006). Friends, friendsters, and top 8: Writing community into being on social network sites by *First Monday*, volume 11, number 12, URL:
http://firstmonday.org/issues/issue11_12/boyd/index.html

Bustos, L (2007). Facebook e-commerce applications. http://www.getelastic.com/ecommerce-facebook-applications-reviewed/, posted Oct. 12, 2007, accessed 9/14/2008.

Carlson, J.M. and Doyle, J. (2000). Highly Optimized Tolerance: A Mechanism for Power Laws in Designed Systems. Physical Review E, 60(2), 1412–1427.

Ceyhan, S. (2008), personal communication.

Chan, C. (2007), personal communication.

Crandall, D., D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri (2008). Feedback effects between similarity and social influence in online communities. KDD-08.

Ebay neighborhoods, author unknown, http://neighborhoods.ebay.com/, accessed 9/14/2008.

Ellison, N., C. Steinfeld, and C. Lampe. Spatially bounded online social networks and social capital: the role of Facebook. In Proceedings of the Annual Conference of the International Communication Association, 2006.

Goldberg, David; David Nichols, Brain M. Oki, Douglas Terry (1992). "Using collaborative filtering to weave an information tapestry". *Communications of the ACM* **35** (12): 61–70.

Golder, Scott A., Dennis Wilkinson and Bernardo A. Huberman (2007). "Rhythms of Social Interaction: Messaging within a Massive Online Network" *3rd International Conference on Communities and Technologies* (CT2007). East Lansing, MI. June 28-30, 2007.

Golbeck, J. 2006. Generating Predictive Movie Recommendations from Trust in Social Networks. Proceedings of the Fourth International Conference on Trust Management.

Herlocker, J. L, Konstan, J. A, Borchers, A. and Riedl, J (1999). An algorithmic framework for performing collaborative filtering", in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 230--237. ACM Press, (1999).

Hill, S., F. Provost, and C. Volinsky. Network-based marketing: Identifying likely adopters via consumer networks. Statistical Science, 22(2):256–276, 2006a.

Hogg, T., Wilkinson, D., Szabo, G. and Brzozowski, M. 2008. Multiple Relationship Types in Online Communities and Social Networks. In *Proc. of the AAAI Spring Symposium on Social Information Processing.* AAAI Press.

Huang, Z., H. Chen, D. Zeng (2004). Applying Associative Retrieval Techniques to Alleviate the Sparsity Problem in Collaborative Filtering. ACM Trans. on Information Systems, 22(1), 116–142.

Jensen, D. D., P. R. Cohen (2000). Multiple comparisons in induction algorithms, Machine Learning Journal 38, pp. 309–338.

Kautz, H., Selman, B., and Shah, M. Referral Web: combining social networks and collaborative filtering. Communications of the ACM,40(3): 63-65, 1997.

Koren, Y. (2008), Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model. In Proceedings of the 2008 SIGKDD Conference.

Lenhart, A., and M. Madden (2007). Social Networking Websites and Teens: An Overview Washington, DC: Pew Internet & American Life Project.

Leskovec, J., L. A. Adamic, B. A. Huberman (2007) The dynamics of viral marketing. ACM Transactions on the Web 1(1).

Linden, G., Smith, B., and York, K., Amazon. com Recommendations: Item-to-Item Collaborative Filtering - IEEE INTERNET COMPUTING, Jan.-Feb. 2003 pp. 77-80.

Manski, C (1993). Identification of Endogenous Social Effects: The Reflection Problem. The Review of Economic Studies, Vol. 60, No. 3 (Jul., 1993), pp. 531-542.

Massa, P. and P. Avesani. Trust-aware collaborative filtering for recommender systems. Proc. of International Conference on Cooperative. 2004.

McPherson M., Lynn Smith-Lovin, and James M Cook (2001). Birds of a Feather: Homophily in Social Networks Annual Review of Sociology Vol. 27: 415-444.

Mirza, B.J.,Keller, B.J., and Ramakrishnan, N. (2003). Studying Recommendation Algorithms by Graph Analysis. Journal of Intelligent Information Systems, 20(2), 131–160.

Mull, M. The Characteristics of a High-Volume Recommender System, presentation at Recommenders 06 Summer School.

Oestreicher-Singer, G. and Sundararajan, A., 2007. Recommendation Networks and Peer Effects in Electronic Commerce.

Oestreicher-Singer, G. and A. Sundararajan (2008). Identifying Social Effects Using Networked ECommerce Data. SCECR-08.

Olsen, P./ (2006) Murdoch's MySpace Makes E-Commerce Debut, Forbes.com , May 15, 2006 accessed 9/22/2008.

Papagelis, D. Plexousakis, and T. Kutsuras. Alleviating the sparsity problem of collaborative filtering using trust inferences. In Proceedings of iTrust 2005, pages 224–239, 2005.

Perugini, S., M. A. Goncalves, & E. A. Fox (2004). A connection centric survey of recommender system research. Journal of Intelligent Information Systems 23[1].

Resnick, Paul, *Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, John Riedl* "GroupLens: An Open Architecture for Collaborative Filtering of Netnews," *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, Chapel Hill, NC: Pages 175-186.

Richtel. M. (2007) New Social Sites Cater to People of a Certain Age. New York Times, September 12, 2007.

Sarwar , Badrul M, George Karypis, Joseph A Konstan, John Riedl. Analysis of Recommender Algorithms for E-commerce Proceedings of the 2nd ACM conference on Electronic commerce Minneapolis, MN Pages: 158 – 167, 2000.

Weng, J., C. Miao, and A. Goh. Improving collaborative filtering with trust-based metrics. Proc. of the 2006 ACM symposium on Applied computing, pages 1860–1864, New York, NY, USA, 2006.

Whittaker, S., L. Terveen, W. Hill, and L. Cherny. The dynamics of mass interaction (1998). In CSCW '98: Proceedings of the 1998 ACM conference on Computer supported cooperative work, pages 257–264, New York, NY, USA.

Wilkinson, D. M. "Strong regularities in online peer production." In *Proceedings of the 2008 ACM Conference on E-Commerce*, Chicago, IL, July 2008.

Zheng, R., F. Provost and A. Ghose (2007). Social Network Collaborative Filtering: Preliminary Results. In Proceedings of the Sixth Workshop on eBusiness (WEB2007).