$$Obj_{rss} = \sum_{\mathbf{x}} \sum_{\mathbf{z} \in friends(\mathbf{x})} \frac{1}{2} S^{+}_{\mathbf{x},\mathbf{z}} \|U\mathbf{x} - U\mathbf{z}\|_2^2 \tag{13}$$
$$= \sum_{\mathbf{x}} \sum_{\mathbf{z} \in friends(\mathbf{x})} \frac{1}{2} S^{+}_{\mathbf{x},\mathbf{z}} (\mathbf{x} - \mathbf{z})^T U^T U (\mathbf{x} - \mathbf{z})$$

While we note these extensions are straightforward, they have the simple property that they allow the system to learn the latent user projection matrix U *as a function of user features* in order to minimize the social regularization penalty. Just as the Matchbox objective in Section 3.2.1 allows us to exploit user and item features in MF-based CF, these new social regularization objectives permit more flexibility in exploiting user features in *learning* user similarity.

3.3.2 Hybrid Information Diffusion + SCF (Obj_{phy})

One major weakness of MF methods is that they cannot model direct joint features over user and items — they must model user and item features independently in order to compute the independent latent projections $U\mathbf{x}$ and $U\mathbf{z}$. Unfortunately, this prevents standard MF objectives from modeling direct user-to-user information diffusion [3] — the unidirectional flow of information (e.g., links) from one user to another. This is problematic because if user $\mathbf{x}$ *always* likes what $\mathbf{z}$ has posted or liked, then we would like to shortcut the latent representation and simply learn to recommend user $\mathbf{z}$'s liked or posted items to user $\mathbf{x}$.

We fix this deficiency of MF by introducing another objective component in addition to the standard MF objective, which serves as a simple linear regressor for such information diffusion observations. The resulting hybrid objective component then becomes a combination of latent MF and linear regression objectives.

For the linear regressor $\mathbf{w}^T \mathbf{f}_{\mathbf{x},\mathbf{y}}$, we make use of the *same* weight vector $\mathbf{w}$ and feature vector $\mathbf{f}_{\mathbf{x},\mathbf{y}}$ mentioned in Section 2.2; $\mathbf{f}_{\mathbf{x},\mathbf{y}}$ is fully defined for our empirical evaluation in Section 4.2.3. As previously noted, $\mathbf{f}_{\mathbf{x},\mathbf{y}}$ includes *joint* user and item features from the social network, in particular binary *information diffusion* [3] features for *each* friend $\mathbf{z} \in friends_{\mathbf{x}}$ indicating if $\mathbf{z}$ liked (or disliked) $\mathbf{y}$. As a consequence, learning $\mathbf{w}$ allows the linear regressor to predict in a personalized way for any user $\mathbf{x}$ whether they are likely to follow their friend $\mathbf{z}$'s preference for $\mathbf{y}$.

Formally, to define our hybrid information diffusion plus SCF objective, we additively combine the output of the linear regression prediction with the Matchbox prediction:

$$Obj_{phy} = \sum_{(\mathbf{x},\mathbf{y}) \in D} \frac{1}{2} (R_{\mathbf{x},\mathbf{y}} - [\sigma]\mathbf{w}^T \mathbf{f}_{\mathbf{x},\mathbf{y}} - [\sigma]\mathbf{x}^T U^T V y)^2 \tag{14}$$

3.3.3 Co-preference Regularization (Obj_{cp})

A crucial aspect missing from other SCF methods is that while two users may not be globally similar or opposite in their preferences, there may be sub-areas of their interests which can be correlated to each other. For example, two friends may have similar interests concerning technology news, but different interests concerning political news. *Co-preference regularization* aims to learn such selective co-preferences. The motivation is to constrain users $\mathbf{x}$ and $\mathbf{z}$ who have similar or opposing preferences to be similar or opposite in the *same* latent item space relevant to item $\mathbf{y}$.

We use $\langle \cdot, \cdot \rangle_{\bullet}$ to denote a re-weighted inner product. The purpose of this inner product is to "mask" enforcement of latent space similarities or dissimilarities between users to be restricted to the same latent spaces as the co-preferred items. To this end, the objective component for *co-preference regularization* along with its expanded form is

Figure 1: The Facebook LinkR App showing two link recommendations to a user. The first link recommendation is from a non-friend and hence only shows the link description. The second link recommendation is from a friend and includes the friend's commentary on the link as well as the link description. Users have the option of liking or disliking each recommendation as well as providing feedback.

$$Obj_{cp} = \sum_{(\mathbf{x},\mathbf{z},\mathbf{y}) \in C} \frac{1}{2} (P_{\mathbf{x},\mathbf{z},\mathbf{y}} - \langle U\mathbf{x}, U\mathbf{z} \rangle_{V\mathbf{y}})^2 \tag{15}$$
$$= \sum_{(\mathbf{x},\mathbf{z},\mathbf{y}) \in C} \frac{1}{2} (P_{\mathbf{x},\mathbf{z},\mathbf{y}} - \mathbf{x}^T U^T \operatorname{diag}(V\mathbf{y}) U\mathbf{z})^2$$

We might also define a *social co-preference spectral regularization* approach, but our experiments so far have not indicated this works as well as the above objective.

In contrast to social regularization defined previously, co-preference regularization does not require knowledge of friendships or user interactions to determine co-preferences and hence can enforce regularization constraints between *all* users.

4. EVALUATION FRAMEWORK

Next we discuss our Facebook Link Recommendation App (LinkR), data collected, and our evaluation methodology.

4.1 Link Recommendation App on Facebook

To evaluate existing and newly proposed (S)CF methods discussed in this paper, we created a Facebook application (i.e., a Facebook "App") that recommends links to users every day, where the users may give their feedback on the links indicating whether they *liked* it or *disliked* it. Figure 1 shows our Facebook LinkR App as it appears to users.

The functionalities of the LinkR application on a daily basis are as follows:

1. Collect new data shared by users and their friends.
2. Initiate retraining of all active (S)CF link recommendation algorithms on the latest collected data. C and D from Section 2.1 are populated from all explicit likes and dislikes observed via the Facebook LinkR App and all "likes" observed via the Facebook interface.
3. Post-retrain, recommend three links to the users according to their assigned recommendation algorithm.
4. Collect feedback from the users on whether they liked or disliked the recommendations as well as any additional commentary the user wishes to provide.

Details of (S)CF link recommendation algorithms and user assignments will be discussed shortly; first we cover data collected by the LinkR App and used by the recommenders.

4.2 Facebook Data Collected

At its peak membership, 111 users had elected to install the Facebook App developed for this project. From this user base, we were able to gather data on over 37,626 users and 605,847 links in total by the end of the evaluation period.

4.2.1 User Data

Data that are collected and used to define the user feature vector $\mathbf{x}$ introduced in Section 2.1 for the LinkR Facebook App are defined as follows:

- $[\mathbf{x}_{id} = id] \in \{0,1\}, \forall id$: every unique Facebook ID (user) recorded in the App was assigned its own binary indicator in $\mathbf{x}$; all id indicators are mutually exclusive.
- $gender \in \{0(\text{female}), 1(\text{male})\}$.
- $age \in \mathbb{N}$.

We note that the indicator of friendships for $\mathbf{x}$ is stored in the $friends_{\mathbf{x}}$ set defined in Section 2.1 and used in various previous objective definitions, but not explicitly stored in $\mathbf{x}$.

4.2.2 Link Data

Data that are collected and used to define the item feature vector $\mathbf{y}$ introduced in Section 2.1 for the LinkR Facebook App are defined as follows:

- $[\mathbf{y}_{poster} = id] \in \{0,1\}, \forall id$: binary indicator feature for the id of the user who posted the link; all such binary indicator features are mutually exclusive.
- $[\mathbf{y}_{wall} = id] \in \{0,1\}, \forall id$: binary indicator feature for the id of the user on whose wall the link was posted; all such binary indicator features are mutually exclusive.
- Count of total link "likes" on Facebook.
- Count of total link shares on Facebook.
- Count of total link comments posted on Facebook.

4.2.3 Joint User and Link Data

The feature vector $\mathbf{f}_{\mathbf{x},\mathbf{y}}$ used in Sections 2.2 and 3.3.2 for the LinkR Facebook App is defined as the *concatenation* of $\mathbf{x}$, $\mathbf{y}$ (above) and the following additional social network information diffusion features:

- $\mathbf{z}$ liked $\mathbf{x} \in \{0,1\}, \forall \mathbf{z} \in friends_{\mathbf{x}}$: for every friend $\mathbf{z}$ of user $\mathbf{x}$, we have a binary information diffusion feature indicating whether user $\mathbf{z}$ liked item $\mathbf{y}$ (recall that $\mathbf{f}_{\mathbf{x},\mathbf{y}}$ is built w.r.t. a specific user $\mathbf{x}$ and item $\mathbf{y}$).

4.2.4 Interaction Data

We define *# interactions between user* $\mathbf{x}$ *and user* $\mathbf{z}$ in Section 2.1 as follows (all interactions are equally weighted):

1. Being friends.
2. Posting, liking, or commenting on an item (link, photo, video, or message) on a user's wall.
3. Being tagged together in the same photo or video.
4. Attending the same school or class, playing sports together, working together for a company or on a project.

4.3 Live Online Recommendation Trials

LinkR users were randomly and blindly assigned one of four algorithms in each of two live trials (algorithm details in Section 5). The rationale for assigning a single recommendation algorithm to a user was to obtain survey feedback from a user on their assigned algorithm to understand qualitative recommendation issues from a holistic algorithm perspective not necessarily obvious from quantitative measures alone.

Figure 1 shows the LinkR App interface, which displays both friend and non-friend link recommendations and allows the user to rate each link as like or dislike and provide optional feedback. LinkR recommended three links per day to avoid position bias and information overload. In early testing, users commented that many links older than two weeks were outdated or broken so LinkR only recommends links posted in the past two weeks that the user has not already posted, liked, or disliked. Based on first trial feedback, in the second trial we avoided recommendations of (i) non-English links and (ii) links lacking a text description.

5. EMPIRICAL RESULTS

Here we analyze results collected during two LinkR trials.[1]

5.1 First Trial

In our first trial, we evaluated four (S)CF algorithms:

1. **k-Nearest Neighbor (KNN)**: Section 2.3.1 (N=50)
2. **Support Vector Mach. (SVM)**: Section 2.2 (C=2)
3. **Matchbox (Mbox)**: Matchbox CF (λ=10^2, K=5)
$$Obj_{pmcf} + \lambda Obj_{ru} + \lambda Obj_{rv}$$
4. **Social Matchbox (Soc. Mbox)**: user feature socially regularized Matchbox SCF (λ_{rs}=10^{-3}, λ=10^2, K=5)
$$Obj_{pmcf} + \lambda_{rs} Obj_{rs} + \lambda Obj_{ru} + \lambda Obj_{rv}$$

Objectives for (Soc.) Mbox were given in Section 3 and optimized via gradient descent as in Appendix A. λ's for (Soc.) Mbox were tuned prior to the start of the trial by a systematic line (grid) search over 10^n for $n \in \{-5, -4, \ldots, 5\}$ to maximize accuracy on 25% held-out data, training on the other 75%. This was repeated for $K \in \{3, 5, 7, 10, 15, 20, 30\}$ to find the best K. N and C were tuned similarly via line search over $N \in \{1, 2, \ldots, 250\}$ and $C \in [10^{-4}, 10^4]$.

[1] All code used in these experiments is available at `http://code.google.com/p/social-recommendation/`. The conditions of our ethics approval #2011/142 from the Australian National University for conducting human trials on Facebook require our privacy policy (`http://dmm.anu.edu.au/linkr/website/pp.php`) to prohibit public sharing of data collected during these experiments.

Trial 1 – Aug. 25, 2011 to Oct. 13, 2011

	SMB	MB	SVM	KNN	Total
Users All	26	26	28	28	108
Users ≥ 10	13	9	13	5	40
Users ≥ 30	9	3	11	3	26
Ratings All	819	526	901	242	2508
Ratings ≥ 10	811	505	896	228	2440
Ratings ≥ 30	737	389	851	182	2159
Clicks All	383	245	413	218	1259

Trial 2 – Oct. 14, 2011 to Feb. 10, 2012

	SMB	Sp.MB	Sp.CP	SHyb.	Total
Users All	27	27	29	28	111
Users ≥ 10	15	11	8	12	46
Users ≥ 30	12	9	5	10	36
Ratings All	1434	882	879	614	3809
Ratings ≥ 10	1411	878	863	602	3754
Ratings ≥ 30	1348	850	802	570	3570
Clicks All	553	320	278	199	1350

Table 1: Number of users assigned per algorithm in the first and second trials. ≥ 10 (≥ 30) indicates data for the subset of users with at least 10 (30) ratings. Data from non-rating users (and their friends) was important for the performance of all algorithms.

First trial details are provided in Table 1 (top); algorithm performance is shown in Figure 2 (top). 95% binomial proportion confidence intervals (using the asymmetrical Wilson score interval method [18]) are shown for the combined data for all users of each algorithm. While user usage varies, this method of combining all user data for CF system performance analysis is a standard evaluation approach for CF systems; notably, RMSE over *all combined ordinal user ratings* was used for determining the winner of the Netflix prize[2].

Except for Mbox, most algorithms performed comparably on non-friend recommendations. For friend recommendations, Soc. Mbox performed best, where it appears that social regularization helped it effectively find latent representations of friends with similar interests; it performed better than Mbox without social regularization and SVM which attempted to explicitly model information diffusion from friends. While KNN had low usage, it would be statistically unlikely to match Soc. Mbox's performance on friend recommendations — to do so if the amount of KNN data were doubled, nearly all new ratings would have to be "like".

5.2 Second Trial

For the second trial, Soc. Mbox was included as a baseline since it was the top performer from the first trial. The remaining three algorithms were all relatively orthogonal Soc. Mbox extensions or variants based on the *three novel objective functions* defined in Section 3.3 (all used $K = 5$):

1. **Social Matchbox (Soc. Mbox)**: unchanged
2. **Spectral Matchbox(Sp. Mbox)**: (λ_{rss}=10^{-3}, λ=10)
$$Obj_{pmcf} + \lambda_{rss} Obj_{rss} + \lambda Obj_{ru} + \lambda Obj_{rv}$$
3. **Social Hybrid (Soc. Hybrid)**: (λ_{rs}=10^{-3}; λ=10^4)
$$Obj_{phy} + \lambda_{rs} Obj_{rs} + \lambda Obj_{ru} + \lambda Obj_{rv} + \lambda Obj_{rw}$$
4. **Spectral Copreference(Sp. CP)**: (λ_{cp}=10^{-4}; λ=10)
$$Obj_{pmcf} + \lambda_{cp} Obj_{cp} + \lambda Obj_{ru} + \lambda Obj_{rv}$$

[2]`http://www.netflixprize.com/`

All objectives are defined in Section 3 and optimized via gradient descent as in Appendix A. λ and K parameters for Soc. Mbox were left unchanged from the first trial in order to use it as a fixed comparative baseline across both trials. We chose $K = 5$ for all other recommendation algorithms to provide a controlled comparison with Soc. Mbox. With $K = 5$ fixed, all other λ parameters were tuned prior to the start of the second trial using the same systematic grid search methodology as described for the first trial.

Second trial details are provided in Table 1 (bottom); on the start of the second trial, users were notified that they would be randomly assigned to new algorithms and encouraged to re-engage with the LinkR App if they had not been using it. Two email reminders were sent during the trial.

Second trial results using the same evaluation methodology as described for the first trial are shown in Figure 2 (bottom). Following are key observations from this trial:

- Soc. Mbox did not perform as well in the second trial as it had in the first trial. We hypothesize that Soc. Mbox may have performed better if λ_{rs} and λ were better tuned for the amount of data in the second trial. To evaluate this hypothesis, in the following table, we show the accuracy of Soc. Mbox at predicting link likes/dislikes on second trial data, training on 75% of the data and testing on the remaining 25%:

	λ_{rs}=10^{-1}	=10^{-2}	=10^{-3}	=10^{-4}	=10^{-5}
λ=10^1	0.325	0.307	0.301	0.437	0.540
λ=10^2	0.306	0.301	**0.300**	0.295	0.300
λ=10^3	0.297	0.301	0.307	0.300	0.301

  Here we show results for the second trial parameter settings in bold that achieve prediction accuracy of 0.300; however if both λ and λ_{rs} are reduced, we note a substantial improvement to 0.436 and 0.540. It appears less regularization of U and V is needed in the presence of the additional data in the second trial and the accuracy differential here suggests the need to periodically re-tune parameters to maintain optimal performance.

- Spec. Mbox performed exceedingly well in the second trial and this suggests that our novel feature-based *spectral* social regularization is likely a better method of regularization for U than the social regularization of Soc. Mbox. Even when considering the best achievable performance of Soc. Mbox (54%), this would still fall well below Spec. Mbox's impressive 65%.

- Soc. Hybrid statistically ties Spec. Mbox at recommending friend links (where it can learn user-to-user information diffusion), but performs less well on non-friend links (where there is no such diffusion). These results suggest that the space- and compution-efficient low-dimensional learning of Spec. Mbox can recommend friend links just as well as Soc. Hybrid's modeling of explicit user-to-user information diffusion.

- Given that each LinkR user shared co-preferences with 535.1 other users on average (indicating that this data is far from sparse), it would appear from the performance of Spec. CP that co-preferences serve as a somewhat noisy social regularization constraint compared to social regularization based on interactions between friends as exemplified by Spec. Mbox's performance.

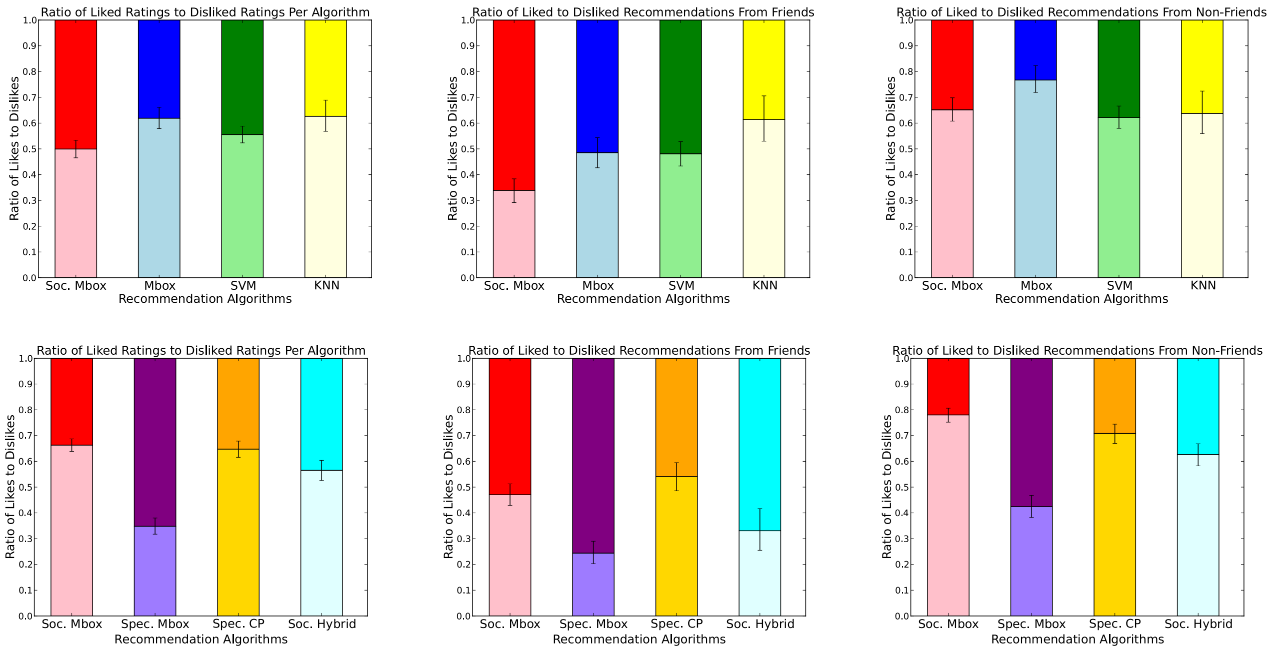


Figure 2: Stacked bar graphs of online results for the first (top) and second (bottom) user trials. The fraction of likes is displayed above the fraction of dislikes. Results are also broken down by link type: (left) all, (center) friend only, (right) non-friend only. 95% binomial proportion confidence intervals are shown.

5.3 User Behavior and Data Analysis

Overall, users had a general bias to like links recommended by friends more than non-friends; importantly, we note that users could see the names and comments of friends whose links were recommended, indicating the importance of context in recommendation. Next we briefly analyze other user behavior and data collected during both trials of the LinkR App that can be helpful in building future SCF systems.

5.3.1 *Click evidence*

In Figure 3(a), we observe the ratings of links that users clicked on. The most important thing we notice here is that even though users clicked on a link, they were somewhat likely to rate it as a dislike (roughly $\frac{2}{3}$ like to $\frac{1}{3}$ dislike).

One might hypothesize that perhaps users clicked on links more often with no description to find out what they were and most often disliked them — this might explain the high number of dislikes for clicked links. However, examining both Figures 3(b) and (c), we observe that whether a description was present had a relatively minor impact on whether a link was clicked or liked, so we cannot infer that the disliked links were simply the ones lacking a description.

Then the insight from this analysis is extremely important for SCF recommendation design because it states that click data is a somewhat weak indicator of likes and that even if one could predict clicks with perfect accuracy, this would only yield roughly $\frac{2}{3}$ accuracy for likes prediction.

5.3.2 *Impact of Popularity*

In Figures 3(d) and (e) we analyze the impact of global link popularity (in terms of total shares on Facebook) on how much LinkR App users liked a link. The trend is clear for both friend (d) and non-friend (e) links: users tend to like the most popular (top quartile) links the least compared to all other quartiles. In general, users tended to most prefer links that were somewhat popular (middle quartiles). From this we can infer that while the most popular links may be liked by the most people, they are not liked by everyone on average; this suggests that link popularity should not be weighted too heavily in determining link recommendations.

Individual Link Comments

Comment Type	#	%
not interested	88	36.5%
wrong language	37	15.4%
really liked it!	35	14.5%
bad YouTube	25	10.4%
seen it already	25	10.4%
problem / dead	20	8.3%
outdated	7	2.9%
miscellaneous	4	1.7%

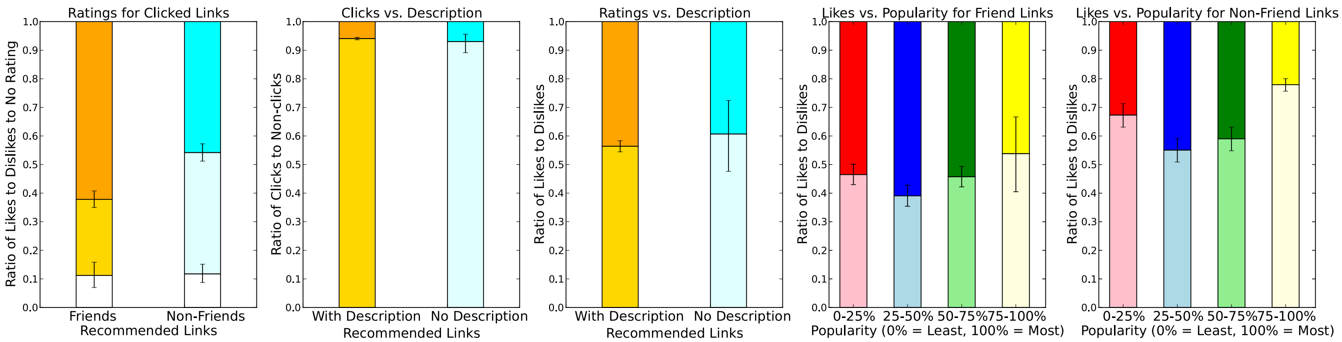
User Survey Comments
want more control over recommendations made (music, blogs, news)
want option to see > 3 recommendations
links need description / context or explanation of recommendation
more variety, diversity

Table 2: Individual link comments (aggregated) and notable user survey requests (paraphrased).

5.3.3 *Link and Survey Comments*

We collected individual link recommendation comments in the LinkR App as shown in Figure 1 and we also ran a user survey toward the end of both trials to collect qualitative feedback on the overall LinkR user experience. Due to space limitations, we briefly summarize this data in Table 2.

Table 2 (left) shows link comments classified into general classes and ranked by frequency. Users were easily annoyed (i) if they could not read the language of the link — this issue was addressed with a language filter in the second trial, (ii) if a YouTube or other link was inaccessible — YouTube links were some of the most popular links on Facebook and were also frequently removed for copyright violations, (iii) if a user had seen a similar link topic already — e.g., users quickly



(a) (b) (c) (d) (e)

Figure 3: Stacked bar graphs for rating and click data collected during both trials. The fraction of likes (or clicks) is displayed above the fraction of dislikes (or non-clicks) – and above the fraction of not-rated links for (a). 95% binomial proportion confidence intervals are shown. (a) ratings for clicked links, (b) clicks vs. description, (c) ratings vs. description, (d) ratings vs. quartile of popularity for friends, and (e) non-friends.

got tired of repeated news links on Steve Jobs death during the second trial even though the links had different content, or (iv) if links were deemed to be outdated — e.g., a news article had been superceded by more recent information. On the other hand, if a user was pleased with a recommendation that they were not otherwise aware of, they often indicated this. In Table 2 (right) we show four notable user survey comments requesting modifications to the LinkR experience.

6. CONCLUSIONS

In this paper, we evaluated existing algorithms and proposed new algorithms for social collaborative filtering via the task of link recommendation on Facebook. Importantly, we outlined three main deficiencies in existing social collaborative filtering (SCF) matrix factorization (MF) techniques and proposed novel objective functions that (a) extended existing social regularization SCF approaches to incorporate user features by drawing on ideas from Matchbox [17], (b) modeled direct user-to-user information diffusion, and (c) modeled restricted common interests among all users (friend and non-friend) with social co-preference regularization.

We evaluated existing baselines and then evaluated algorithms based on optimization of these new objectives in Section 5 via live online user trials taking place over five months with over 100 Facebook App users and data from over 37,000 Facebook users. Results show that our novel feature-based social spectral regularization extension of Matchbox achieves an overall correct "likes" prediction rate of 65% — performance far exceeding that of all other algorithms trialed.

User feedback has opened up many new possibilities for further improving the SCF user experience. Future work can include: providing explanations for recommended content; incorporating *genre* features to provide a fine-grained model of user preference among different content; and enforcing diversity among recommendations to prevent redundancy.

Acknowledgements

This work was partially funded by a Google Research Award. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

7. REFERENCES

[1] M. Balabanović and Y. Shoham. Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40:66–72, March 1997.

[2] R. M. Bell and Y. Koren. Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In *ICDM-07*, 2007.

[3] J. Brown and P. Reinegen. Social ties and word-of-mouth referral behavior. *Journal of Consumer Research*, 1(3):350–362, 1987.

[4] C.-C. Chang and C.-J. Lin. *LIBSVM: a Library for Support Vector Machines*, 2001.

[5] C. Cortes and V. Vapnik. Support-vector networks. In *Machine Learning*, pages 273–297, 1995.

[6] P. Cui, F. Wang, S. Liu, M. Ou, and S. Yang. Who should share what? item-level social influence prediction for users and posts ranking. In *International ACM SIGIR Conference (SIGIR)*, 2011.

[7] K. Lang. NewsWeeder: Learning to filter netnews. In *12th International Conference on Machine Learning ICML-95*, pages 331–339, 1995.

[8] W.-J. Li and D.-Y. Yeung. Relation regularized matrix factorization. In *IJCAI-09*, 2009.

[9] D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1):503–528, Aug 1989.

[10] H. Ma, I. King, and M. R. Lyu. Learning to recommend with social trust ensemble. In *SIGIR-09*, 2009.

[11] H. Ma, H. Yang, M. R. Lyu, and I. King. Sorec: Social recommendation using probabilistic matrix factorization. In *CIKM-08*, 2008.

[12] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *WSDM-11*, 2011.

[13] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS 14*, 2001.

[14] K. B. Petersen and M. S. Pedersen. The matrix cookbook, 2008.

[15] P. Resnick and H. R. Varian. Recommender systems. *Communications of the ACM*, 40:56–58, March 1997.

[16] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *NIPS 20*, 2008.
[17] D. H. Stern, R. Herbrich, and T. Graepel. Matchbox: large scale online bayesian recommendations. In *WWW-09*, pages 111–120, 2009.
[18] E. B. Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Assoc. (JASA)*, 22:209–212, 1927.
[19] S. H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha. Like like alike: Joint friendship and interest propagation in social networks. In *WWW-11*, 2011.

APPENDIX

A. GRADIENT-BASED OPTIMIZATION

We seek to optimize sums of the objectives in Section 3 and will use gradient descent for this purpose.

For the overall objective, the partial derivative w.r.t. parameters $\mathbf{a}$ are as follows:

$$\frac{\partial}{\partial \mathbf{a}} Obj = \frac{\partial}{\partial \mathbf{a}} \sum_i \lambda_i Obj_i = \sum_i \lambda_i \frac{\partial}{\partial \mathbf{a}} Obj_i \qquad (16)$$

Anywhere a sigmoidal transform occurs $\sigma(o[\cdot])$, we can easily calculate the partial derivatives as follows

$$\frac{\partial}{\partial \mathbf{a}} \sigma(o[\cdot]) = \sigma(o[\cdot])(1-\sigma(o[\cdot]))\frac{\partial}{\partial \mathbf{a}} o[\cdot]. \qquad (17)$$

Hence anytime a $[\sigma(o[\cdot])]$ is optionally introduced in place of $o[\cdot]$, we simply insert $[\sigma(o[\cdot])(1-\sigma(o[\cdot]))]$ in the corresponding derivatives below.

Because most objectives below are not convex in U, V, or $\mathbf{w}$, we apply an *alternating gradient descent* approach [16]. In short, we take derivatives of U, V, and $\mathbf{w}$ in turn while holding the others constant. Then we apply gradient descent in a round-robin fashion until we've reached local minima for all parameters; for gradient descent on one of U, V, or $\mathbf{w}$ with the others held constant, we apply the L-BFGS optimizer [9] with derivatives defined below.

Before we proceed to our objective gradients, we define abbreviations for three useful vectors:

$$\begin{aligned} \mathbf{s} &= U\mathbf{x} & \mathbf{s}_k &= (U\mathbf{x})_k;\ k = 1\dots K \\ \mathbf{t} &= V\mathbf{y} & \mathbf{t}_k &= (V\mathbf{y})_k;\ k = 1\dots K \\ \mathbf{r} &= U\mathbf{z} & \mathbf{r}_k &= (U\mathbf{z})_k;\ k = 1\dots K \end{aligned}$$

All matrix derivatives used for the objectives below can be verified in [14].

$$\begin{aligned} \frac{\partial}{\partial U} Obj_{pmcf} &= \frac{\partial}{\partial U} \sum_{(\mathbf{x},\mathbf{y})\in D} \frac{1}{2}\Bigg(\underbrace{(R_{\mathbf{x},\mathbf{y}} - [\sigma]\overbrace{x^T U^T V\mathbf{y})}^{o_{\mathbf{x},\mathbf{y}}}}_{\delta_{\mathbf{x},\mathbf{y}}}\Bigg)^2 \\ &= -\sum_{(\mathbf{x},\mathbf{y})\in D} \delta_{\mathbf{x},\mathbf{y}}[\sigma(o_{\mathbf{x},\mathbf{y}})(1-\sigma(o_{\mathbf{x},\mathbf{y}}))]\mathbf{t}\mathbf{x}^T \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial V} Obj_{pmcf} &= \frac{\partial}{\partial V} \sum_{(\mathbf{x},\mathbf{y})\in D} \frac{1}{2}\Bigg(\underbrace{(R_{\mathbf{x},\mathbf{y}} - [\sigma]\overbrace{x^T U^T V\mathbf{y})}^{o_{\mathbf{x},\mathbf{y}}}}_{\delta_{\mathbf{x},\mathbf{y}}}\Bigg)^2 \\ &= -\sum_{(\mathbf{x},\mathbf{y})\in D} \delta_{\mathbf{x},\mathbf{y}}[\sigma(o_{\mathbf{x},\mathbf{y}})(1-\sigma(o_{\mathbf{x},\mathbf{y}}))]\mathbf{s}\mathbf{y}^T \end{aligned}$$

$$\frac{\partial}{\partial U} Obj_{ru} = \frac{\partial}{\partial U}\frac{1}{2}\operatorname{tr}(U^T U) = U \qquad \frac{\partial}{\partial V} Obj_{rv} = V$$
$$\frac{\partial}{\partial \mathbf{w}} Obj_{rw} = \frac{\partial}{\partial \mathbf{w}}\frac{1}{2}\mathbf{w}^T\mathbf{w} = \mathbf{w}$$

$$\begin{aligned} \frac{\partial}{\partial U} Obj_{rs} &= \frac{\partial}{\partial U} \sum_{\mathbf{x}} \sum_{\mathbf{z}\in friends(\mathbf{x})} \frac{1}{2}\Bigg(\underbrace{S_{\mathbf{x},\mathbf{z}} - \mathbf{x}^T U^T U\mathbf{z}}_{\delta_{\mathbf{x},\mathbf{y}}}\Bigg)^2 \\ &= -\sum_{\mathbf{x}} \sum_{\mathbf{z}\in friends(\mathbf{x})} \delta_{\mathbf{x},\mathbf{y}} U(\mathbf{x}\mathbf{z}^T + \mathbf{z}\mathbf{x}^T) \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial U} Obj_{rss} &= \frac{\partial}{\partial U} \sum_{\mathbf{x}} \sum_{\mathbf{z}\in friends(\mathbf{x})} \frac{1}{2} S^+_{\mathbf{x},\mathbf{z}}(\mathbf{x}-\mathbf{z})^T U^T U(\mathbf{x}-\mathbf{z}) \\ &= \sum_{\mathbf{x}} \sum_{\mathbf{z}\in friends(\mathbf{x})} S^+_{\mathbf{x},\mathbf{z}} U(\mathbf{x}-\mathbf{z})(\mathbf{x}-\mathbf{z})^T \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial \mathbf{w}} Obj_{phy} &= \frac{\partial}{\partial \mathbf{w}} \sum_{(\mathbf{x},\mathbf{y})\in D} \frac{1}{2}\Bigg(\underbrace{R_{\mathbf{x},\mathbf{y}} - [\sigma]\overbrace{\mathbf{w}^T\mathbf{f}_{\mathbf{x},\mathbf{y}}}^{o^1_{\mathbf{x},\mathbf{y}}} - [\sigma]\mathbf{x}^T U^T V\mathbf{y}}_{\delta_{\mathbf{x},\mathbf{y}}}\Bigg)^2 \\ &= -\sum_{(\mathbf{x},\mathbf{y})\in D} \delta_{\mathbf{x},\mathbf{y}}[\sigma(o^1_{\mathbf{x},\mathbf{y}})(1-\sigma(o^1_{\mathbf{x},\mathbf{y}}))]\mathbf{f}_{\mathbf{x},\mathbf{y}} \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial U} Obj_{phy} &= \frac{\partial}{\partial U} \sum_{(\mathbf{x},\mathbf{y})\in D} \frac{1}{2}\Bigg(\underbrace{R_{\mathbf{x},\mathbf{y}} - [\sigma]\mathbf{w}^T\mathbf{f}_{\mathbf{x},\mathbf{y}} - [\sigma]\overbrace{\mathbf{x}^T U^T V\mathbf{y}}^{o^2_{\mathbf{x},\mathbf{y}}}}_{\delta_{\mathbf{x},\mathbf{y}}}\Bigg)^2 \\ &= -\sum_{(\mathbf{x},\mathbf{y})\in D} \delta_{\mathbf{x},\mathbf{y}}[\sigma(o^2_{\mathbf{x},\mathbf{y}})(1-\sigma(o^2_{\mathbf{x},\mathbf{y}}))]\mathbf{t}\mathbf{x}^T \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial V} Obj_{phy} &= \frac{\partial}{\partial V} \sum_{(\mathbf{x},\mathbf{y})\in D} \frac{1}{2}\Bigg(\underbrace{R_{\mathbf{x},\mathbf{y}} - [\sigma]\mathbf{w}^T\mathbf{f}_{\mathbf{x},\mathbf{y}} - [\sigma]\overbrace{\mathbf{x}^T U^T V\mathbf{y}}^{o^2_{\mathbf{x},\mathbf{y}}}}_{\delta_{\mathbf{x},\mathbf{y}}}\Bigg)^2 \\ &= -\sum_{(\mathbf{x},\mathbf{y})\in D} \delta_{\mathbf{x},\mathbf{y}}[\sigma(o^2_{\mathbf{x},\mathbf{y}})(1-\sigma(o^2_{\mathbf{x},\mathbf{y}}))]\mathbf{s}\mathbf{y}^T \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial U} Obj_{cp} &= \frac{\partial}{\partial U} \sum_{(\mathbf{x},\mathbf{z},\mathbf{y})\in C} \frac{1}{2}\Bigg(\underbrace{P_{\mathbf{x},\mathbf{z},\mathbf{y}} - \mathbf{x}^T U^T \operatorname{diag}(V\mathbf{y})U\mathbf{z}}_{\delta_{\mathbf{x},\mathbf{z},\mathbf{y}}}\Bigg)^2 \\ &= -\sum_{(\mathbf{x},\mathbf{z},\mathbf{y})\in C} \delta_{\mathbf{x},\mathbf{z},\mathbf{y}} \operatorname{diag}(V\mathbf{y})U(\mathbf{x}\mathbf{z}^T + \mathbf{z}\mathbf{x}^T) \end{aligned}$$

In the following, $\circ$ is the Hadamard elementwise product:

$$\begin{aligned} \frac{\partial}{\partial V} Obj_{cp} &= \frac{\partial}{\partial V} \sum_{(\mathbf{x},\mathbf{z},\mathbf{y})\in C} \frac{1}{2}(P_{\mathbf{x},\mathbf{z},\mathbf{y}} - \mathbf{x}^T U^T \operatorname{diag}(V\mathbf{y})U\mathbf{z})^2 \\ &= \frac{\partial}{\partial V} \sum_{(\mathbf{x},\mathbf{z},\mathbf{y})\in C} \frac{1}{2}\Bigg(\underbrace{P_{\mathbf{x},\mathbf{z},\mathbf{y}} - (\overbrace{U\mathbf{x}}^{\mathbf{s}} \circ \overbrace{U\mathbf{z}}^{\mathbf{r}})^T V\mathbf{y}}_{\delta_{\mathbf{x},\mathbf{z},\mathbf{y}}}\Bigg)^2 \\ &= -\sum_{(\mathbf{x},\mathbf{z},\mathbf{y})\in C} \delta_{\mathbf{x},\mathbf{z},\mathbf{y}}(\mathbf{s}\circ\mathbf{r})\mathbf{y}^T \end{aligned}$$