Local Optimality of User Choices and Collaborative Competitive Filtering

Shuang Hong Yang
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
shy@gatech.edu

October 5, 2010

Abstract

We describe a novel framework for learning recommender models for recommendation systems, which views user-system-item interactions as an *opportunity give-and-take* process, and encodes both "collaboration" and "competition" mechanisms underlying the interaction. The proposed framework leverages the latent factor models of collaborative filtering to encode "collaboration" (via factor sharing); and in the meanwhile, it utilizes a type of objectives that implies *local optimality of user choices* to encode "competition". Specifically, it takes into account both the *revenue* and the *opportunity cost* of each user decision; and, by optimizing a new objective that are analogous to the *economic profit*, it encourages that every opportunity being taken by a user be locally the best among the opportunities being offered to him/her. Such competition among candidates opportunities imposes stronger supervision and in turn leads to better generalization to unseen interactions. Empirical results indicates that the *collaborative-competitive filtering* (CCF) approaches improve dramatically recommendation performance compared with traditional collaborative filtering models (e.g., nDCG score is boosted from 0.14 to 0.71 on Yahoo! Pulse data, which is a huge 400% improvement).

We also discuss a Bayesian Generative model that enables joint learning of explore-exploit strategy and factorization recommender models. We call for attentions from recommendation industry to test this model on real recommendation system.

1 Introduction

Recommendation systems have become a core component for today's personalized online business. Instead of listing all the items [1] in a massive menu as shown to the users in traditional offline businesses, a recommendation system adapts the services according to the interest of each individual user by presenting to the user with a small subset of items that would potentially interest her. Such ability of interest targeting (i.e. match user with interesting items and item with interested users) of recommendation system has contributed to the success of many eCommerce companies (e.g. Amazon, Netflix, Pandora, Facebook)

[1] Example items in recommender systems: retailing products, local shops, movies, games, music, books, video, News articles, advertisements, Web pages, friends, experts, etc.

and is now the key to almost all kinds of long tail businesses (Brynjolfsson et al., 2003, 2007).

However, modeling such subtle aspect as user-to-item interest is not a trivial task – it requires predicting the response of an user-item interaction before it actually happens. Existing systems resort unanimously to collaborative filtering (CF) techniques, which learn to predict future interactions by collaboratively uncovering historical records of user-item responses in such a way that "related" items get similar responses from "related users"(Sarwar et al., 2001; McLaughlin & Herlocker, 2004; Salakhutdinov & Mnih, 2008; Agarwal & Chen, 2009; Chen et al., 2009; Koren et al., 2009). The rationale is as follows: usually the observed historical interactions are extremely noisy and sparse (often less than 1%); by using the "collaboration", among users or among items or both, CF greatly alleviates the issue of data sparseness and in turn makes reliable prediction by aggregating the interaction evidences across different items/users to discover meaningful interaction patterns.

The current state-of-art approaches to collaborative filtering rely on a regression-based framework, either approximating the click-through-rate (CTR) by minimizing the mean-square-error (MSE) loss or maximizing the likelihood of user-item interactions based on the Bernoulli cross-entropy loss. Yet, the performance of such approaches has already arrived at a limit and become the bottleneck for higher-quality recommendation systems.

In this paper, we argue that, besides "collaboration", the "competition" among items is another fundamental mechanism underlying user-item interactions that are worthwhile exploiting for predictive recommendation. We describe a new perspective for user choice in recommendation systems. In particular, we view the user-system-item interaction as an *opportunity give-and-take* process:

1) a user inquires the system (e.g. visits a movie recommendation website);
2) the system offers a set of (personalized) opportunities (e.g. recommends a small list of movies that are potentially interesting to the user);
3) the user chooses one or more from these offers and takes actions accordingly (e.g. click a link, rent a movie, view a News article, purchase a product).

We develop a theory of "local optimality" for user choices, which states that a user is most likely to take the opportunity that is locally optimal among those opportunities being offered to her. This theory imposes a "competition" over the items that the system offers to a user.

We establish a novel framework for recommender learning based on this thoery. The proposed *collaborative competitive filtering* (CCF) exploits both the "collaboration" and "competition" mechanisms. In particular, it leverages the multiplicative latent factor model of collaborative filtering to capture "collaboration" among users and items; but instead of estimating CTR or likelihood by optimizing MSE or cross entropy as in CF, CCF takes into account both the *revenue* and the *opportunity cost* of each user decision; and, by optimizing a new objective that are analogous to the *economic profit* in Microeconomics, it encourages that every opportunity being taken by a user be locally the best among the opportunities being offered to her. Such competition among opportunities imposes stronger supervision and in turn leads to better generalization to unseen interactions.

From a machine learning viewpoint, our proposed framework is a hybrid of *local* and *global* learning, where a global model (e.g., factorization model) is learned by optimizing a local loss function (e.g., the profit loss function). In CCF, the global latent factor model encodes collaboration among users and items (via factor sharing), whereas the local contextualized loss function captures the competition among items within each offer contexts.

The local loss function imposes stronger supervision/constraint than global ones (e.g. MSE, logistic) and in turn leads to more predictive latent factor models.

Empirical results on Yahoo! Pulse data indicates that our *collaborative-competitive filtering* (CCF) approaches improve dramatically the recommendation performance compared with traditional collaborative filtering models (e.g., nDCG score is boosted from 0.14 to 0.71, a huge 400% improvement).

We also propose a Bayesian model that allows joint learning of explore-exploit strategy and collaborative-competitive filtering. We call for attentions from recommendation industry to test this model on real recommender systems.

2 Problem formulation

Consider the interaction in a recommendation system: where we have a set of users $\mathcal{U} = \{u = 1, 2, \ldots, U\}$ and a set of items $\mathcal{I} = \{i = 1, 2, \ldots, I\}$; for a given user u, the system recommends a small subset of items $\mathcal{O} = \{i_1, \ldots, i_l\}$ to display to the user, and u in turn chooses a subset (possibly empty) $\mathcal{D} \subset \mathcal{O}$ from $\mathcal{O}$ and takes actions accordingly (e.g. buys one of the recommended products). To assist building such an system, we have a trace of historical user-system-item interactions in the form of $\{(u_t, \mathcal{O}_t, \mathcal{D}_t)\}$, where t is the index of an interaction.

Hereafter, we refer to $\mathcal{U}$ as user space, $\mathcal{I}$ as item (Opportunity) space, $\mathcal{O}_t$ as offer set, and $\mathcal{D}_t$ as decision set.

We consider the latent factor CF models for recommender learning. Latent factor models embeds both user and item into the same space (e.g., Euclidean, simplex) via latent factors, θ_u for each user u and θ_i for each item i, then recommendation for a new inquiry from user u is done by ranking items based on $f(\theta_u, \theta_i)$ and choosing the top-ranked ones. Usually, a multiplicative ranking function is used: $f(\theta_u, \theta_i) = \theta_u^\top \theta_i$.

3 The state-of-the-art of collaborative filtering

Existing CF approaches learn latent factors, θ_u and θ_i, in a regression based framework.

Bilinear Ridge Regression (Koren, 2008; Agarwal & Chen, 2009) The most popular learning formulation is to minimize the estimation error of click-through rate (CTR) in the sense of least square error:

$$\theta_u, \theta_i \quad = \quad \arg\min ||\rho_{ui} - \theta_u^\top \theta_i||^2 + c||\theta||^2, \tag{1}$$

where the CTR $\rho_{ui} = \frac{\#\{i \in \mathcal{D}_t | u_t = u\}}{\#\{i \in \mathcal{O}_t | u_t = u\}}$, $\#\{i \in \mathcal{O}_t | u_t = u\}$ is the offer frequency (i.e., the total frequency of item i being recommended to u), and $\#\{i \in \mathcal{D}_t | u_t = u\}$ the decision frequency (i.e., the frequency i being chosen (i.e., clicked) by u).

Bilinear Logistic Regression (Miller et al., 2009; Agarwal & Chen, 2009) Another popular formulation for CF is to optimize the Bernoulli cross-entropy loss function instead of l_2:

$$\theta_u, \theta_i \quad = \quad \arg\max \left(y_{ui} \log \frac{\exp(\theta_u^\top \theta_i)}{\exp(\theta_u^\top \theta_i) + 1} \right) + c||\theta||^2 \tag{2}$$

where the binary indicator variable $y_{ui} = \delta(i \in \mathcal{D}_u)$. This is essentially a logistic regression model learned on one-class (i.e., positive only) examples.

Problems with Regression-based Models The regression based approaches have gained a lot of success and become the current state-of-the-art of collaborative filtering. However, such approaches are lacking in several aspects:

- Data sparseness (i.e. the interactions are highly incomplete such that a vast majority of responses are not observed) is a key issue for recommender learning. To avoid overfitting, stronger supervision or constraints than MSE or cross-entropy is needed for more predictive recommender models.
- The ridge regression model is dominated by examples with large CTR-estimation residues. It often performs poorly because the historic response data are extremely sparse and noisy.
- The logistic regression model essentially approaches the task as binary classification. However, with examples missing from one class, logistic regression classifiers generally perform terribly because the classifier is biased toward one class while being uncurbed at the other.
- Both RR and LR only capture one aspect of the interactions, i.e., either $\mathcal{O}$ or $\mathcal{D}$ or the ratio. And both are global learning methods (i.e., global factor model learned globally with global loss), neglecting the contextual information (i.e., the context $\mathcal{O}_t$ where the decision is made), which is invaluable for learning predictive recommender models.

4 Collaborative competitive filtering

We argue that the contexts in which user's decisions are made should be taken into account in CF learning. The rationale is that, in practice, even the same user u could make different decisions when facing different contexts $\mathcal{O}$. For instance, an offer (e.g., item) would not have been chosen if it were not presented to the user at the first place; likewise, user might choose to accept another offer if the context $\mathcal{O}$ changes such that a better offer (e.g., a more interesting item) is presented to her.

We present the *local optimality of user choice*, which implies a mechanism of "competition" among items in each context $\mathcal{O}$ such that each user choice is locally optimal in that context. We propose two formulations for *collaborative competitive filtering* (CCF) in this section, which are justified in the next section.

LOCAL OPTIMALITY OF USER CHOICE *Given a set of offers, $\mathcal{O}$, a user u always chooses the offer that is locally optimal in the context of $\mathcal{O}$, i.e.: $i^* = \arg\max\{r_{ui} : i \in \mathcal{O}\}$, where r_{ui} is the revenue that u earns by choosing i.*

We assume a multiplicative model for revenue, $r_{ui} = \theta_u^\top \theta_i$, in order to learn a latent factor recommender model from user decision traces. The aforementioned theory induces an local-optimality constraint which could be translated into an ideal loss function for latent factor learning:

$$\forall\, i^* \in \mathcal{D}: \quad \theta_u^\top \theta_i^* \geqslant \max\{\theta_u^\top \theta_i | i \in \mathcal{O} \setminus \mathcal{D}\}), \tag{3}$$

This loss function is, however, computationally intractable as the optimization is provably NP-hard. To this end, we propose two formulations based on two surrogate loss functions.

Contextualized Softmax Model Our first formulation is to use *softmax* function as a surrogate of *max*. Particularly, we assume the following model for the probability of an offer being taken by user u in the context of an offer set $\mathcal{O}$:

$$p(i^*|u,\mathcal{O}) = \frac{\exp(r_{ui^*})}{\sum_{i\in\mathcal{O}}\exp(r_{ui})}. \tag{4}$$

We learn the latent factors by the following MLE estimation:

$$\max \quad \sum_{t,i^*\in\mathcal{D}_t} \log\left(\frac{\exp(\theta_u^\top\theta_{i^*})}{\sum_{i\in\mathcal{O}_t}\exp(\theta_u^\top\theta_i)}\right) + c||\theta||^2. \tag{5}$$

This is a convex optimization problem and could be solved efficiently and globally: for example, we could use stochastic gradient descent for sequentially learning when the data set is large.

This model has an interesting connection with the logistic regression model. Essentially, our model uses a local loss function, which could be seen as logistic regression with context-aware local partitions:

$$\theta_u, \theta_i \quad = \quad \arg\max\left(y_{ui}\log\frac{\exp(\theta_u^\top\theta_i)}{\exp(\theta_u^\top\theta_i) + \sum_{i'\in\mathcal{O}_t\backslash i}\exp(\theta_u^\top\theta_{i'})}\right) + c||\theta||^2. \tag{6}$$

Note that, in contrast to the classical logistic regression model Eqn(2) that uses a constant 1 in the partition, our softmax model uses a local partition $\sum_{i'\in\mathcal{O}_t\backslash i}\exp(\theta_u^\top\theta_{i'})$ that is dependent on the context $\mathcal{O}_t$. From another perspective, our model uses the context as background and maximizes the probability that every choice of the user is locally the winer of the context set. Roughly speaking, it uses (implicitly) the non-choices in the context set virtually as negative examples.

Contextualized Hinge Model Our second formulation views the task as a pairwise preference learning and uses the non-choices averagely as an negative example.

$$\begin{aligned} &\min\sum_t\sum_i \xi_i + c||\theta||^2 \\ &\text{subjectto}: \quad \theta_u^\top\theta_i - \theta_u^\top\theta_{-i} \geqslant 1 + \xi_i,\ \forall\, i\in\mathcal{D}_t \end{aligned} \tag{7}$$

where $\theta_u^\top\theta_{-i} = \frac{1}{l-1}\sum_{i'\in\mathcal{O}_t\backslash i}\theta_u^\top\theta_{i'}$ is the average revenue of the non-choices, $l = |\mathcal{O}_t|$. This formulation learn latent factors by maximizing the marginal utility between user choice and the average of non-choices. This is essentially a contextualized bilinear RankSVM model (Herbrich et al., 1999), and could be solved by existing RankSVM QP solvers.

5 The opportunity give-and-take (GAT) process

The user-system-item interaction in recommendation system can be viewed as an instance of the *opportunity give-and-take* (GAT) process. In this section, we give brief definition to the GAT process and draw justification to our proposed framework.

DEFINITION [GAT]: *An opportunity give-and-take process is a process of interactions among an agent u, a system S and a set of opportunities $\mathcal{I}$; at an interaction t:*

- *u is given a set of opportunities $O_t \subset \mathcal{I}$ by S; other opportunities that are not in $\mathcal{O}_t$ is unaccessible to u at interaction t.*
- *Because of resource restrictions (e.g., time, energy), u can only choose to take a subset of the offered opportunities: $\mathcal{D}_t \subset \mathcal{O}_t$, $|\mathcal{D}_t| = l_t$.*
- *Each opportunity $i \in \mathcal{O}_t$ could potentially give u a revenue of r_{ui} if being taken.*

We assume an agent u is a rational decision maker; for each decision, u considers both revenue and opportunity cost, and decides which opportunity to take based on the potential profit of each opportunity in $\mathcal{O}$.

- Revenue: the revenue r_{ui} is the gross gain of u from taking the opportunity i; We consider a latent factor model in this paper $r_{ui} \sim \theta_u^\top \theta_i$.
- Opportunity Cost: the opportunity cost c_{ui} is the potential loss of u from taking an opportunity i excluding her to take other opportunities. c_{ui} equals to the revenue of the second best alternative: $c_{ui} = \max\{r_{ui'} : i' \in \mathcal{O} \setminus i\}$
- Profit: the profit $\pi_{ui} = r_{ui} - c_{ui}$ is the net gain of an decision.

PROPOSITION: *A rational decision is a decision maximizing the profit: i =* $\arg\max_{i \in \mathcal{O}} \pi_{ui}$.

This proposition provide directly justification to our proposed framework: the probability of observing an decision $i \in \mathcal{D}_t$ depends on the profit $\pi_{ui} = \theta_u^\top \theta_i - \max_{i' \in \mathcal{O} \setminus \mathcal{D}} \theta_u^\top \theta_{i'}$, rather than the revenue $r_{ui} = \theta_u^\top \theta_i$[2]; the more the profit π_{ui} is, the more likely u will take i; if the profits of all the opportunities are marginal, u will not invest her resource (e.g., time, energy) and none of the opportunities will be taken.

6 Experiments

This section presents preliminary experiments. We test the proposed two CCF models in comparison with CF models on a data set crawled from Yahoo! pulse social network. The data consists of 386 Yahoo! approved applications, 124,792 users, and 2,932,553 interactions indicating which user installs which application. Because the contextual information (the offer set $\mathcal{O}_t$ for each interaction t) is missing, we manually create a fixed-size pseudo-offer set for each interaction. Specifically, for every positive observation, e.g. $y_{ui} = 1$, we randomly sample a handy set of missing (unobserved) entries $\{y_{ui'}\}_{i'=1:m}$, and treat such items $\{i'\}$ as non-choices (e.g. $y_{ui'} = -1$,). In our experiments, we choose $m = 4$ pseudo non-choices; in other words, we assume the offer size l_t =5.

For comparison, we test the two CCF models (referred to as $CCF.Softmax$ and $CCF.Hinge$) against the two standard CF models (referred to as $CF.L2$ and $CF.Logistic$) discussed in Section 3.

For recommendation, the latent factor recommender models lead to a ranking of items according to the multiplicative score function. Hence it is natural to use ranking metrics to assess performance. We use the nDCG or *normalized Discounted Cumulative Gain* score, which gives larger credit to top-ranked entities and is widely used in IR community.

For our evaluation, we use a cross-validation setting where we randomly partition the data into two equally sized pieces and use one for training and the other for testing. The nDCG scores are computed on testing data only, and they are averaged over five random repeats.

[2]Both of the regression-based approaches are derived from revenue-maximizing decisions.

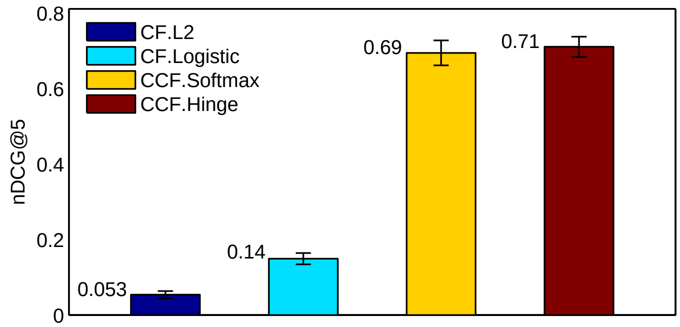

Figure 1: nDCG comparison of CF and CCF models on Y! Pulse data.

The results are reported in Figure 1. The comparison is striking. Among the four models, CF with MSE loss performs the worst, the mean nDCG@5 score is only 0.053. CF with logistic loss is a little better, the average nDCG@5 is 0.14. In contrast, the two CCF models boost the nDCG dramatically. The softmax model achieves ndCG@5 as high as 0.69, while the hinge model is even better, the nDCG@5 score is as high as 0.71, which is a 400% improvement compared to logistic regression based CF.

Our experiments are limited due to lack of real contextual data (i.e. the offer set $\mathcal{O}_t$ for each interaction t). Because all the published recommendation data sets we are aware of do not contain such contextual information, our comparison is based on pseudo contextual data. We believe that real contextual information contains more useful information for modeling and predicting user behavior. As such, we expect similar or even better results on real recommendation data where real contextual information is available. We call for industry attentions to test our CCF model on their recommender system.

7 The stochastic give-and-take (SGAT) process for joint CCF and exploration strategy learning

In this section, we present a Bayesian generative models for the opportunity give-and-take process, which employs the exploit-explore scheme for opportunity generation and the proposed CCF framework for decision making. This generative model, called *stochastic give-and-take (SGAT)*, could be used for joint learning of latent factor models as well as exploration strategies. We call for attentions from recommendation industry to test this model on real recommender systems.

SGAT: Given user space $\mathcal{U}$ and opportunity space $\mathcal{I}$, a generative OGAT process generates an interaction t as follows:

User sampling: u_t ~Multinomial(β)
Offer sampling: the system S employs an explore-exploit scheme to generate the offer set $\mathcal{O}_t$, for example:

- sample offer size: l_t ~Poison(ϵ)
- for $i = 1$ to l_t
    · sample $i \sim p(i|\theta_u^\top \theta_i, \gamma)$

Decision sampling: user u_t decides $\mathcal{D}_t$ according to her resource restriction and the potential profit of each offer, e.g.:

- sample decision size: m_t ~Poison(ε)
- for $i = 1$ to m_t
  · sample $p(i|u_t, \mathcal{O}_t) = \frac{\theta_u^\top \theta_i}{\sum_{i' \in \mathcal{O}_t} \theta_u^\top \theta_{i'}}$, $i \in \mathcal{O}_t$.

where $p(i|r_i, \gamma)$ is an explore-exploit strategy with utility (payoff) r_i and exploration probability γ[3]; β controls the frequency of user participation (e.g., some users visit the recommender system more frequently), ε captures the bias of user decision (e.g., some users more frequently take at least one of the recommender's suggestions). In practice, the offer size l might be a fixed number (e.g., in news recommendation, l is usually set to 5).

This generative model provide a single unified model for learning both the exploration strategy and the latent factor models simultaneously from the interaction data.

8 Conclusion

We have developed a theory of *local optimality for user choices* in recommendation system. We presented the framework of *collaborative competitive filtering* for recommender learning, which leverages latent factor models of CF to encode collaborations among items and users, and in the meanwhile, it also encodes the competition among the items being recommended to the users. CCF takes into account both the revenue and the opportunity cost of each user decision, and by optimizing the profit loss function, CCF encourages that each decision the user makes be locally the best among the opportunities being offered to her. We established two models in this spirit. Preliminary experiments indicate that CCF diamagnetically improves recommendation performance compared with standard CF approaches. We also presented a Bayesian model for joint learning of exploit-explore strategy and CCF factorization model. We call for attentions from recommendation industry to test our model on their recommender system.

References

Agarwal, Deepak and Chen, Bee-Chung. Regression-based latent factor models. In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 19–28, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-495-9. doi: http://doi.acm.org/10.1145/1557019.1557029.

Auer, Peter. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3:397–422, 2003. ISSN 1532-4435.

Brynjolfsson, Erik, Hu, Yu J., and Smith, Michael D. Consumer surplus in the digital economy: Estimating the value of increased product variety at online booksellers. *MANAGEMENT SCIENCE*, 49(11):1580–1596, November 2003. doi: 10.1287/mnsc.49.11.1580.20580. URL `http://dx.doi.org/10.1287/mnsc.49.11.1580.20580`.

[3]The simplest exploration strategy is to use the γ-greedy algorithm. Also because $r_i = \theta_u^\top \theta_i$ is linear w.r.t θ_i, more advanced bandit algorithm, such as the recently proposed LinRel(Auer, 2003) and LinUCB (Li et al., 2010) algorithms could be readily employed.

Brynjolfsson, Erik, Hu, Yu J., and Simester, Duncan. Goodbye pareto principle, hello long tail: the effect of search costs on the concentration of product sales. *MIT Center for Digital Business Working Paper*, December 2007. URL `http://papers.ssrn.com/sol3/papers.cfm?abstract_id=953587`.

Chen, Ye, Pavlov, Dmitry, and Canny, John F. Large-scale behavioral targeting. In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 209–218. ACM, 2009. ISBN 978-1-60558-495-9. URL `http://dblp.uni-trier.de/db/conf/kdd/kdd2009.html#ChenPC09`.

Herbrich, Ralf, Graepel, Thore, and Obermayer, Klaus. Support vector learning for ordinal regression. In *In International Conference on Artificial Neural Networks*, pp. 97–102, 1999.

Koren, Yehuda. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *KDD '08: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 426–434, 2008.

Koren, Yehuda, Bell, Robert, and Volinsky, Chris. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009. ISSN 0018-9162. doi: 10.1109/MC.2009.263.

Li, Lihong, Chu, Wei, Langford, John, and Schapire, Robert E. A contextual-bandit approach to personalized news article recommendation. In *WWW '10: Proceedings of the 19th international conference on World wide web*, pp. 661–670, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-799-8. doi: http://doi.acm.org/10.1145/1772690.1772758.

McLaughlin, Matthew R. and Herlocker, Jonathan L. A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 329–336, New York, NY, USA, 2004. ACM. ISBN 1-58113-881-4. doi: http://doi.acm.org/10.1145/1008992.1009050.

Miller, Kurt, Griffiths, Thomas, and Jordan, Michael. Nonparametric latent feature models for link prediction. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C. K. I., and Culotta, A. (eds.), *NIPS '09: Advances in Neural Information Processing Systems 22*, pp. 1276–1284, 2009.

Salakhutdinov, Ruslan and Mnih, Andriy. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *ICML '08: Proceedings of the 25th international conference on Machine learning*, pp. 880–887, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4. doi: http://doi.acm.org/10.1145/1390156.1390267.

Sarwar, Badrul, Karypis, George, Konstan, Joseph, and Reidl, John. Item-based collaborative filtering recommendation algorithms. In *WWW '01: Proceedings of the 10th international conference on World Wide Web*, pp. 285–295, New York, NY, USA, 2001. ACM. ISBN 1-58113-348-0. doi: http://doi.acm.org/10.1145/371920.372071.