An Incentive-Based Architecture for Social Recommendations

Rajat Bhattacharjee*
Google Inc.
rajatb@google.com

Ashish Goel†
Stanford University
ashishg@stanford.edu

Konstantinos Kollias‡
Stanford University
kkollias@stanford.edu

ABSTRACT

We present an incentive-based architecture for providing recommendations in a social network. We maintain a distinct reputation system for each individual and we rely on users to identify appropriate correlations and rate the items using a system-provided recommendation language. The key idea is to design an incentive structure and a ranking system such that any inaccuracy in the recommendations implies the existence of a profitable arbitrage opportunity, hence making the system resistant to malicious spam and presentation bias. We also show that, under mild assumptions, our architecture provides users with incentive to minimize the Kullback-Leibler divergence between the ratings and the actual item qualities, quickly driving the system to an equilibrium state with accurate recommendations.

Categories and Subject Descriptors

H.4.m [**Information Systems Applications**]: Miscellaneous

General Terms

Economics, Theory

Keywords

Incentives, Information aggregation, Recommender systems

---

*Research conducted while at Stanford University. Research supported by NSF ITR grant 0428868 and NSF award 0339262.

†Department of Management Science and Engineering and (by courtesy) Computer Science, Stanford University. Research supported by NSF ITR grant 0428868 and gifts from Google, Microsoft, and Cisco.

‡Department of Management Science and Engineering, Stanford University. Research supported by an A. G. Leventis Foundation Scholarship and the Stanford-KAUST alliance for excellence in academics.

RecSys'09, October 23–25, 2009, New York, New York, USA.

1. INTRODUCTION

Recommender systems [1] are an important Internet monetization tool. They help in monetizing the heavy tail and play an important role in the success of Internet-based businesses. The primary task of a recommender system is to suggest items of interest to its users. To this end, any correlation information regarding the similarities among various products and among the interests of individual users may prove very useful. Social networking services, whose success provides a fertile field for web-based commercial activity, offer a rich collection of such information. Coupling these tools and exploiting the information extracted from social networks, facilitates the generation of high-quality personalized recommendations.

The approach of identifying similarities between users in the domain of social recommender systems has been applied in the form of collaborative filtering techniques. However, these systems make no guarantees about the quality of the recommendations. It has been experimentally observed [7] that in systems where individual decisions are influenced by the decisions of others, quality alone doesn't determine the success of an item. We refer to this phenomenon as presentation bias. Spam is another deterrent in the effective functioning of these systems [4, 6]. Since the owner of an item has much to gain from its success, there is an incentive for agents to game the system.

Another approach, outlined by Bhattacharjee and Goel [2, 3], is to use incentive-based mechanisms for making ranking systems robust to presentation bias and spam. The work there assumed a simple setting of homogeneous population, and had no provisions for the mechanisms to work in the framework of personalized social recommendations. In this paper, we consider recommendations in this more complex landscape. We also show that under mild assumptions, the architecture we present provides users with incentive to minimize the Kullback-Leibler divergence between the ratings and the item qualities, leading to a fast convergence to an equilibrium state with high-quality recommendations.

2. THE INSPECT-UTILIZE MODEL

We break down a typical interaction of a user with her personalized reputation system into three steps. (1) In the first step, the system provides the user with a ranked list of recommended items (e.g. books). (2) Next, the user chooses to inspect the top j items. (3) Finally, among these items, the user utilizes a subset S (in the book example, purchases some of the books) and we say that a utility generation event has occurred for the items in S. We now present a model which captures this interaction. The set $\mathcal{E} = \{1, 2, \ldots, n\}$ models the n items and the set $\mathcal{U} = \{1, 2, \ldots, m\}$ models the m users of the system. The users interact with their individual reputation systems and may also update the ratings of the items. The model specifics follow.

1. **Quality.** We define as $q_{e,u}$, the probability that user u utilizes item e, conditioned on the fact that u has inspected e. We will refer to this probability as the quality of e with respect to user u. The task of the system is to give, for each user, an ordering of the items according to their quality with respect to the user. The actual value of $q_{e,u}$ is *unknown*.

2. **Slots.** The k slots, $1, 2, \ldots, k$ are the placeholders for the recommendations that are presented to each user. We assume that the probability with which user u inspects slot i is known and denote it by $p_{i,u}$. Obviously, it is the case that $p_{1,u} \geq p_{2,u} \geq \ldots \geq p_{k,u}$.

3. **Utility.** If user u utilizes an item e, we say a utility generation event has occurred. For simplicity, we will assume that all utility generation events result in the same revenue, R. This implies no loss of generality, since different revenue equivalents can be easily folded into the quality values.

3. OUR ARCHITECTURE

We now describe our feedback and incentive-based architecture which is designed in a way that users benefit from correcting the rankings.

1. **Social graph.** The users are organized in a social graph G with one node for very $u \in \mathcal{U}$ and edges between related individuals.

2. **Feedback scores.** Each of the m personalized reputation systems maintains n feedback scores, one for each item. For every user $u \in \mathcal{U}$ and for every item $e \in \mathcal{E}$, $\tau_{e,u}$ is the feedback score of e in the personalized reputation system of user u, which is not allowed to drop below 1.

3. **Tokens.** In our model, a rater (user) is able to alter the scores $\tau_{e,u}$ by placing tokens. A token T_i is a tuple $\{u, e, r\}$, where u is the rater who places the token, e the item on which the token is placed, and r a *token vector*. The token vector r has m elements and each element r_u is the increase/decrease of $\tau_{e,u}$. Restrictions on the possible token vectors used are imposed by the recommendation language (details below). The order of arrival of tokens is given by subscript i. A constraint is that at any given time the sum of the contributions made by a user is bounded by γ which is a system parameter. This means that, after placing her first few tokens, a user has to make negative contributions to some scores, in order to gain the right of making future positive contributions.

4. **Recommendation language.** For any $u \in \mathcal{U}$, the system defines a set, L_u, of allowed *recommendation vectors*. These m sets constitute the recommendation language. A vector $r = (r_1, r_2, \ldots, r_m) \in L_u$, must be a non-negative vector whose elements have unit sum. The token vectors used by u are scaled recommendation vectors. This means u can use vector αr, where α is a positive or negative real number, as a token vector if and only if $r \in L_u$. For the rest of the paper, with a slight abuse of language and notation, we allow the token vectors of user u to be linear combinations of the recommendation vectors in L_u.

5. **Revenue distribution.** The fraction of the revenue to be distributed as incentive among the users is determined by parameter $\beta \leq 1$. The parameter $s > 1$ controls the relative importance of tokens placed earlier on an item (to incentivize discovery of new items and deal with presentation bias). Suppose a utility generation event occurs for an item e by user u, and results in R amount of revenue being generated for the system. Let $\mathcal{T}$ be the set of all the tokens in the system. For a given token $T_i = \{u', e, r\} \in \mathcal{T}$, we define $w(T_i)$ as the weight by which $\tau_{e,u}$ was changed due to T_i. If T_i was placed on item e, then $w(T_i) = r_u$, otherwise $w(T_i) = 0$. Further, we define, $W(T_i) = 1 + \sum_{T_j \in \mathcal{T}: j \leq i} w(T_j)$, as the value of $\tau_{e,u}$ after T_i was placed. The revenue share of the rater who placed token T_i (user u') is given by
$$\beta R(s-1) \int_{W(T_i)-w(T_i)}^{W(T_i)} \frac{dx}{x^s}.$$
This definition ensures that the total shared revenue is bounded by βR and that the share of a rater does not depend on future tokens. For each u, we maintain an account acc_u. Depending on the sign of the revenue share, the amount is added or subtracted from acc_u. The situation arising from bankruptcy of a user has been discussed in [3].

4. MAIN RESULTS

In this section, we present our main results. We define as *visibility* of item e for user u, the probability with which e is inspected by u, after being ranked in her recommendations and denote it $\eta_{e,u}$. The expected rate at which revenue is generated for an item e, by user u, is given by $\eta_{e,u} q_{e,u}$. Let,
$$f(e, u) = q_{e,u} \frac{\eta_{e,u}}{\tau_{e,u}^s}.$$
Note that f is proportional to the profit (or loss) that the rater can expect to make by making an instantaneous positive (or negative) contribution to the feedback score of item e, in the reputation system of user u (since it is proportional to the derivative of the revenue sharing expression from Section 3). Our structure uses the ranking algorithm described by Bhattacharjee and Goel in [3], which has the following properties.

1. Higher feedback score implies higher visibility, that is, $\tau_{e,u} \geq \tau_{e',u'} \Rightarrow \eta_{e,u} \geq \eta_{e',u'}$.

2. If $\tau_{e,u} \geq \tau_{e',u'}$, then $\frac{\eta_{e,u}}{\tau_{e,u}^s} \leq \frac{\eta_{e',u'}}{\tau_{e',u'}^s}$, where $s > 1$ is the same as earlier.

3. Under mild assumptions (see [3]), the ranking algorithm ensures $\eta_{e,u} = \lambda \tau_{e,u}$, for all $e \in \mathcal{E}$ and $u \in \mathcal{U}$, where λ is a constant. Since $s > 1$, this subsumes the last property (when the assumptions hold). Briefly and informally the assumption states that the vector

of normalized ratings is majorized [5] by the vector of normalized slot inspection probabilities.

Now, using f, we will explain the notion of a *profitable arbitrage opportunity*.

Definition 1. *For any user, the act of updating the scores of items $e_1, e_2, \ldots, e_j$, using the token vectors $r^1, r^2, \ldots, r^j$ (remember that each token vector is a linear combination of allowed recommendation vectors and can have both positive and negative entries), presents a profitable arbitrage opportunity when,*

$$\sum_{i=1}^{j} \sum_{u \in \mathcal{U}} r_u^i f(e_i, u) > 0 \text{ and } \sum_{i=1}^{j} \sum_{u \in \mathcal{U}} r_u^i = 0.$$

The inequality means that the user has instantaneous profit when making those recommendations and the equality guarantees that the recommendation are compliant with the γ bound on the sum of the user's contributions (equal negative and positive contributions).

We now define the notion of an inverted ranking in the recommendations.

Definition 2. *We say that the pair (e, e') is a case of an* inverted ranking *in the recommendations for some user u, if $q_{e,u} < q_{e',u}$ and $\eta_{e,u} \geq \eta_{e',u}$.*

At this point, we will define the notion of a *complete recommendation language.*

Definition 3. *Let 1_u be the vector of size m with the u-th position equal to 1 and all other positions equal to 0. We will say that a recommendation language is complete when it has the following properties.*

1. *For every $u \in \mathcal{U}$, there exists a linear combination of all allowed recommendation vectors, which equals 1_u. That is, for every $u = 1, 2, \ldots, m$, there exist real numbers $\alpha^u_{u',r}$ such that, $\sum_{u' \in \mathcal{U}} \sum_{r \in L_{u'}} \alpha^u_{u',r} r = 1_u$. This property ensures that there is a way for the users to update the score of some item e just in the reputation system of user u.*

2. *For every user u, $I = (\frac{1}{m}, \frac{1}{m}, \ldots, \frac{1}{m}) \in L_u$. That is, we expect that the recommendation language allows any user to place a recommendation for every user in the system.*

We now give the following theorem, which relates any inaccuracy in the recommendations with a profitable opportunity for the users.

Theorem 1. *Assuming that the recommendation language is complete, the existence of an inverted ranking pair (e, e') in the recommendations for some user u, implies the existence of a profitable arbitrage opportunity for some user.*

Proof. Since (e, e') is an inverted ranking pair in the recommendations for u, we have $q_{e,u} < q_{e',u}$ and $\eta_{e,u} \geq \eta_{e',u}$. Combining this with the properties of the ranking algorithm, mentioned earlier in this section, we get $f(e, u) < f(e', u)$. This means the function f is not constant for all pairs (e, u). Let $(e_h, u_h) = \arg\max_{e,u} f(e, u)$ and $(e_l, u_l) = \arg\min_{e,u} f(e, u)$. We also know that the recommendation language is complete, so we will write $\alpha_{u,r}, \beta_{u,r}$ for the multipliers that give,

$$\sum_{u} \sum_{r \in L_u} \alpha_{u,r} r = 1_{u_h} \text{ and } \sum_{u} \sum_{r \in L_u} \beta_{u,r} r = 1_{u_l}. \quad (1)$$

We now examine the vector $\sum_u \sum_{r \in L_u} \alpha_{u,r} r$. Since it is equal to 1_{u_h}, it follows that the sum of its elements is equal to 1, hence we get

$$\sum_{u'} \sum_{u} \sum_{r \in L_u} \alpha_{u,r} r_{u'} = 1 \Rightarrow \sum_{u} \sum_{r \in L_u} \alpha_{u,r} \left(\sum_{u'} r_{u'} \right) = 1.$$

From the definition of an allowed recommendation vector, we know that the elements of every r have unit sum. Which gives us, $\sum_u \sum_{r \in L_u} \alpha_{u,r} = 1$. Using the same argument we get the same result for the multipliers The set $\mathcal{U} = \{1, 2, \ldots, m\}$ models the m users of the system, who perform the role of inspecting the recommendations and, potentially, utilizing the items.$\beta_{u,r}$ as well. So,

$$\sum_{u} \sum_{r \in L_u} \alpha_{u,r} = \sum_{u} \sum_{r \in L_u} \beta_{u,r} = 1. \quad (2)$$

Now consider that some user u performs the following recommendations. Initially, u applies the token vector $\sum_{r \in L_u} \alpha_{u,r} r$ on item e_h (we will call this Recommendation 1) to get instantaneous profit,

$$\sum_{r \in L_u} \alpha_{u,r} \sum_{u'} r_{u'} f(e_h, u'). \quad (3)$$

Next, u applies the token vector $-\sum_{r \in L_u} \beta_{u,r} r$ on item e_l (Recommendation 2) to get instantaneous profit,

$$-\sum_{r \in L_u} \beta_{u,r} \sum_{u'} r_{u'} f(e_l, u'). \quad (4)$$

Finally, u applies the token vector $\left[\sum_{r \in L_u} (\beta_{u,r} - \alpha_{u,r})\right] I$ on some arbitrary item e' (Recommendation 3) to get instantaneous profit,

$$\left[\sum_{r \in L_u} (\beta_{u,r} - \alpha_{u,r}) \right] \sum_{u'} \frac{1}{m} f(e', u'). \quad (5)$$

Observe that the sum of the contributions made by u is exactly 0, hence, there is no danger in violating the γ limit on the sum of the contributions. We claim that there is some user u, such that the sum of her instantaneous profits for placing recommendations 1, 2, and 3 is positive and, thus, there exists a profitable arbitrage opportunity for u. We will prove this by showing that if we take the sum of the instantaneous profits (3), (4), (5), and sum it over all u, we get a positive number. Starting with summing (3) for all u and combining with (1), we get that the sum of the instantaneous profits Recommendation 1 is,

$$\sum_{u} \sum_{r \in L_u} \alpha_{u,r} \sum_{u'} r_{u'} f(e_h, u') = f(e_h, u_h). \quad (6)$$

Similarly, for Recommendation 2, from (1) and (4) we get,

$$-\sum_{u} \sum_{r \in L_u} \beta_{u,r} \sum_{u'} r_{u'} f(e_l, u') = -f(e_l, u_l). \quad (7)$$

Finally, for Recommendation 3, summing (5) for all u and combining with (2), we get,

$$\sum_u \left[\sum_{r \in L_u} (\beta_{u,r} - \alpha_{u,r}) \right] \sum_{u'} \frac{1}{m} f(e', u') = 0. \qquad (8)$$

Now, summing expressions (6), (7), and (8), we get that the sum of the instantaneous profits that each user would have by placing recommendations 1, 2, and 3, unilaterally, is $f(e_h, u_h) - f(e_l, u_l) > 0$. □

We now prove a conditional theorem which implies that at equilibrium the feedback scores are correlated with the quality scores. The precondition refers to the mild assumptions given in the description of the algorithm in [3] and mentioned in the beggining of this section.

THEOREM 2. *If there exists a ranking algorithm which ensures visibility $\eta_{e,u} = \lambda \tau_{e.u}$, where λ is a constant for all (e, u), then there exists an arbitrage opportunity unless the ratio $\frac{q_{e,u}}{\tau_{e,u}^{s-1}}$ is constant for all (e, u).*

PROOF. Suppose $\frac{q_{e,u}}{\tau_{e,u}^{s-1}}$ is not constant for all (e, u). Then, there exist two pairs (e^*, u^*), (e', c') such that $f(e^*, u^*) < f(e', u')$. Following arguments similar to the ones used in the last proof, we can prove that there exists an arbitrage opportunity. □

We now focus on the quadratic incentive scheme, that is, the case $s = 2$. Consider the following potential function,

$$\mathcal{P}(q, \tau) = -\sum_{e,u} q_{e,u} \log \tau_{e,u}.$$

Notice that if we interpret q and τ as distributions, then the above potential function is equivalent to the Kullback-Leibler divergence (remember that q is fixed), which is,

$$D_{KL}(q||\tau) = \sum_{e,u} q_{e,u} \log \frac{q_{e,u}}{\tau_{e,u}}.$$

In the light of this observation, the next theorem says that it is most profitable for users to leave feedback which provides the most additional information relative to the current state of the system, leading to an equilibrium where all items are correctly rated for all users.

THEOREM 3. *If there exists a ranking algorithm which ensures visibility $\eta_{e,u} = \lambda \tau_{e,u}$, where λ is a constant for all (e, u), then,*

1. *The potential function $\mathcal{P}$ is minimized when the feedback scores are directly proportional to the quality scores, that is, there exists a constant α for all (e, u) such that $\tau_{e,u} = \alpha q_{e,u}$.*
2. *At any given time, the most profitable arbitrage opportunity is presented by the recommendation which minimizes the potential function $\mathcal{P}$ the most (given the restrictions placed by the recommendation language).*

PROOF. The proof presented here follows the proof of the famous Gibbs' inequality in information theory. However, for the sake of completeness, we give the details here.

Let $Q = \sum_{e,u} q_{e,u}$ and $T = \sum_{e,u} \tau_{e,u}$ (both Q and T are constant). We modify $\mathcal{P}$ to get,

$$\mathcal{P}^*(q, \tau) = \sum_{e,u} \frac{q_{e,u}}{Q} \log \frac{\tau_{e,u} Q}{q_{e,u} T}.$$

Since $q_{e,u}$'s are fixed, $\mathcal{P}$ is minimized where $\mathcal{P}^*$ is maximized, and vice versa. Note that $\log x \leq x - 1$, with equality if and only if $x = 1$. We get,

$$\mathcal{P}^*(q, \tau) \leq \sum_{e,u} \frac{\tau_{e,u}}{T} - \sum_{e,u} \frac{q_{e,u}}{Q} = 0.$$

Hence, the potential function $\mathcal{P}$ is uniquely minimized at $\tau_{e,u} = q_{e,u} \frac{T}{Q}$. The second part of the theorem follows from the observation that the partial derivative of $\mathcal{P}(q, \tau)$ with respect to $\tau_{e,u}$ is $f(e, u)$. □

5. DISCUSSION

At this point, we conclude the paper with a discussion of how our scheme might be applied in practical systems. Consider a Netflix-like recommender system and an underlying social structure. The set of items is the set of movies and the social graph is given by the social structure. A recommendation can be made to all users who like sci-fi movies, or to all those belonging to a specific age group. Also, some user, who might know a user u (a friend) well, can specifically recommend a movie to u. All those recommendations can be made possible by including the appropriate vectors in the recommendation language. Every user interacts with a designed interface, which presents options compatible with human intuition and then translates the user's selections into recommendation vectors. The recommendation language can also be used to capture correlations between the interests of individual users, given from collaborative filtering techniques. The quality of a movie with respect to a user is the probability that the user rents a movie conditional on the fact that she inspects the recommendation.

6. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. 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):734–749, 2005.

[2] R. Bhattacharjee and A. Goel. Incentive based ranking mechanisms. In *NetEcon '06: Proccedings of the first workshop on the Economics of networked systems*, pp. 62–68, New York, NY, USA, 2006. ACM.

[3] R. Bhattacharjee and A. Goel. Algorithms and incentives for robust ranking. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 425–433, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[4] S. K. Lam and J. Riedl. Shilling recommender systems for fun and profit. In *WWW*, pp. 393–402, 2004.

[5] A. W. Marshall and I. Olkin. *Inequalities: Theory of Majorization and Its Applications*. Academic Press, New York, 1979.

[6] M. O'Mahony, N. Hurley, N. Kushmerick, and G. Silvestre. Collaborative recommendation: A robustness analysis. *ACM Trans. Interet Technol.*, 4(4):344–377, 2004.

[7] M. Salganik, P. Dodds, and D. Watts. Experimental Study of Inequality and Unpredictability in an Artificial Cultural Market. *Science*, 311(5762):854–856, 2006.