Market-Based Recommendations: Design, Simulation and Evaluation *

Yan Zheng Wei, Luc Moreau and Nicholas R. Jennings
Department of Electronics and Computer Science
University of Southampton
Southampton SO17 1BJ, UK
{ yzw01r, L.Moreau, nrj }@ecs.soton.ac.uk

ABSTRACT

This paper reports on the design, implementation, and evaluation of a market-based recommender system that suggests relevant documents to users. The key feature of the system is the use of market mechanisms to shortlist recommendations in decreasing order of user perceived quality. Essentially, the marketplace gives recommending agents the incentive to adjust their bids to different levels according to their belief about the corresponding user perceived quality. In order to test the efficiency of our marketplace design, this paper reports on our simulation results for different types of users with different information needs. In this context, we demonstrate that the bids from recommendations with different user perceived quality levels converge at different price levels and that the bidding agents can relate their bids to their internal belief about the quality of their recommendations.

Keywords

Recommender System, Auctions, Market-Based Approach

1. INTRODUCTION

This paper reports on the design, implementation and evaluation of a recommender system that seeks to manage the problem of information overload. Generally speaking, such systems help make choices among recommendations from all kinds of sources without having sufficient personal experience of all these alternatives [10]. In a typical system, recommendations are provided as inputs and the system then aggregates and directs them to appropriate recipients. Thus a recommender system's main value lies in information aggregation and its ability to match the recommenders with those seeking recommendations.

Recommender systems have been applied to many application domains and many different techniques have been used to make the recommendations. For example, some are based on the correlation between the item contents (such as term frequency inverse document frequency [11] and weighting [4]), while others are based on the correlation between users' interests (such as votes [5] and trails [3]). However, there is no universally best method for all users in all situations and we believe this situation is likely to continue as ever more methods are developed. Moreover, the ranking of relevance produced by the different methods can vary dramatically from one another. Given this situation, we believe the best way forward in this area is to allow the multiple recommendation methods to co-exist and to provide an overarching system that coordinates their outputs such that only the best recommendations (from whatever source or method) are presented to the user. To this end, in [9] we developed a system that integrates multiple recommendation methods to help the users with the problem of "where to go next?" when they browse the Web (see Figure 1). This system used a *market-based approach* to achieve such coordination. This is because the problem of selecting appropriate recommendations to place in the limited sidebar space can be viewed as one of scarce resource allocation and markets are an effective solution for this class of problems [2]. This market structure was refined in [14] to one that theoretically is Pareto-optimal, social welfare maximizing, stable and fair to all the recommending agents. Given this analysis, this paper reports on the actual implementation of the marketplace and its evaluation to determine whether the theoretical properties hold in practice. This, in turn, is the key to whether the market can actually be used as an overarching coordination framework for our practical software development endeavor (which is the next stage of our work). In more detail, we develop three kinds of agents (reward agents, recommending agents and user agents) that operate in the marketplace and a number of simulations were performed to assess the system's operational characteristics.

The remainder of this paper is structured in the following manner. Section 2 further develops the market mechanism in [14] and details the behaviour of the three kinds of agents. Section 3 details and then empirically evaluates the performance of the marketplace. Section 4 outlines related work in terms of reducing information overload and market-based systems. Finally, section 5 concludes and points to future work.

2. DESIGNING THE SYSTEM

Our marketplace operates according to the following metaphor. A *user agent* acting on behalf of the user is selling sidebar space where recommendations may be displayed (see left hand side of Figure 1). The number of such slots is fixed and limited. Information providers (the component *recommending agents*) want to get their recommendations advertised in the user's browser and so compete in the marketplace to maximize their individual gain by purchasing this advertising space when they have what they believe are

*This research is funded in part by QinetiQ and the EPSRC Magnitude project (reference GR/N35816).

Figure 1: Browser with Recommendations

good recommendations. Their bids indicate how much they are willing to pay for such slots. The recommender system acts as the auctioneer and selects the most valuable items (highest bids) which it then displays as its recommendations (those agents that provided these *shortlisted* items are then charged according to their bids). The user then chooses some of these recommendations (or not) according to their interests. The agents that provided the user-selected recommendations receive some reward (since these recommendations are deemed useful) from the *reward agent*, while those not chosen receive no reward. The agents with recommendations rewarded might increase their revenue and those agents that make poor recommendations make losses (since they have to pay to advertise their recommendations). Thus, over the longer-term, the agents that make good recommendations become richer and so are able to get their recommendations advertised more frequently than those whose recommendations are infrequently chosen by the user.

In [14] we introduced the valuation of a recommendation from two perspectives. First, from the viewpoint of the user, how well a recommendation satisfies the user is termed the *user perceived quality*. Second, from the viewpoint of a recommending agent with a specific recommendation method, the relevance score it computes for a particular recommendation is termed its *internal quality*. The role of the marketplace is then to try and connect these two quality values by imposing a reward regime that incentivises the recommending agents to bid in a manner that establishes an appropriate correlation between these values and their bid price. To evaluate our mechanism we will use the following evaluation metrics:

Price Convergence: This happens if the price with respect to a user perceived quality level converges after a number of rounds of auctions. This is important from the viewpoint of the bidding agents since it enables them to learn to bid rationally at a certain level to maximize their revenue. Without convergence, an agent will never know how much to bid with respect to a given recommendation and therefore the marketplace behaviour will be chaotic.

Efficient Shortlists: The marketplace should be capable of shortlisting the recommendations in decreasing order of the user perceived quality after a number of auction iterations. This is important from the point of view of the users since they want only a few of the best recommendations.

Clear Incentive: The market should incentivize the recommending agents to differentially bid for different internal quality levels. This is important because the agents need to relate their bids to the internal quality of the recommendations.

Stability: A marketplace is stable if it provides all agents with an incentive to behave in a particular way over time. The marketplace should be designed to be stable because if a self-interested agent is better off behaving in some other manner than desired, it will do so. Stability is important because without it the system behaviour is unpredictable.

Fairness: The market is fair if it gives all recommendations equal opportunity of being shortlisted (irrespective of the agent or method that generates them). This is important because we want the system to shortlist the best recommendations in an unbiased manner.

With these metrics in place, we first outline the auction protocol for our marketplace (see [14] for more details and a justification of our design choice) before detailing the recommending, the user and the reward agents (see Figure 2). The particular protocol we employ is a *generalized first-price sealed-bid auction*, in which all agents whose recommendations are shortlisted pay an amount equal to their valuation of the advertisement and those whose recommendations selected by the user are awarded. In more detail, the market operates in the following manner. Each time the user browses a new page the auction is activated. In each such activation, the auctioneer agent calls for a number of bids (one bid contains one recommendation and its corresponding price), say M ($M > 0$ and the value of M is fixed) equal to the number of recommendations it is seeking. There are S recommending agents in the system and each recommending agent submits at most M bids in each auction. After a fixed time, the auctioneer agent ranks all the bids it has received by their bidding price, and directs the M bids with the highest prices to the user's browser. Those agents with shortlisted bids pay according to how much they bid. Those agents whose recommendations are not shortlisted pay nothing. The user may then take up a number of these shortlisted recommendations in which case the agents that supplied them are rewarded.

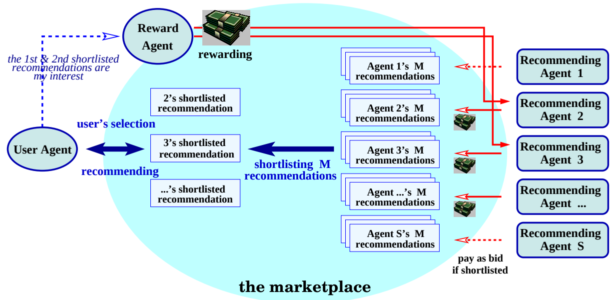

Figure 2: The Auction Protocol

2.1 Designing The Reward Agent

The reward agent determines the reward paid to the agents who make recommendations selected by the user. Given a fixed amount of total payoff to be distributed to N user selected recommendations, we have designed a Pareto efficient and social welfare maximizing reward mechanism [14]. The reward mechanism is as follows. All user-selected recommendations are ordered by the reward agent in decreasing rank of user perceived quality (such that $Q_1 > Q_2 > \cdots > Q_N$). The user-selected-recommendation with the h^{th} highest user perceived quality, Q_h ($h \in [1..N]$), is termed *the h^{th} rewarded recommendation*. To simplify the description of the reward mechanism, the following variables are used: let P_{M+1} be the highest not shortlisted bid; P_h be the bidding price of the h^{th} rewarded recommendation; and P^*_m ($m \in [1..M]$) be the historical average bidding price of the m^{th} shortlisted recommendation during the system's lifetime (note the bidding agents do not actually know this value). With these definitions in place, the reward to the h^{th} rewarded recommendation, R_h, is defined as:

$$R_h = \delta \cdot Q_h \cdot P_{M+1} - \ \alpha \cdot |P^*_h - P_h| \qquad (1)$$

where δ and α are two system coefficients ($\delta > 0$ and $\alpha > 1$) that are set according to the system's desired characteristics (see section 3).

2.2 Designing The Recommending Agents

In this subsection, we discuss how an agent generates a bid and how it relates the bidding price to its internal quality for a recommendation.

Each agent has a set of recommendations available to suggest. From the point view of a recommending agent, what it needs to do is to compute the relation between its local perception of relevance and the user's current context. Having done this, it can then bid an appropriate price to maximize its revenue. Given a specific recommending agent, the agent will relate its bidding price to its knowledge about the user perceived quality (reflected by the rewards it has received) with respect to different internal quality levels. We term this relationship between the bidding price and the internal quality an agent's *strategy profile*. This profile is on a per agent basis. It records an agent's bidding price for different internal quality levels and indicates how an agent should relate its bid to its internal quality.

In order to build up its strategy profile, an agent will divide its range of internal quality values into a number of equal segments. For each such segment, the agent will base its next bid (P^{next}) on its last bid price (P^{last}) and what happened to this bid:

1. **The Bid Was Not Shortlisted**
   The agent will increase the bid price to get them shortlisted:
   $P^{next} = Y \cdot P^{last} \quad (Y > 1)$.

2. **The Bid Was Shortlisted But Not Rewarded**
   The agent will decrease the bid price to lose less:
   $P^{next} = Z \cdot P^{last} \quad (0 < Z < 1)$.

3. **The Bid Was Rewarded**
   In [14] we proved that an agent can be aware of whether its bid price is close to or far from the market equilibrium price with respect to the user perceived quality of the segment. This can be achieved from the agent's knowledge about the *profit* (reward obtained minus bidding price that has been paid with respect to one recommendation) made from the segment in the last few rewarded rounds. The fact is that the profit increases (or decreases) when the bid price is close to (or far from) the equilibrium price. We also proved that only by minimizing the difference between its bids and the equilibrium price ($|P^*_h - P_h|$ see equation (1)) can an agent maximize its revenue. Therefore, an agent will keep increasing (decreasing) the next bid price until it makes less profit, whereupon it will decrease (increase) its price. To describe the behaviour with respect to the last bid rewarded, several variables are involved: *current profit* is the profit made from the current auction and *last profit* is that of the last auction; ΔP is the small amount of adjustment made to the last bid so as to move closer to the equilibrium price (each time ΔP changes its sign, its absolute value shrinks to chase the equilibrium price more accurately with respect to the user perceived quality of the segment); P_{lim} is the minimum absolute value of ΔP ($|\Delta P|$ cannot shrink indefinitely because otherwise it will reduce to 0 and would become useless). With these variables in place, the behaviour with respect to last bid rewarded is defined as below ($0 < X < 1$):

```
IF ( current profit > last profit )          //no profit loss
    P^next = P^last + ΔP
ELSE                                         //profit loss occurs
    ΔP = (−1) · ΔP                           //change sign of adjustment
    IF ( |ΔP| > P_lim )
        ΔP = X · ΔP                          //shrink adjustment
    P^next = P^last + ΔP
```

2.3 Designing The User Agent

To evaluate our marketplace we need to simulate the choices of a user in selecting recommendations. We do this by deploying a user model inside the user agent. Building on the user simulation of [1], we adopt the following models. When a user faces a set of shortlisted recommendations, the user will first visit some of them and will then have a valuation of each visited recommendation. Thus, a user assigns a number, Q_i ($i \in [1..M]$, $Q_i \in [0..100]$), to each visited recommendation according to his valuation of the recommendation. This number Q_i corresponds to the user perceived quality. Based on this value, a user's behaviour with respect to selecting shortlisted recommendations can be classified by:

Independent Selection: The selection of one recommendation is independent of the others. Once the user perceived quality of a recommendation is greater than or equal to a particular *acceptance threshold* (AT), the recommendation is accepted and rewarded. Those recommendations with user perceived quality less than AT will not be selected and therefore receive no reward.

Search-Till-Satisfied Behaviour: The selection of one recommendation is dependent on other recommendations that are ranked above it in the list. In this case, the user stops searching when a recommendation is encountered that has a user perceived quality greater than or equal to a particular *satisfaction threshold* (ST).

By means of an illustration, Table 1 is an example of a user's decision under the two different models. All recommendations with user perceived quality above the AT (60) are selected in case of independent selection. However, Q_7,

Shortlisted Recommendations	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	Q_8	Q_9	Q_{10}
User Perceived Quality	70	50	75	30	60	82	90	85	65	55
Independent Selection	1	0	1	0	1	1	1	1	1	0
Search-Till-Satisfied	1	0	1	0	1	1	0	0	0	0

Table 1: User's Decision of Different models
Both models have the same AT of 60. Search-till-satisfied model has an ST of 80. "1" means the recommendation is selected to be rewarded, while "0" means not selected.

Q_8 and Q_9 are not selected by the search-till-satisfied behaviour though their user perceived quality is above the AT. This is because Q_6 with a quality of 82 is found and since this is above the ST (80) the user stops searching.

3. SIMULATION AND EVALUATION

We have simulated the marketplace in order to study its properties and its suitability for producing good recommendations. To this end, this section presents the experimental settings of the simulation with respect to the reward agent, the recommending agents and the user agent. The system properties are also evaluated empirically.

3.1 Experimental Setting

The configuration of the reward agent, recommending agents and user agent are presented in this subsection. The number of bids called for (M) is not under the control of any of these three kinds of agents and we set it to the value of ten.

3.1.1 Configuring the Reward Agent

The reward agent controls two system variables: δ and α (see equation (1)). δ effects the volume of the credit paid to a user-selected-recommendation. The bigger δ is, the more the recommendation is paid. α effects the sensibility of the incentives the marketplace delivers to the recommending agents to let them be aware of the equilibrium (because the recommending agents need large alterations to chase the equilibrium price if α is big). In our experiment, we set $\delta = 1.5$ and $\alpha = 1.5$ based on our experience that these values enable the recommending agents to increase their revenue by making good recommendations over the long term and chase the equilibrium quickly.

3.1.2 Configuring the Recommending Agents

Our marketplace contains many different recommendation methods (represented as agents) that will each have their own valuation of the relevance of a particular recommendation to the user in a particular context. This means the same document may have varying internal quality scores with each different method. In our experiments, we simulate the internal quality of one method by a random variable that follows a normal distribution. The probability density function of the normal distribution is defined as [1]:

$$N(\mu, \sigma^2): \quad f(x) = \frac{1}{\sqrt{2\pi}\ \sigma}\ e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \qquad x \in [0, 1]$$

where μ and σ are the mean value and variance of the random samples. We assume that three recommendation methods are used in our system. Each recommendation method relates to one of the three distributions: $N(0.35, 0.1^2)$, $N(0.5, 0.1^2)$ and $N(0.65, 0.1^2)$ (see Figure 3). These distributions were chosen based on our previous studies [9] in this area. In this case, the mean of one distribution represents the average value of the internal quality of all samples generated by the corresponding method. The middle range between one unit of variance on both sides of the mean of one distribution contains the majority of the samples (about 68 percent of its total). We set the variance to a small value (0.1). This means the three different methods share only a few internal quality levels (this is important because otherwise it is hard to distinguish different methods).

[1]We fix the sample into the range [0, 1] (rather than ($-\infty$, $+\infty$)) because most internal quality results are a percentage number and are positive.

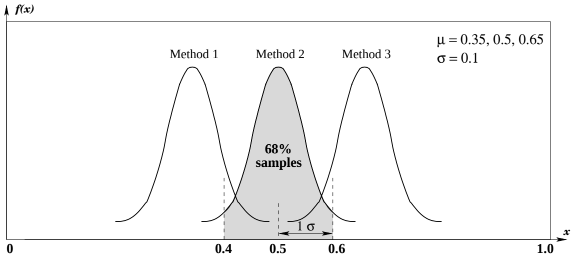

Figure 3: Different Recommendation Methods' Internal Quality Profiles

One of the key objectives of the recommending agents is to build up their strategy profiles so that they can relate their bidding price to their internal quality based on their knowledge about the reward (which reflects the user perceived quality of the recommendations). In order to learn such characteristics for all internal quality levels, each agent segments its strategy profile into 20 continuous segments. In each auction, a recommending agent needs to compute the internal quality of ten recommendations and make ten corresponding bids. In the early auction rounds, all agents' strategy profiles are empty. With an empty strategy profile, an agent will bid proportionally (because an agent can only expect a high internal quality recommendation to receive a high user perceived quality and more reward than a low internal quality recommendation) to the internal quality of ten recommendations based on an initial seeding price. We set different values (randomly generated from the range [128, 256]) for different recommending agents (because different agents valuate their recommendations differently with their empty strategy profiles). After each auction, all strategy profile segments that any of the ten bids belong to record and update information about: the last bid status (not shortlisted, shortlisted but not rewarded or rewarded), the last bid price, the last rewarded price, the last rewarded profit, ΔP, and the number of rewards. Based on such information about each segment, and using the bidding strategy presented in subsection 2.2, an agent can compute its bids in subsequent auctions if there are recommendations that belong to this segment. After a number of iterations, those segments that cover the majority of samples will have sufficient information to reach the equilibrium price and form a stable strategy profile.

3.1.3 Configuring the User Agent

When real users use a recommender system, they actually visit the shortlisted recommendations (or not) and therefore have a valuation about how much each recommendation satisfies them. Hence, when simulating the user we assume the user agent knows its valuation for each recommendation and correspondingly we assign the user perceived quality based on this valuation. From a recommending agent's viewpoint, the internal quality reflects the internal valuation of a recommendation. However, the relationship between the user perceived quality and the internal quality of any given recommendation is hard to determine. To this end, in simulating user perceived quality for a given recommendation we make two assumptions [2]:

Assumption 1: *With respect to any given recommendation method (represented as $N(\mu, \sigma^2)$), there exists a relationship* $\mathbf{f}$ *between the user perceived quality Q and the internal quality q of a recommendation.* More formally:

$$\forall N(\mu, \sigma^2),\ \exists \mathbf{f} \ \longrightarrow \ \textit{if}\ q \sim N(\mu, \sigma^2),\ \textit{then}\ Q = \mathbf{f}(q)$$

Assumption 2: *Based on assumption 1, given a recommendation method that follows a specific internal quality distribution $N(\mu, \sigma^2)$, a recommendation with higher internal quality q receives higher user perceived quality Q.* To simplify the problem, we further assume that Q has a linear relationship $\mathbf{f}$ with q with respect to a given $N(\mu, \sigma^2)$. More formally:

$$\forall N(\mu, \sigma^2),\ \exists \mathbf{f} \ \longrightarrow$$
$$\textit{if}\ q \sim N(\mu, \sigma^2),\ \textit{then}\ Q = \mathbf{f}(q) = k \cdot q + b \quad (k > 0)$$

where b is a coefficient that can be any value.

Based on these assumptions, when faced with recommendations provided by different methods (represented by different $N(\mu, \sigma^2)$), a user has a different value of k and b to assign to the value of Q for q generated from $N(\mu, \sigma^2)$. With respect to the three recommendation methods of Figure 3, what a user agent does is to normalize all the internal quality samples generated from the three distribution functions into one with a single distribution. To simplify this process, we define the normalization function as follows:

$$Q(q) = \begin{cases} 100 \cdot (q + 0.15) + \theta & \text{if } q \sim N(0.35, 0.1^2) \\ 100 \cdot q + \theta & \text{if } q \sim N(0.50, 0.1^2) \\ 100 \cdot (q - 0.15) + \theta & \text{if } q \sim N(0.65, 0.1^2) \end{cases} \quad (2)$$

where θ is the noise added to q and is, in fact, another random variable that follows a uniform distribution within the range of [-5, 5]. Thus, the value of Q is within the range of [0, 100] irrespectively of the noise.

3.2 Analysis

Having outlined the set up of the three kinds of agents specified in subsection 3.1, this section evaluates the system properties, through various simulations, with respect to the evaluation criteria specified in section 2.

Price Convergence. In [14] we proved that the marketplace can reach an equilibrium such that the shortlisted prices converge at different levels with respect to different user perceived quality levels. To evaluate this, we arranged 200 auctions with 10 shortlisted recommendations using the independent selection user model ($AT = 58$) and 9 recommending agents (every three share one method defined in Figure 3) to see if the marketplace does indeed have such a convergence property. From Figure 4, we can see that the shortlisted prices converge (for example, the 4^{th} and 10^{th} bid oscillate around 180 and 170 respectively, which indicate P_4^* and P_{10}^* respectively) after about 60 auctions. Because of the limited space, we do not show the case of the search-till-satisfied user model being used in a separate figure (the market converges slowly with $ST = 80$ and $AT = 58$ compared with the independent selection since fewer agents are rewarded in this case and they need more bids to chase the equilibrium price). From our simulations, we find that the shortlisted prices always converge after a number of iterations, but the speed of the convergence depends on the setting of the parameters α, AT, ST, Y, Z and ΔP (as introduced in subsection 3.1). Because of the limited space, we do not discuss all the effects of these variables in detail[3].

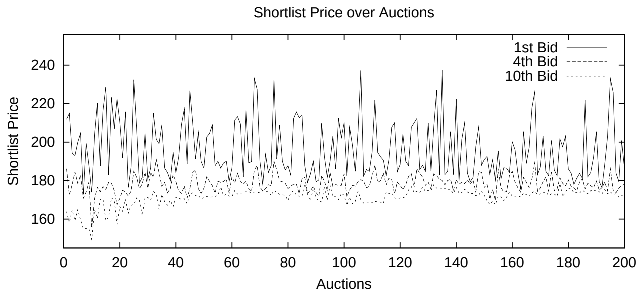

Figure 4: Convergence of Shortlisted Prices

Efficient Shortlists. The most important feature of our system is its capability to shortlist the best recommendations in decreasing order of user perceived quality when the market converges. To this end, Figure 5(a) shows the user perceived quality of the shortlisted recommendations at the 60^{th} auction (which is after the convergence). We can see that the quality of the ten shortlisted recommendations has an overall tendency to decrease in most cases (although there are some exceptions). Figure 5(b) shows the average user perceived quality of ten continuous auctions after the convergence (from 60^{th} to 69^{th}). By averaging over these auctions, we can see that the user perceived quality decreases monotonically. Thus, Figure 5 tells us the important information that our market mechanism is indeed capable of shortlisting the best recommendations in decreasing order of user perceived quality.

Clear Incentive. Now we aim to see if the recommending agents can relate their bids to the internal quality of their recommendations (meaning can an agent generate a steady strategy profile?). Again, we use the results from the price

[2]We believe both of these assumptions are reasonable as an initial point of departure. In the future, however, we seek to assess their validity in practice.

[3]In summary, AT and ST effect the number of recommendations being rewarded (because more agents are rewarded if their values are small). If an agent is frequently rewarded then it obtains more learning feedback and so can chase the equilibrium faster. A further effect of AT and ST is that the equilibrium price stability trades off with the speed of price convergence. This is because with a high value of AT and ST, only a small amount of learning information is available for the agents (because fewer agents are rewarded). Hence the agents need more adjustments to chase the equilibrium. The variables Y, Z and ΔP also effect an agent's speed of chasing the equilibrium. With high value of these variables, an agent alters its price rapidly to reach the equilibrium price. Again, however, this trades off with the stability of its bids.

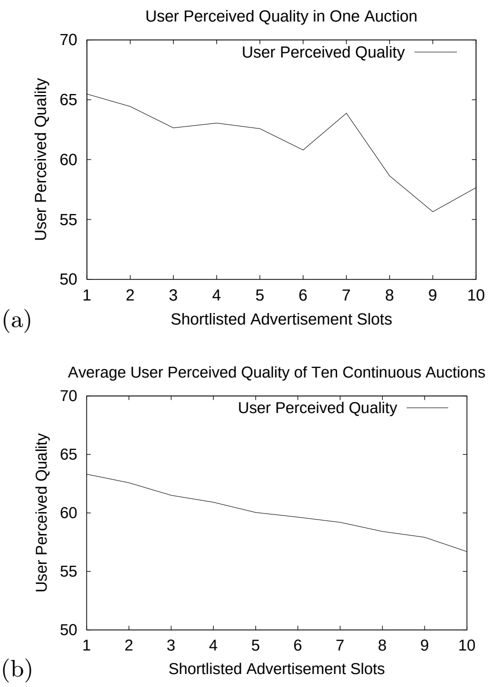

Figure 5: User Perceived Quality of Shortlisted Recommendations

convergence experiments. In this case, each recommending agent builds up its strategy profile from its knowledge about the bids with respect to its 20 internal quality segments. Specifically, Figure 6(a) shows the bidding prices for different segments of the recommending agent with the internal quality distribution $N(0.65, 0.1^2)$ in the above simulation. From Figure 6(a), we can see that the price from the segment 0.65∼0.70 (with the majority of samples) keeps rising before the 60^{th} auction (the market convergence) and oscillates around 160 (the equilibrium price for recommendations from the segment 0.65∼0.70) afterwards. This is because the agent raises its price from a low initial bid to chase the equilibrium price before the market converges and alters its price to minimize the difference between its bids and the equilibrium price so as to maximize its revenue. From Figure 6(a), we can also see that the agent's prices for segments 0.55∼0.60 and 0.75∼0.80 converge at about 130 and 170 respectively. Figure 6(a) indicates that the agent is capable of "learning" from the marketplace to alter its bids to certain levels in order to chase the equilibrium price. The solid line in Figure 6(b) (marked "Bidder 1") plots this agent's strategy profile and the dashed and dotted lines represent the strategy profiles of the two recommending agents with the internal quality distributions $N(0.5, 0.1^2)$ and $N(0.35, 0.1^2)$. From our observation of various simulations, a recommending agent's strategy profile changes quickly before the market converges and then becomes relatively stable after the convergence. This tells us that the marketplace delivers a clear incentive to recommending agents enabling them to bid rationally and that agents with different recommendation methods are capable of relating their bids to their internal qualities by forming a steady strategy profile.

Now we are going to remove the second assumption made when designing the user agent in subsection 3.1. Once the

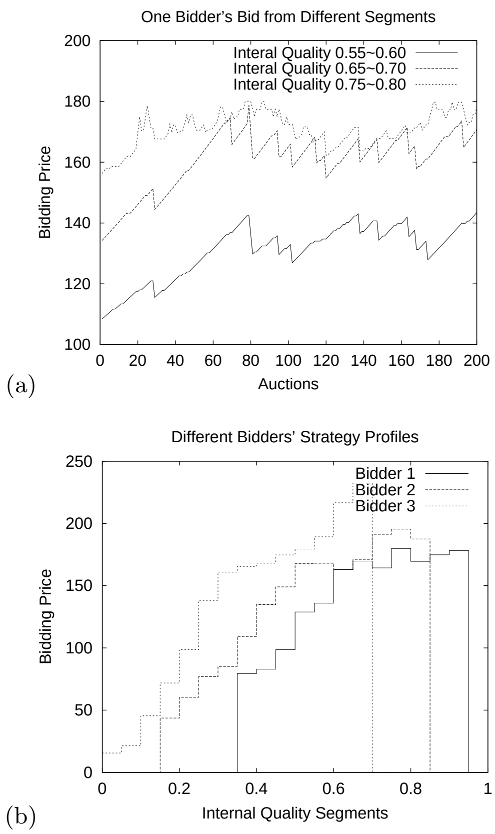

Figure 6: Strategy Profile

first assumption holds (meaning there is a steady relationship $Q = \mathbf{f}(q)$ given that $q \sim N(\mu, \sigma^2)$) irrespect of whatever $\mathbf{f}$ is (whether $k > 0$ or not and whether $\mathbf{f}$ is linear or not), given a specific value of q, there a steady value of Q ($= \mathbf{f}(q)$) always arises. Therefore, an agent with this specific method can always alter its price of q to the equilibrium price with respect to Q, if there are sufficient samples from the internal quality segment containing q. For example, if the user agent assigns Q to recommendations from the agent in Figure 6(a) following a behaviour of $Q = k \cdot q + b$ ($k < 0$), the solid line in Figure 6(b) will have an overall tendency to decrease. Therefore, the recommending agents can always generate their strategy profiles without assumption 2 once assumption 1 holds.

Stability. To evaluate the stability of the market with respect to bidding strategies, we now consider what happens if some of the agents take a greedy strategy (bid as much as possible to outbid others). To this end, we select one recommending agent as the greedy bidder and the other agents still take the strategy introduced in subsection 2.2. All recommending agents are endowed with an initial credit of 65535. The greedy bidder always bids much higher than the others to get its recommendations shortlisted so as to make profit. However, this greedy bidder does not receive any more rewards from its recommendations when compared with the rewarded recommendations provided by the other non-greedy bidders. This is because the reward is not based on the bid price, but rather on the user perceived quality. With the same amount of reward with respect to the same

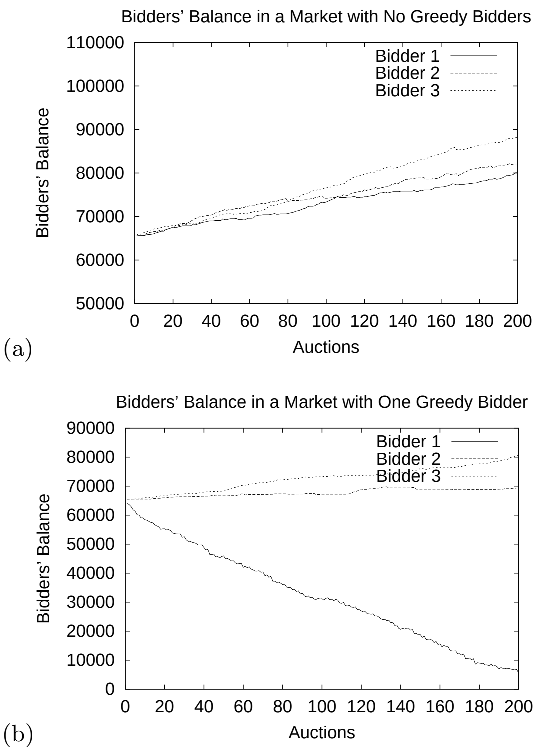

Figure 7: Greedy versus Non-Greedy Bidder

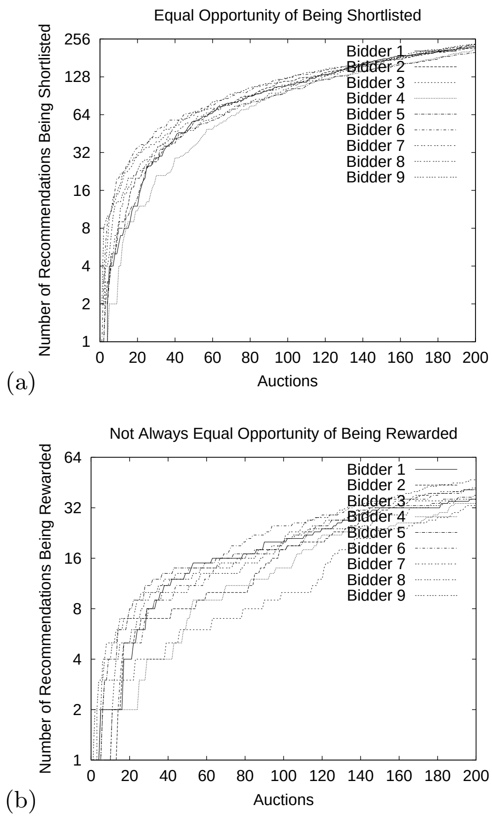

Figure 8: Opportunity of Being Shortlisted and Rewarded

level of user perceived quality, however, the greedy bidder pays much more for each of its shortlisted recommendations. Therefore, the greedy bidder goes bankrupt over time as shown in Figure 7(b), while the other non-greedy bidders keep increasing their balance steadily. In comparison, when no greedy bidders participate, all recommending agents keep increasing their balance as shown in Figure 7(a). Therefore, this experiment indicates that no greedy bidders can survive in our system.

Fairness. We expect the market to be fair to all recommending agents irrespectively of the recommendation method they use. To this end, we organize nine recommending agents taking three different recommendation methods in the marketplace. From Figure 8(a) it can be seen that the curves that represent the number being shortlisted (including both being rewarded and shortlisted but not rewarded) for each agent are close to each other. This, in turn, means all agents have an equal opportunity of being shortlisted. Thus the market is fair. Figure 8(b) shows that Bidder 3 is rewarded fewer times than the others (however this is because its recommendations are of lower user perceived quality). This, in turn, highlights the fact that a fair market does not necessarily guarantee equal opportunity of being rewarded. The opportunity of being rewarded depends on the user perceived quality.

3.3 System Properties

From the simulations demonstrated in subsection 3.2, we have shown that our marketplace can always reach an equilibrium when the shortlisted prices converge at different levels with respect to different user perceived quality levels. After the price converges, the marketplace is capable of shortlisting recommendations in decreasing order of user perceived quality. Also, after the convergence, all recommending agents can build up their strategy profiles based on their bidding and the knowledge about the user's preference with respect to their internal quality. Having achieved this, all agents can relate their bids to the internal quality of their recommendations. In addition, the marketplace is stable and shows fairness to all recommending agents irrespectively of the bidding strategies and recommendation methods they utilize.

4. RELATED WORK

A number of information filtering tools have been developed to cope with the problem of information overload. For example, SIFT [15] is a large scale information dissemination service capable of directing information of users' interests to the subscribed users. [13] demonstrates a learning approach to personalized information filtering using relevance feedback and genetic algorithms. SIGMA uses a marketplace to assist the filtering agents to learn the user's interests when they query Usenet news articles [6]. However, these systems tend to filter based on document content and in many cases in our Web browsing domain issues such as quality, style and other machine unparsable properties are the key to giving good recommendations [12].

Thus, recommender systems [10] have been advocated. GroupLens [7] is a system using collaborative filtering [5] to make recommendations of Usenet news to help people find articles they will like in the huge stream of available articles. Letizia [8] was an early agent-based recommender system which is also coupled to a Web browser; it has the additional feature of scouting ahead of the user's current position on the Web. However, while such recommender systems tackle the weaknesses of content-based filtering techniques, each system employs a variety of techniques that are more or less successful for particular users in particular contexts. For this reason, we believe effective recommender systems should incorporate a wide variety of such techniques and that some form of overarching framework should be put in place to coordinate the various recommendations so that only the best of them (from whatever source) are presented to the user. Hence, our use of the marketplace is to perform this role.

The most related work to our own is that of [1] which uses a market to competitively allocate consumers' attention space. This work develops an adaptive bidding strategy that can learn the consumer's preferences. However, the recommendations in this work are not the same as those of information recommender systems. Specifically, they are more general concepts about banners or common products. In comparison, recommendations in information systems specifically means the items that are stored and transmitted over the Internet and the World Wide Web. On the other hand, the marketplace of [1] is not targeted specifically at recommendations, but rather at the allocation of consumer's attention space.

In summary, with respect to the two perspectives of quality illustrated in this paper (internal and user perceived), content-based filtering systems only take internal quality into account whereas collaborative filtering systems predominantly rely on user perceived quality. In contrast to all these works, our market-based approach uses both measures and correlates them in order to produce effective recommendations.

5. CONCLUSIONS AND FUTURE WORK

This paper has outlined the design and evaluation of a recommender system that uses market-based techniques to incorporate multiple recommendation methods into a coherent framework. Our various simulations have shown that the marketplace works efficiently in making good recommendations across multiple recommendation methods and the theoretical properties of the mechanism (as designed in [14]) do indeed hold. Specifically, the marketplace is capable of shortlisting the best recommendations in decreasing order of user perceived quality. The marketplace is also capable of giving the recommending agents incentives to bid rationally in chasing the equilibrium price with respect to the user perceived quality of their recommendations. This in turn means that the marketplace delivers to the bidding agents a means of relating their bids to their internal quality. Moreover, the market is stable and is fair to all recommending agents. Having shown these results in simulation, the next and final step is to test the system's operation with real users in a practical application context.

6. REFERENCES

[1] Sander M. Bohte, Enrico Gerding, and Han La Poutré. Competitive market-based allocation of consumer attention space. In *Proc. on Electronic Commerce*, pages 202–205, US, 2001.

[2] Scott H. Clearwater, editor. *Market-Based Control. A Paradigm for Distributed Resource Allocation*. World Scientific, 1996.

[3] David De Roure, Wendy Hall, Siegfried Reich, and Aggelos Pikrakis *et al.* Memoir - an open framework for enhanced navigation of distributed information. *Information Processing and Management*, 37:53–74, 2001.

[4] Samhaa El-Beltagy, Wendy Hall, David De Roure, and Leslie Carr. Linking in context. In *Proc. on Hypertext and Hypermedia*, pages 151–160, Denmark, 2001.

[5] David Goldberg, David Nichols, Brain Oki, and Douglas Terry. Using collaborative filtering to weave an information tapestry. *Comm of the ACM*, 35(12):61–70, 1992.

[6] Grigoris J. Karakoulas and Innes A. Ferguson. Sigma: Integrating learning techniques in computational markets for information filtering. In *AAAI 1996 Spring Symposium on Machine Learning in Information Access*, Stanford, CA, 1996.

[7] J. Konstan, B. Miller, D. Maltz, and J. Herlocker *et al.* Grouplens: applying collaborative filtering to usenet news. *Comm of the ACM*, 40(3):77–87, 1997.

[8] Henry Lieberman. Letizia: An agent that assists web browsing. In *Proc of the International Joint Conference on Artificial Intelligence*, Canada, 1995.

[9] Luc Moreau, Norliza Zaini, Jing Zhou, and Nicholas R. Jennings *et al.* A market-based recommender system. In *Proc. of the AOIS02 Workshop*, Italy, 2002.

[10] Paul Resnick and Hal R. Varian. Recommender Systems. *Communications of the ACM*, 40(3):56–58, 1997.

[11] G. Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[12] Upendra Shardanand and Pattie Maes. Social information filtering: algorithms for automating word of mouth. In *Proc. on Human factors in computing systems*, pages 210–217, 1995.

[13] Beerud Sheth and Pattie Maes. Evolving agents for personalized information filtering. In *Proc on Artificial Intelligence for Applications*, pages 345–352, US, 1993.

[14] Yan Zheng Wei, Luc Moreau, and Nicholas R. Jennings. Recommender systems: A market-based design. In *Proc on Autonomous Agents and Multiagent Systems*, Australia, 2003.

[15] T. Yan and H. Garcia-Molina. SIFT—A tool for wide-area information dissemination. In *Proc 1995 USENIX Technical Conf*, pages 177–186, US, 1995.