

The Information Cost of Manipulation-Resistance in Recommender Systems

Paul Resnick

School of Information, University of Michigan
Ann Arbor, MI 48109 USA
presnick@umich.edu

Rahul Sami

School of Information, University of Michigan
Ann Arbor, MI 48109 USA
rsami@umich.edu

ABSTRACT

Attackers may seek to manipulate recommender systems in order to promote or suppress certain items. Existing defenses based on analysis of ratings also discard useful information from honest raters. In this paper, we show that this is unavoidable and provide a lower bound on how much information must be discarded. We use an information-theoretic framework to exhibit a fundamental tradeoff between manipulation-resistance and optimal use of genuine ratings in recommender systems. We define a recommender system to be (n, c) -robust if an attacker with n sybil identities cannot cause more than a limited amount c units of damage to predictions. We prove that any robust recommender system must also discard $\Omega(\log \frac{n}{c})$ units of useful information from each genuine rater.

Categories and Subject Descriptors

I.2.6 [Computing Methodologies]: Artificial Intelligence—Learning

General Terms

Algorithms, Reliability

Keywords

Recommender systems, manipulation-resistance, shilling, information loss

1. INTRODUCTION

Content posted on the Internet is not of uniform quality, nor is it equally interesting to different audiences. Recommender systems guide people to items they are likely to like, based on their own and other people’s subjective reactions.

Authors and other parties often want to direct attention to particular items. Google, Yahoo!, and others channel this into a multi-billion dollar advertising marketplace. But to the extent that people rely on recommender systems to guide

their attention, there are also natural incentives for promoters to manipulate the recommendations. An attacker may rate strategically rather than honestly and may introduce multiple entities, sometimes called *sybils* or *shills* [11, 6], to rate on behalf of the attacker.

The general scenario we analyze is that a sequence of raters rate an item, and then a prediction is made about a target person’s reaction to the item. We will refer to people’s opinions generically as ratings, whether users explicitly enter them, in the form of ratings or tags, or the system infers them from implicit behavioral indicators such as purchases, read times, bookmarks, or links.

Some of the raters are honest; they acquire private information about each item and report it honestly. Some are sybils under the control of an attacker, who reports fake ratings to manipulate the predictions. A recommender system combines the ratings it has received so far to predict whether a target person will like each item. For example, an additional positive rating might increase the predicted probability that the target will like the item, while an additional negative rating might reduce the predicted probability. In order to personalize predictions and to resist manipulation, the recommender may allow some raters to influence the prediction more than others, depending on each rater’s pattern of ratings of other items and how useful those ratings have been.

Resnick and Sami [14] presented a particular manipulation-resistance algorithm, the Influence Limiter, that can be overlaid on any recommender. It gives only a tiny influence to a new entity, then increases that influence as the entity provides informative ratings. The algorithm is provably resistant to any attack involving a bounded number of sybils. The influence limits, however, create an inefficiency: the recommender throws away information from new raters who are honest and informative but who have not yet proven themselves to be so.

Several other authors have suggested using statistical metrics on ratings to distinguish “attack” identities from “regular” identities, and eliminate the former [3, 12, 7, 15, 8]. Mobasher *et al.* [9] survey this literature and classify attack strategies. Any process of weeding out attackers based on the distribution of their ratings, however, risks throwing away information from informative raters who are misclassified as attackers.

This paper asks whether the information losses incurred in these approaches are necessary. Is there a fundamental tradeoff between resistance to manipulation by an attacker with a large but bounded number of sybils, and use of infor-

©ACM, 2008. This is the author’s version of the work. It is posted here by permission of the ACM for your personal use. Not for redistribution. The definitive version appears in the Proceedings of the ACM Recsys08 Conference, Lausanne, Switzerland, October 2008.
Copyright 2008 ACM .

mation from informative raters? The answer is yes. Indeed, preventing damage by an attacker who merely injects noise in the form of random guesses requires discarding the same order or magnitude of information as that discarded by the Influence Limiter algorithm.

We evaluate a recommender’s predictions by applying a scoring rule or loss function that compares the predictions made to the eventual evaluations that a target user gives to the items. The information loss of a recommender is the expected increase in loss from using it rather than an ideal recommender.

We prove two variants of an information loss lower bound. The first describes how many ratings a manipulation-resistant algorithm must monitor before it allows a rater to have a large influence on a person’s recommendation. The second describes the minimum information loss per honest rater that a manipulation-resistant algorithm must incur in the worst case. It is a function of the maximum number of sybils that the algorithm will be resistant to, and of the maximum damage they can be allowed to cause. Thus, an immediate consequence is that no scheme can be resistant to manipulation by an unbounded number of sybils without throwing away all information from honest raters.

2. THE MODEL

In this section, we detail and justify a formal model that enables the analysis of manipulation-resistance as well as information efficiency of recommender algorithms. We present the model in three stages. In section 2.1, we introduce a formal model of a recommender’s predictions and an information-theoretic measure of the error in those predictions. Section 2.2 provides a formal model of the information that honest raters acquire and reveal to the recommender through their ratings. Section 2.3 introduces an attack model, a measure of the damage that an attacker causes in terms of increased error of the predictions the recommender generates, and a formal definition of (n, c) -robustness.

2.1 Predictions and Scores

The output of the recommender is a prediction about the reaction that a particular target person will have to an item. For simplicity, we assume that a prediction is expressed as a probability that the target will like an item, and a target eventually makes a binary report of liking the item or not. The binary reports do not limit the generalizability of our results: any scheme that is resistant to manipulation of predictions about finer-grained reports from targets must also be resistant to manipulation of predictions over more-restricted binary reports.¹

More formally, there is a space Ω of possible states (item types) and a label space $\mathcal{L} = \{HI, LO\}$ of possible responses to an item from a target. As we shall see in the next section, we can, without losing generality, confine all uncertainty in the model to uncertainty about the state; the state deter-

¹Our model generalizes naturally to systems where targets select from a fixed set of labels, such as 1-5 stars. In that case, a recommender would have to predict the probability of each label being chosen (e.g., 30% chance of 5 stars; 20% chance of 4; 50% chance of 3). Many existing recommenders predict only the mean (e.g., 3.8), which could be extended in a variety of ways to predict probabilities for each of the individual labels; our lower bound implies a limit on the effectiveness of any such extension.

mines the reactions of all raters and potential targets with certainty. In particular, for each $\omega \in \Omega$ there is a corresponding value $l(\omega) \in HI, LO$ that describes the target’s reaction to items in that state. A prediction expresses the probability $q \in [0, 1]$ of a *HI* label.

Since predictions are probabilistic, they are not simply right or wrong. A prediction that assigns a higher probability to the outcome that occurs is more correct, or has less error, than one that assigns a lower probability. We employ the quadratic scoring rule to assign a loss or error score to a prediction, after the target reports liking the item or not. Formally:

$$L(HI, q) = -(1 - q)^2; \quad L(LO, q) = -q^2$$

Note that the loss or error is 0 when an extreme prediction of 0 or 1 is made and the target agrees with the prediction. The worst error of -1 occurs when an extreme prediction is made that the target disagrees with. This error measure corresponds to using the expected squared error (the variance) as a measure of uncertainty, which has a long history in statistics. It also corresponds to the use of mean squared error as a measure of prediction accuracy in recommender systems, though only for systems that make predictions on a 0-1 scale and have targets report binary outcomes.

2.2 Partial Information Model

We now proceed to define the damage of an attacker and the information loss of a recommender system. Intuitively, the damage is the increase in expected error score of the predictions made with the attacker present over those made without the attacker. The information loss is the increase in the expected error score of the predictions made by the actual recommender over those made by an ideal recommender.

More precise definitions, however, depend on a specification of the environment of a sequence of item ratings. How much contribution or damage a rater will make may depend on how much information other raters will provide. We need a model of the partial information provided by each additional rater. We use a standard model of information partitions, consistent with the model in [14], though we have simplified some of the notation and adjusted the exposition.

There is a set I of items to be rated. The state $\omega \in \Omega$ of each item $i \in I$ is drawn independently according to a distribution defined by probability mass function $p : \Omega \rightarrow [0, 1]$ that gives the relative likelihood of different states.

There is a set J of raters. Each rater r who evaluates an item receives a signal, some partial information about the item. The state ω determines with certainty the signal that rater r receives, but many states may produce the same signal.² Thus, each realized signal picks out a subset of Ω , the set of states that would cause that rater to receive that signal. Subsets of states yielding different signals are non-overlapping and exhaustive (in each state the rater sees exactly one signal) so they form a partition.

For the purposes of analyzing the information available to

²Some readers may be more familiar with a model of partial information where a state determines a probability distribution over signals for each rater, rather than a fixed signal. In our model, there would instead be more states, each corresponding to one of the possible realized signals; the relative likelihood of the states would correspond to the probability distribution over signals in the other model.

an ideal recommender, we assume that a rater’s report fully reveals the component Ω . We do not model explicitly the mechanics of how a rater reports (e.g., on a 1-5 scale). In practice, the inability for a rating to reflect all the private information the rater has acquired may be one source of information loss for a recommender.

Different raters may notice different things about items or have different tastes. We model this as each rater r having a type $\pi(r)$, where $\pi(r)$ is a partition of Ω into components corresponding to the different signals that rater r could receive. Let Π denote the space of all possible rater types.

A rater sequence $X = (r_1, i_1), (r_2, i_2), \dots, (r_T, i_T)$ specifies, for each time period t , the unique identifier for a rater r_t who will rate which item i_t . Note that the same item may appear multiple times ($i_{t_1} = i_{t_2}$), and the same rater may appear multiple times ($r_{t_1} = r_{t_2}$), but each pair is distinct (i.e., each rater examines each item only once). We denote the subsequence consisting of the first t time steps by X_t .

A rating history $Y = (X, \{y_1, \dots, y_T\})$ is a combination of a rater sequence and a realization of the raters’ signals. The rating y_t reveals the signal that r_t received for item i_t , and thus the component $s_{i_t}(y_t)$ consisting of those item types that are still possible given r_t ’s realized signal.

Each realized rating of item i provides information about whether the target will like the item, by eliminating some of the previously possible states of the item. Given the set of ratings on item i in a rating history Y , by raters with known types given by π , item i ’s state must lie in a subset $\hat{s}(i, Y, \pi)$ that is the intersection of the components identified by each of the ratings of i .

All the possible subsets that could arise through different realized ratings for i in X_t form a partition of Ω , which we denote by $\hat{\pi}_i(X_t)$. Note that each component of $\hat{\pi}_i(X_t)$ is either a component of $\hat{\pi}_i(X_{t-1})$ (if the final rating reveals no new information) or is a strict subset of a component of $\hat{\pi}_i(X_{t-1})$ (if the final rating eliminates some possible states). Thus, we say that $\hat{\pi}_i(X_t)$ is a *refinement* of $\hat{\pi}_i(X_{t-1})$.

We will refer to a hypothetical recommender that makes the best predictions possible, given full information about the rater types $\pi(X)$ and the entire rating history Y , as an *ideal recommender*. The best prediction of whether the target will like item i is the probability of a HI label, conditional on the rating history and the rater types: that is, conditional on the state being in $\hat{s}(i, Y, \pi)$. We denote this optimal prediction by

$$q(i, Y, \pi) = \frac{\sum_{\omega \in \hat{s}, l(\omega) = HI} p(\omega)}{\sum_{\omega \in \hat{s}} p(\omega)}$$

The prior probability that the target will like an item i , before receiving ratings, is $q(i, Y_0, \pi) = \sum_{\omega \in \Omega, l(\omega) = HI} p(\omega)$.

Any joint distribution of rater and target preferences can be represented within this model. For example, suppose the system has a single rater r_1 in addition to the target. Further, suppose that r_1 can discern two kinds of items, which we denote $'+$ and $'-$. Say the target likes 60% of items that r_1 labels $'+$, and 40% of items that r_1 labels $'-$. We can model this with a state space $\Omega = \{+H, +L, -H, -L\}$, prior probabilities $p(+H) = p(-L) = 0.3$, $p(-H) = p(+L) = 0.2$, partition $\pi_1 = \{\{+H, +L\}, \{-H, -L\}\}$, and state labels $l(+H) = l(-H) = HI$, $l(+L) = l(-L) = LO$. The prior probability of a HI label $q(i, Y_0, \pi)$ is $.3 + .2 = .5$. The prior probability of a $'+$ rating is also $.5$. After a $'+$ rating, $q(i, '+', \pi) = .3 / (.3 + .2) = .6$.

We summarize our model of the underlying information setting with the following definition:

DEFINITION 1. A **partial information structure** $\mathcal{P} = (\Omega, p, I, J, \pi, X)$ consists of a state space Ω , a prior probability distribution p , a set of items I , a set of rater ids J , a function π that maps rater ids to types, and a sequence of (item id, rater id) pairs X specifying the ratings to be acquired.

2.3 Random Noise Attacks

We now introduce attacks into our model of the recommendation process. We model attackers as creating sybil identities and injecting strategically chosen ratings into the rating sequence. We will prove lower bounds on the information loss suffered by any recommender algorithm that is robust in the face of such attacks. That is, we will show that, for any algorithm, there exists a partial information structure and an attack strategy such that either the expected damage for the attacker is high or the expected information loss absent an attacker is high. In fact, our results only require robustness against a special class of attacks, *random guessing attacks*, which are defined below. Naturally, the lower bound applies *a fortiori* to recommenders that are resistant to all possible attack types.

DEFINITION 2. An **n -sybil random noise attack strategy** $A = (J_A, \pi', X')$ on an information structure $\mathcal{P} = (\Omega, p, I, J, \pi, X)$ is comprised of three elements.

- J_A is a set of n sybil identities.
- A function π' specifies for each sybil identity $r' \in J_A$ a type $\pi'(r') \in \Pi$ corresponding to some real type.
- X' is an expanded rater sequence that is constructed by inserting pairs of the form (r_t, i_t) at any number of locations in X , where $r_t \in J_A$ and $i_t \in I$.

For each time t such that $r_t \in J_A$, the attacker receives no signal but generates a fake rating. It selects randomly among the signals that the real type $\pi'(r_t)$ could have received, if there were such a real type acting at this point in the sequence. That is, the ratings from previous non-attacker entities in the rating sequence determine the set of states that are still possible. One of these states is selected at random, according to the relative likelihood of those states given by the original probability distribution p . The rating reported is then the one that identifies the component of $\pi'(r_t)$ that contains the selected state.

Expected Damage:

Next we quantify the damage caused by an attack A on an information structure \mathcal{P} . Intuitively, it is the expected difference in the loss between the predictions that a recommender would make with and without the addition of the attacker’s ratings.

Up to this point, we have concentrated on the best possible prediction that an ideal recommender system could make given a sequence of ratings *and* information provided by knowledge of the raters’ types, i.e., the structure of the raters’ partitions. An actual recommender system sees the ratings and target labels, but it does not know anything about the underlying partition structure except information that can be inferred from the rating history. Let ℓ denote

a vector of the target’s ultimate labels, one for each item i . We then model an actual recommender system R by a prediction function $q^R(i, Y, \ell)$, which processes a realized rating history and target labels to predict, for each item i , a probability of the target labeling item i “HI”. We assume that the target labels each item right after its last rating in the sequence Y , and that the recommender R cannot access the value of a label l_i until it is labeled. The quality of the prediction for item i is assessed just before the target labels item i ; the recommender may condition its prediction on the rating history and revealed labels up to that time.

Let $\omega = \{\omega_i\}$ be a vector of states for all items in I . Given the states ω , the signals of all genuine raters are determined; let $Y(\omega)$ be the realized history for the genuine raters. As the states are independently drawn, $p(\omega) = \prod p(\omega_i)$.

Let Z be any realization of the random ratings made by an attacker in X' . The attack ratings Z may depend on earlier ratings by genuine raters. However, these raters’ ratings are completely determined by ω , and so the conditional probability $P(Z|\omega)$ is determined by the type π' the attacker is posing as. We use $p_{\pi', \omega}(Z)$ to denote this conditional probability. Let $Y'(\omega, Z)$ be the rating history obtained by merging a genuine history $Y(\omega)$ with the attack ratings Z .

DEFINITION 3. The **expected damage** $ED(R, \mathcal{P}, A)$ of an attack strategy A on a recommender algorithm R for an information structure \mathcal{P} is defined as:

$$ED(R, \mathcal{P}, A) \stackrel{\text{def}}{=} \sum_{i \in I} \sum_{\omega_i \in \Omega} p(\omega) \sum_Z p_{\pi', \omega}(Z) [L[l(\omega_i), q^R(i, Y'(\omega, Z), \ell)] - L[l(\omega), q^R(i, Y(\omega), \ell)]]$$

Robustness:

We are now ready to state our definition of robustness.

DEFINITION 4. A recommender system R is **(n, c)-robust** against random noise attacks iff, for any partial information structure \mathcal{P} , and any n -sybil random noise attack strategy A on that structure, $ED(R, A, \mathcal{P}) \leq c$.

Information Loss:

Finally, we define the expected information loss $IL(R, \mathcal{P}, A)$ that a given recommender R incurs for a rater sequence X' that may include some attacker sybils. Intuitively, it is the expected difference between the loss on the predictions q^R made by the recommender and the loss on predictions q that an ideal recommender would make, knowing the types of all the honest raters and which raters were attacker sybils.

DEFINITION 5. The **information loss** $IL(R, \mathcal{P}, A)$ of a recommender algorithm R on an information structure \mathcal{P} together with attack A is defined as:

$$IL(R, \mathcal{P}, A) \stackrel{\text{def}}{=} \sum_{i \in I} \sum_{\omega_i \in \Omega} p(\omega) \sum_Z p_{\pi', \omega}(Z) [L[l(\omega_i), q^R(i, Y'(\omega, Z), \ell)] - L[l(\omega), q(i, Y(\omega), \pi)]]$$

Information loss for a practical recommender may arise for several reasons. In order to provide a simple user interface, a recommender may not elicit ratings in a form that fully reveals the rater’s partition. It may have an incorrect way of interpreting a rater’s ratings, because it does not yet have enough experience with that rater to infer its type. Or the recommender may include features intended to resist manipulation that also cause it to inefficiently use information from some raters.

3. LOWER BOUNDS ON INFORMATION LOSS

The key intuition behind our lower bounds is the observation that no algorithm that has observed only a small number of ratings can distinguish (with high probability) a genuine rater from a random-noise sybil. Only after a sufficiently long sequence of observations can we reliably detect that the former moves the predictions in the right direction more often. A manipulation-resistant algorithm must keep the expected influence of a random-noise sybil very small to prevent an attacker from using n of them to cause significant damage. Thus, it must also limit the influence of a genuine rater for a number of rounds.

We use a simple family of information structures to construct our proofs. This family of instances is sufficient for the *worst-case* information loss result. In section 4, we discuss generalizations of this result to other information structures, as well as other loss functions. We begin by proving a lower bound in the context of 1-sybil attacks; in section 3.2, we use this to prove our main result, on n -sybil attacks.

3.1 Lower bound on 1-sybil attacks

In this section, we prove a lower bound on information loss in 1-sybil attacks. Throughout this section, suppose that we have been given a damage bound d . We consider a family of information structures $\mathcal{P}(b)$, where the parameter b denotes the magnitude of information the informed raters have. We begin by defining the base structure $\mathcal{P}(b) = (\Omega, p_b, I, X(b), \pi_b)$, as follows. The state space is defined as $\Omega = \{+H, +L, -H, -L\}$; $p_b(+H) = p_b(-L) = 0.25 + b/2$, $p_b(-H) = p_b(+L) = 0.25 - b/2$. The target assigns labels $l(+H) = l(-H) = HI$, $l(+L) = l(-L) = LO$. I is a set of $2m_b$ items, where m_b will be specified later. There is only one rater r and thus each item gets only one rating. The rater rates half the items: $X = (r, i_1), (r, i_2), \dots, (r, i_{m_b})$. The rater’s partition π is such that she receives the signal $y_i = '+'$ when the state is either $+H$ or $+L$. Note that $p_b(l(\omega) = HI | y_i = '+') = p_b(+H) / (p_b(+H) + p_b(+L)) = 0.5 + b$, and $p_b(l(\omega) = HI | y_i = '-') = 0.5 - b$. The prior probability that the target’s label is HI is 0.5, as is the probability that $y_i = '+'$.

We next introduce a corresponding family of attacks $A(b) = (J_A, \pi'_b, X'(b))$. The set J_A consists of only one sybil identity r' . The partition $\pi'_b(r') = \pi_b(r)$ is as described above. In other words, every time r' rates an item i' , she will randomly report $y_{i'} \in \{+, -\}$, each with probability 0.5.

The sybil rates the other m_b items not rated by the genuine rater. The rater sequence $X'(b)$ is constructed randomly, as follows: The attacker flips a fair coin, and if it is heads, she inserts all her m_b ratings *before* r ’s first rating. If the coin comes up tails, she instead inserts all her ratings *after* r ’s last rating.

We will now prove that, for any recommender R , we can set a value m_b such that either $ED(R, \mathcal{P}(b), A(b)) \geq d$, or $IL(R, \mathcal{P}(b), A(b))$ is $\Omega(\log \frac{1}{d})$.

Consider two extreme options for the recommender in making predictions for an item, given the single rating available, which might be from a genuine rater or from an attacker. One extreme option is to ignore the rating and predict 0.5. This avoids any expected damage, since the rating has no effect. When the rater is a sybil, ignoring the rating also adds nothing to the expected information loss. When the rater is genuine, however, there is an expected informa-

tion loss from not moving to the correct prediction of either $0.5 + b$ or $0.5 - b$.

The other extreme option is for the recommender to treat the rater as genuine, predicting $0.5 + b$ or $0.5 - b$ depending on the rating. Here, if the rater is honest, there is no damage or information loss, since the prediction matches both the ideal prediction and the prediction that would have been made in a scenario where all the raters were genuine. If, however, the rater is a sybil, and the rating is just noise, there will be expected information loss and expected damage.

Of course, the recommender need not make only these extreme choices. It can partially incorporate a rating, moving the prediction only partway from 0.5 to $0.5 + b$ or $0.5 - b$. This will increase the information loss when the rater is genuine, and increase both damage and loss when it is not.

A key ingredient of our lower bound is that we bound the *influence* a rater can have after a sequence of ratings. The following definition is specialized to our setting, in which each item is rated by a single rater:

DEFINITION 6. Consider any realized sequences Y' and ℓ of ratings and labels in the structure $\mathcal{P}(b)$ with attack $A(b)$, and suppose that (r_t, i_t) is the rating at time t . Let Y'_{t-1} denote the realized ratings upto time $t - 1$, and ℓ_{t-1} denote the subset of labels that have been revealed before the t^{th} rating. Then, we define the influence of rater r_t on item i_t as the value β given by:

$$\beta = \max \left\{ \gamma \mid \begin{array}{l} q^R(i_t, Y'_{t-1}, \ell_{t-1} | y_t = '+') \geq 0.5 + \gamma, \\ q^R(i_t, Y'_{t-1}, \ell_{t-1} | y_t = '-') \leq 0.5 - \gamma \end{array} \right\}$$

Hereafter, we assume that each rater can effect the same change to the predicted probability in either direction, *i.e.*, upward or downward. This allows for a simpler statement of the lower bound. Dropping this assumption would not alter our result in any significant way.

We first show that the influence affects both the damage (for the attacker r') and the loss (for the genuine rater r).

LEMMA 1. Suppose (r', i) is a rating by the attacker, and suppose the influence of r' on item i is β (for given R and Y'). Then, the expected damage on this item is β^2 .

PROOF. As this rating is entered by the random-guessing attacker, if the rating is $'+'$, the item will be labeled HI with probability 0.5 , and LO with probability 0.5 . By definition of the influence β , the recommender predicted a value $0.5 + \beta$ on item i . Thus, the expected loss is

$$0.5(0.5 - \beta)^2 + 0.5(0.5 + \beta)^2 = 0.25 + \beta^2$$

On the other hand, without the attacker's rating, there would have been no ratings on i , and so the recommender would have predicted 0.5 , for an expected loss of 0.25 . Thus, the expected damage of this rating is $0.25 + \beta^2 - 0.25 = \beta^2$. The case when the rating is $'-'$ is symmetric. \square

LEMMA 2. Suppose (r, i) is a rating by the genuine rater, and suppose the influence of r on item i is $\beta < b$ (for given R and Y'). Then, the information loss on this item is $(b - \beta)^2$.

PROOF. The additional damage caused due to the restricted influence can be calculated as the difference between the expected score of the restricted predictions and the expected score of the optimal predictions. Observe that when rater r reports signal $'+'$, the probability that the label is HI is $0.5 + b$ (as is the optimal prediction $q(i, Y', \pi)$), but

the prediction $q^R(i, Y', \ell) = 0.5 + \beta$. Thus, the expected loss on this item is:

$$\begin{aligned} & (0.5 + b)[(0.5 - \beta)^2 - (0.5 - b)^2] \\ & + (0.5 - b)[(0.5 + \beta)^2 - (0.5 + b)^2] \\ & = (b - \beta)^2 \end{aligned}$$

The case in which r reports $'-'$ is symmetric. \square

Now, because of the randomized sequence X' , the recommender algorithm R cannot tell beforehand if the first rater r_1 is the sybil r' or the genuine rater r . Moreover, for the first m_b items, the recommender has no useful information about the second rater (because all its ratings come after the first rater's ratings.)

Now, consider the time step just after $m < m_b$ items have been rated. At any point of time, the only information about the first rater r_1 that is available to the recommender algorithm is the past record of its ratings, and the corresponding target labels on the items. The information u_i available to the algorithm about each item $i \leq m$ can thus be represented by one of the four symbols $\{+H, +L, -H, -L\}$, where $+H$ denotes that r_1 rated $+$ and the target ultimately labeled the item HI , etc. Note that when the rater is honest, the information u_i is exactly the state ω , but when the rating is from a sybil, the information $u_i = +H$ could occur either in state $+H$ or $-H$.

For any item i , the probability of each outcome will be different depending on whether r_1 was the informative rater r , or the uninformative rater r' . We use $P_b()$ to denote the probability mass function for rater r , and $P_0()$ to denote the probability mass function for the random-guesser r' . Likewise, we use $E_0()$ and $E_b()$ to denote expectations with respect to probabilities P_0 and P_b respectively, and Var_0 and Cov_0 to denote variance and covariance with respect to probabilities P_0 . We note that

$$\begin{aligned} P_b(u_i = +H) &= P_b(u_i = -L) &= 0.5(0.5 + b) \\ P_b(u_i = -H) &= P_b(u_i = +L) &= 0.5(0.5 - b) \\ P_0(u_i = +H) &= P_0(u_i = -L) &= 0.25 \\ P_0(u_i = -H) &= P_0(u_i = +L) &= 0.25 \end{aligned}$$

We allow for the recommender algorithms to operate in a path-dependent way. That is, a rater whose first rating is in the correct direction and second in the wrong direction may be assigned a different influence than one whose first rating is in the wrong direction and second in the right direction. Thus, we cannot work with the expected number of guesses in the right or wrong direction; instead, we analyze the probabilities of individual paths of observations.

Fix a recommender algorithm. Suppose rater r_1 has a prediction-outcome history $\mathbf{u} = (u_1, u_2, \dots, u_m)$. Just before item $(m + 1)$ is labeled, the recommender algorithm determines what to predict on that item, depending on r_1 's rating. Thus, the recommender implicitly prescribes the influence of r_1 on item $(m + 1)$. This can depend on the entire path \mathbf{u} , but nothing else, and hence we use the notation $\text{Inf}(\mathbf{u})$ to denote this quantity. If the recommender algorithm is itself randomized, we can let $\text{Inf}(\mathbf{u})$ denote the expected value (over the recommender's randomization) of r_1 's influence on item $(m + 1)$, given history \mathbf{u} .

We prove a lower bound on the number of ratings needed for a rater to build her expected influence to a given level β , for any $(1, d)$ -robust algorithm. To do this, we first fix

a number m of items rated, and bound the expected influence of the honest rater after m rounds. The robustness property gives us an upper bound on $E_0(\text{Inf}(\mathbf{u}))$, the expected influence of the impersonator. We need to extend this to a bound on $E_b(\text{Inf}(\mathbf{u}))$, the expected influence of an honest rater. The two expectations can be quite different because certain sequences \mathbf{u} are much more likely to have come from an informed rater than an impersonator, *i.e.*, $P_b(\mathbf{u}) \gg P_0(\mathbf{u})$. To link them, we use the likelihood ratio function $g(\mathbf{u})$ defined as follows:

$$\begin{aligned} g_i(\mathbf{u}) &\stackrel{\text{def}}{=} \frac{P_b(u_i)}{P_0(u_i)} = (1+2b) \text{ if } u_i = +H \text{ or } -L \\ &= (1-2b) \text{ if } u_i = -H \text{ or } +L \\ g(\mathbf{u}) &\stackrel{\text{def}}{=} \frac{P_b(\mathbf{u})}{P_0(\mathbf{u})} = \prod_i g_i(\mathbf{u}) \end{aligned}$$

The following relation is immediate:

LEMMA 3.

$$E_0(g(\mathbf{u})\text{Inf}(\mathbf{u})) = E_b(\text{Inf}(\mathbf{u}))$$

PROOF. This follows from the definition of $g(\mathbf{u})$. \square

We now seek to prove an upper bound on $E_0(g(\mathbf{u})\text{Inf}(\mathbf{u}))$. To do this, we bound $E_0(g(\mathbf{u}))$ and $E_0(\text{Inf}(\mathbf{u}))$ separately, and then bound the covariance $\text{Cov}_0(g(\mathbf{u}), \text{Inf}(\mathbf{u}))$.

LEMMA 4. $E_0(g(\mathbf{u})) = 1$.

PROOF. First, note that $E_0(g_i(\mathbf{u})) = 0.5(1+2b) + 0.5(1-2b) = 1$. Using the independence of different items i , we have $E_0(g(\mathbf{u})) = \prod_i (E_0(g_i(\mathbf{u}))) = 1$ \square

LEMMA 5. For any $(1, d)$ -robust recommender algorithm, $E_0(\text{Inf}(\mathbf{u})) \leq \sqrt{2d}$.

PROOF. By Lemma 1, if $\text{Inf}(\mathbf{u}) = \beta$, and $r_1 = r'$, the expected damage on item i is β^2 . Thus, conditional on $r_1 = r'$, the expected damage on the m th item is $E_0([\text{Inf}(\mathbf{u})]^2)$. Taking into account the probability that $r_1 = r'$ is 0.5, the expected damage on the i th item is $0.5E_0([\text{Inf}(\mathbf{u})]^2)$. Noting the standard inequality $[E(x)]^2 \leq E(x^2)$, and the fact that the expected damage is no more than d , gives the result. \square

LEMMA 6. The covariance of $g(\mathbf{u})$ and $\text{Inf}(\mathbf{u})$ is bounded by:

$$\text{Cov}_0(g(\mathbf{u}), \text{Inf}(\mathbf{u})) < \sqrt{E_0(\text{Inf}(\mathbf{u}))} e^{2mb^2}$$

PROOF. We use the standard relationship $\text{Cov}(X, Y) \leq \sqrt{\text{Var}(X)\text{Var}(Y)}$. As $\text{Inf}(\mathbf{u}) \in [0, 1]$, we have

$$\text{Var}_0(\text{Inf}(\mathbf{u})) \leq E_0([\text{Inf}(\mathbf{u})]^2) \leq E_0(\text{Inf}(\mathbf{u}))$$

We now bound $\text{Var}_0(g(\mathbf{u}))$. First, note that $E_0([g_i(\mathbf{u})]^2) = 0.5(1+2b)^2 + 0.5(1-2b)^2 = 1 + 4b^2$. Taking logarithms, we have

$$\log E_0([g_i(\mathbf{u})]^2) = \log(1 + 4b^2) \leq 4b^2 \quad (1)$$

Thus, we have:

$$E_0([g(\mathbf{u})]^2) = \prod_i E_0([g_i(\mathbf{u})]^2) \leq e^{4mb^2}$$

This leads to the required bound on the variance:

$$\text{Var}_0(g(\mathbf{u})) = E_0([g(\mathbf{u})]^2) - E_0(g(\mathbf{u})) < e^{4mb^2} \quad \square$$

This leads to our main result on the influence growth rate:

THEOREM 7. In information structure $\mathcal{P}(b)$, with attack $A(b)$, and any $(1, d)$ -robust recommender algorithm R , for any $m \leq \frac{\log(\frac{\beta}{\sqrt{2d}} - 1)}{2b^2}$, the first rater r_1 has expected influence less than β .

PROOF. We seek to bound $E_b(\text{Inf}(\mathbf{u}))$ after m items have been rated. From Lemma 3, this is equivalent to bounding $E_0(g(\mathbf{u})\text{Inf}(\mathbf{u}))$.

From Lemma 5, we know that $E_0(\text{Inf}(\mathbf{u})) \leq \sqrt{2d}$, and from Lemma 4 we know that $E_0(g(\mathbf{u})) = 1$.

Now, consider any $m \leq \frac{\log(\frac{\beta}{\sqrt{2d}} - 1)}{2b^2}$. We must have $2mb^2 \leq \log(\frac{\beta}{\sqrt{2d}} - 1)$, and thus, $e^{2mb^2} \leq \frac{\beta}{\sqrt{2d}} - 1$.

From Lemma 6, we have:

$$\text{Cov}_0(g(\mathbf{u}), \text{Inf}(\mathbf{u})) < \sqrt{2d} e^{2mb^2}$$

Now, we are finally ready to bound the expected influence:

$$\begin{aligned} E_0(g(\mathbf{u})\text{Inf}(\mathbf{u})) &= E_0(g(\mathbf{u}))E_0(\text{Inf}(\mathbf{u})) + \text{Cov}_0(g(\mathbf{u}), \text{Inf}(\mathbf{u})) \\ &< \sqrt{2d} + \sqrt{2d} e^{2mb^2} \\ E_0(g(\mathbf{u})\text{Inf}(\mathbf{u})) &< \sqrt{2d} \frac{\beta}{\sqrt{2d}} = \beta \end{aligned}$$

This completes the proof. \square

Remark: An alternative way to derive the asymptotic form of this lower bound is by reduction to a widely studied problem in statistics: bounding the number of samples (items) required to test a hypothesis (the rater is informative) against an alternative (the rater is a random guesser) while achieving the desired limits on false positive (d) and false negative ($1/2$) rates. We accept the hypothesis whenever the actual influence is more than half the expected influence of an informative rater. The $(1, d)$ -robustness condition implies a bound on the expected influence of the random guesser (Lemma 5), which limits the false positive rate (the fraction of random guessers who would have influence above the threshold). Likewise, honest raters must pass the test at least half the time. Results on hypothesis testing (see [4, Sec. 11.8]) lead to a lower bound on the number of samples.

Total information loss:

Theorem 7 bounds the number of rounds required for the rater r_1 to gain influence β in any $(1, d)$ -robust algorithm. In order to get an information loss bound, we set β to $b/2$ in the statement of Theorem 7. Until this influence has been reached, we know that the rater will be effectively restricted. This leads to a lower bound on the total information loss due to limiting the influence of a rater:

THEOREM 8. Any $(1, d)$ -robust recommender algorithm R must have $IL(R, \mathcal{P}(b), A(b)) \geq \frac{\log \frac{1}{d} + \log(\frac{b^2}{32})}{32}$.

PROOF. With probability 0.5, the first rater is the informed rater r . We seek to prove a lower bound on the total additional information loss due to the restricted influence of rater r . Using the linearity of expectation, we have

$$E(\text{total additional loss}) = \sum_i E(\text{additional loss on } i)$$

First, we observe that for $b^2 < 32d$, the stated bound is negative and hence trivially true. Thus, it is sufficient to

consider the case in which $b^2 \geq 32d \Rightarrow b \geq 4\sqrt{2d}$. Setting $\beta = b/2$, we know from theorem 7 that the honest rater must have influence less than β for the first $m = \log(\frac{\beta}{\sqrt{2d}} - 1)/2b^2$ rounds. For any $i \leq m$, by Lemma 2, the expected additional loss the target incurs on item i due to the rater's restricted influence is at least $b^2/4$. Thus, the total additional loss, conditioned on rater r rating first, is at least $\frac{mb^2}{4}$. Taking into account the 0.5 probability that rater r is indeed first, we have an expected loss of $\frac{mb^2}{8}$. Noting that $\beta = \frac{b}{2}$ and $\sqrt{d} \leq \frac{b}{4\sqrt{2}}$, we have:

$$\begin{aligned} \frac{mb^2}{8} &= \frac{\log(\frac{\beta}{\sqrt{2d}} - 1)}{16} = \frac{\log(\frac{1}{\sqrt{d}}) + \log(\frac{b}{2\sqrt{2}} - \sqrt{d})}{16} \\ &\geq \frac{\log(\frac{1}{\sqrt{d}}) + \log \frac{b}{4\sqrt{2}}}{16} = \frac{\log \frac{1}{d} + \log(\frac{b^2}{32})}{32} \quad \square \end{aligned}$$

Remark: The value of m in this bound gives us the required setting for the number of items m_b in $\mathcal{P}(b)$, in terms of the parameter d .

3.2 Loss bound for n -sybil attacks

We can extend $\mathcal{P}(b)$ and $A(b)$ to prove a lower bound for n -sybil attacks; this extension is outlined in this section. The structure has n genuine raters $J = \{r_1, \dots, r_n\}$, each with an identical partition; each rates a disjoint set of items. The attacker also creates n sybils $J_A = \{r'_1, r'_2, \dots, r'_n\}$. The key is that the rating sequence X' is now constructed by an iterative random process: In each iteration, the attacker flips a coin, and if it is heads, he adds the next rater from r onto the list (as long as there are some remaining); otherwise, he adds the next rater from J_A . This construction ensures that, at least until the first n raters' ratings have been labeled, the probability that the next rater is an attacker is exactly 0.5, *even conditioned on knowing the types of the raters up to that point*. Thus, when deciding how much influence to give a rater, the sequence of ratings up to his first rating is irrelevant, and we can apply theorem 8 with our chosen value of d .

In principle, R could allow a different amount of expected damage on each rater, as long as the sum was no more than c . Given the $\Omega(\log(1/d))$ form of the bound, which is convex in d , dividing the expected damage equally minimizes the overall bound, and hence we assume without loss of generality that the expected damage on each of these first n items is be limited to $d = \frac{c}{n}$. Putting this value into the statement of theorem 8 gives us:

THEOREM 9. *For each $b \in (0, 0.5)$, there is an information structure $\mathcal{P}(b, n)$ with n informed raters and attack $A(b, n)$ such that any (n, c) -robust recommender algorithm R incurs an information loss of at least $n \frac{\log \frac{c}{n} + \log(\frac{b^2}{32})}{32}$.*

PROOF. Follows by substituting $d = \frac{c}{n}$ in theorem 8 \square

4. DISCUSSION AND FUTURE WORK

In our lower bound, we constructed a simple family of instances $\mathcal{P}(b)$, and showed that any recommender system must incur an information loss (expected increase in prediction error) on these instances. We thus proved that there must be information loss *in the worst case* instances, where an instance denotes a particular distribution of tastes, specific correlations with the target, and a specific rating order.

It is natural to ask if this information loss is a rare problem that occurs only in pathological instances, or if it is likely to occur in common situations as well. In this section, we outline several reasons to believe that, for any reasonable set of instances, the average-case loss is not likely to be much better than the worst-case lower bound: The bound is robust to more complex instances and different scoring rules; the attack model is fairly simple; and, the bounds would hold under alternative definitions of robustness. We also identify promising directions for future work in light of these results.

In our family of instances, the recommender prediction is always 0.5 before the rater arrives, and $0.5 \pm b$ after optimal use of the rater's information. However, the same form of bound can be derived for other settings. The key step in the lower bound proof is equation 1, through which the variance of the likelihood ratio $g(\mathbf{u})$ is bounded in terms of b^2 . We can consider different instances in which the starting point is $v \neq 0.5$; equation 1 can be shown to hold, with a different constant $\frac{1}{v(1-v)}$ in place of 4. As long as v is bounded away from 0 and 1, this will yield a similar lower bound on information loss. We can also consider information structures in which the base prediction r is not the same for all items, but follows some distribution; or, in which the change magnitude b is not the same for all items, but follows some distribution. Provided the distributions have suitably bounded support, a similar $\Omega(\log(n/c))$ loss bound will follow.

Another generalization is to look at alternative loss functions. In particular, the *log-loss* function is attractive because the informativeness would be quantified in terms of the standard information-theoretic entropy. In this model, too, the lower bound holds; indeed, it holds even more generally, without the restriction mentioned above that v be bounded away from 0 and 1.

Our attacks consist of simple random-guessing strategies, and our bounds thus hold for algorithms robust against any class of attacks that include these attacks. However, one critical feature of the attacks is that they mimic the rating distributions of a plausible genuine rater. One class of manipulation-resistance algorithms identify suspicious raters by matching their rating histories to a set of attack profiles [3, 12, 9, 7, 8]. Some empirical studies have examined the impact on predictions of removing suspicious raters. If there is no noticeable loss in prediction accuracy, our results implies that either the attack profiles are simplistic (i.e., easy to distinguish from genuine raters) or there is significant information redundancy among the genuine raters so that information discarded from some genuine raters can be compensated by information from others. A fruitful direction for future research is to incorporate such redundancy into our formal model.

Our definition of manipulation-resistance states that the expected *net damage* inflicted by the attacker on the target should not be too large. Other notions of robustness are also reasonable. One possibility would be to require that an informationless attacker cannot cause a large movement on any one item. Our lower bound extends to this model as well, because the y -guessing attacks we study are informationless, and we only account for the damage caused on a single item rated. Another alternative is to require that the net damage is never too large (not just in expectation); this is the notion of manipulation-resistance that the Influence Limiter satisfies. Our lower bound applies *a fortiori* to this definition as well. Another reasonable definition is

to require that the aggregate damage *across all legitimate targets* be limited in some way. O’Donovan and Smyth [10] suggest using accuracy information from multiple targets to judge credibility. Our bound of $\Omega(\log(n/c))$ loss applies to this model, but is very weak as an aggregate bound. This is an interesting direction for future research.

In [14], it is shown that the Influence Limiter is (n, c) -robust against a very broad class of strategies, and that the worst-case information loss it induces is $O(\log \frac{n}{c})$. Considered as a function of n alone, the $\Omega(\log n)$ lower bound matches the $O(\log n)$ upper bound. However, the constants in the two bounds differ significantly. The constants in the lower bound proof are probably not optimal, as our bounding technique required approximations that may not be tight. Nevertheless, it is an important challenge for future work to devise manipulation-resistant algorithms that move closer to the lower bound.

5. OTHER RELATED WORK

As mentioned, there are a number of papers on the topic of shilling attacks in recommender systems [11, 6, 3, 12, 7, 15, 8, 9, 14]. In this section, we discuss other literatures that are related to our approach.

We defined and analyzed an influence metric that measures the change that a rater can cause to the predicted probability of an item. Rashid *et al.* [13] propose other algorithm-independent measures of rater influence.

The literature on bounded-regret online learning deals with combining predictions from multiple forecasters and proving worst-case bounds on the error relative to the best predictor that could be chosen in hindsight (see Cesa-Bianchi and Lugosi [2] and references therein). Awerbuch and Kleinberg [1] study manipulation in a different model of the recommendation process: a user samples items and recommendations until he likes an item, at which point he recommends that item to others. They describe an online learning scheme and prove bounds on the number of samples required, even in the presence of adversaries.

There are parallels between our lower bound and the results of Friedman and Resnick [5] on the social costs of cheap pseudonyms. At a high level, both results demonstrate that the possibility of creating false identities forces newcomers to be trusted less, thus leading to a loss of system performance or efficiency. However, the result of Friedman and Resnick relies on a characterization of equilibria in an economic gain model, whereas our model is information-theoretic rather than economic, and does not use equilibrium arguments.

6. CONCLUSION

In this paper, we have presented an information-theoretic model of informativeness and manipulation in recommender systems, and used it to shed light on the tradeoff between using all available information and resisting manipulation. The first insight, based on our lower bound, is that some amount of information loss is unavoidable in a recommender system that resists manipulation. Further, we note that, keeping other parameters fixed, the lower bound on required information loss is unbounded as the number of sybils n is increased towards ∞ . This shows that no useful recommender system can be resistant to manipulation by an attacker with an *unbounded* number of sybils: to achieve this, the recommender would need to essentially throw away all

rating information from genuine raters for indefinitely long, thus rendering it useless.

Acknowledgment

This work was partially supported by the NSF under awards IIS-0812042, CCF-0728768, and IIS-0308006.

7. REFERENCES

- [1] B. Awerbuch and R. D. Kleinberg. Competitive collaborative learning. In *18th Annual Conference on Learning Theory (COLT 2005)*, volume 3559 of *LNAI*, pages 233–248. Springer, 2005.
- [2] N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.
- [3] P.-A. Chirita, W. Nejdl, and C. Zamfir. Preventing shilling attacks in online recommender systems. In *WIDM 05*, pages 67–74, 2005.
- [4] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, New York, 1991.
- [5] E. Friedman and P. Resnick. The social cost of cheap pseudonyms. *Journal of Economics and Management Strategy*, 10(2):173–199, 2001.
- [6] S. K. Lam and J. Riedl. Shilling recommender systems for fun and profit. In *Proceedings of WWW ’04.*, pages 393–402, 2004.
- [7] B. Mehta, T. Hoffman, and P. Fankhauser. Lies and propaganda: detecting spam users in collaborative filtering. In *Proceedings of IUI’07*, 2007.
- [8] B. Mehta and W. Nejdl. Attack resistant collaborative filtering. In *Proceedings of ACM SIGIR 2008 (to appear)*, 2008.
- [9] B. Mobasher, R. Burke, R. Bhaumik, and C. Williams. Towards trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Transactions on Internet Technology*, 7(2):1–40, 2007.
- [10] J. O’Donovan and B. Smyth. Trust no one: Evaluating trust-based filtering for recommenders. In *Proceedings of IJCAI’05*, 2005.
- [11] M. O’Mahony, N. Hurley, and G. Silvestre. Promoting recommendations: An attack on collaborative filtering. In *Proceedings of the 13th International Conference on Database and Expert System Applications*, pages 494–503. Springer-Verlag, 2002.
- [12] M. P. O’Mahony, N. J. Hurley, and G. C. M. Silvestre. Detecting noise in recommender system databases. In *Proceedings of the 2006 International Conference on Intelligent User Interfaces*, pages 109–115, 2006.
- [13] A. M. Rashid, G. Karypis, and J. Riedl. Influence in ratings-based recommender systems: An algorithm-independent approach. In *Proceedings of the SIAM International Conference on Data Mining*, 2005.
- [14] P. Resnick and R. Sami. The influence limiter: Provably manipulation-resistant recommender systems. In *Proceedings of the ACM Recommender Systems Conference (RecSys07)*, 2007.
- [15] J. Sandvig, B. Mobasher, and R. Burke. Robustness of collaborative recommendation based on association rule mining. In *Proceedings of the 2007 ACM Conference on Recommender Systems*, 2007.