Building Trust in Online Rating Systems through Signal Modeling

Presenter: Yan Sun

Yafei Yang, Yan Sun, Ren Jin, and Qing Yang

High Performance Computing Lab
University of Rhode Island
### Online Feedback-based Rating Systems

<table>
<thead>
<tr>
<th>Web Site</th>
<th>Category</th>
<th>Summary of reputation mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>eBay</td>
<td>Online auction house</td>
<td>Buyers and sellers rate one another following transactions</td>
</tr>
<tr>
<td>eLance</td>
<td>Professional Services marketplace</td>
<td>Contractors rate their satisfaction with subcontractors</td>
</tr>
<tr>
<td>Epinions</td>
<td>Online opinions forum</td>
<td>Users write reviews about products/services; other members rate the usefulness of reviews</td>
</tr>
<tr>
<td>Slashdot</td>
<td>Online Discussion board</td>
<td>Postings are prioritized or filtered according to the ratings they receive from readers</td>
</tr>
<tr>
<td>YouTube</td>
<td>Multimedia broadcasting</td>
<td>Viewers rate the video clips</td>
</tr>
<tr>
<td>Amazon</td>
<td>Online shopping site</td>
<td>Shoppers rate the products</td>
</tr>
</tbody>
</table>

- Users submit their opinions regarding to products, services, or other users;
- Submitted opinions are analyzed, aggregated and made publicly available.
An Important Problem: Unfair Ratings

- Unfair ratings -- a critical factor that undermine the reliability of online rating systems.

Individual unfair ratings
- an individual rater provides unfairly high or low ratings, resulting from raters’ personality/habit, careless, or randomness in rating behavior.

Collaborative unfair ratings
- a group of raters providing unfairly high or low ratings to boost or downgrade the overall rating of an object.
Existing Solutions

- Existing solutions
  - Clustering techniques
  - Statistically analysis
  - Endorsement-based quality estimation
  - Entropy-based detection

All based on Majority Rule
A **Challenging Problem: Unfair Ratings**

- No sufficient number of ratings

- Rating values are highly discrete;

- With smart, collaborative unfair raters, majority rule may not hold

Statistical methods, such as clustering, will not work.

Detecting rating is low unless tolerate a high false alarm rate;

Most existing schemes lost their foundation.
Our Novel Idea

- Rating values $\rightarrow$ samples of a random process
- Fair ratings $\rightarrow$ noise
- Unfair ratings $\rightarrow$ signal

**Basic Idea:** Model the overall rating values using an autoregressive (AR) signal modeling technique, and exam the model errors. When the ‘signal’ is presented, the model error is low.
Our Contributions

- An algorithm that detects suspicious ratings in the scenarios where existing techniques do not work;

- A system that utilizes trust models for rating aggregation and improves system reliability.
Classification of Unfair Ratings

- **Individual unfair ratings**
  - an individual rater provides unfairly high or low ratings, resulting from raters’ personality/habit, careless, or randomness in rating behavior.

- **Collaborative unfair ratings**
  - a group of raters providing unfairly high or low ratings to boost or downgrade the overall rating of an object.

  *Strategy 1*: large bias
  *Strategy 2*: moderate bias
Trust-based Reliable Rating Aggregation

Motivation

Introduction

System

Raw Rating

Feature Extraction I

Rating Filter

Feature Extraction II

Rating Aggregation

Aggregated Rating

Observation Buffer

Recom. Buffer

Trust Establishment

Direct Trust Establishment

Indirect Trust Establishment

Trust Record

Direct

Indirect

Rater 1

Rater 2

Malicious Rater Detection

Trust Manager

Record Maintenance

Initialization

Update according to time

Raters’ Option on other raters

normal Rating

Abnormal Rating

normal Rating

normal Rating
Algorithm 1: Detect Suspicious Interval

Motivation
Introduction
System
Algorithms
Algorithm - 1

**Procedure 1** Detecting suspicious interval and updating raters’ suspicious values

1. For each rater $i$, initialize $L_i^{latest} = 0$
2. for $k = 1 : W$ do
3. let $R$ denote the ratings for a certain object in the $k^{th}$ window.
4. find the all-pole model of the signal $R$ using the covariance method [7]. In particular, given the model order $p$, calculate the model coefficients $a = [1, a(1), ... a(p)]$ and normalized model error $e(k)$ ($0 < e(k) < 1$).
5. if $e(k) < threshold$ then
6. The $k^{th}$ window is marked as the suspicious
7. A suspicious level is calculated as $L(k) = scale \cdot (1 - e(k))/threshold$, where scale is scaling factor between 0 and 1.
8. for each rating in the $k^{th}$ window. do
9. assume this rating is from rater $j$
10. if $L_j^{latest} = 0$ then
11. $C_i = C_i + L(k); L_i^{latest} = L(k)$;
12. else
13. if $L_j^{latest} > L(k)$ then
14. $C_i = C_i + L(k) - L_j^{latest}, L_j^{latest} = L(k)$;
15. end if
16. end if
17. end for
18. end if
19. end for

---

AR signal modeling

Examining model error

Suspicious level depends on the model error
Evaluation of Algorithm 1

- Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>simu_time = 60</td>
<td>simulation time is 60 days</td>
</tr>
<tr>
<td>arrival_rate = 3</td>
<td>rating arrival is a Poisson process with arrival rate 3</td>
</tr>
<tr>
<td>R_level = 11</td>
<td>ratings which have 11 levels can be 0, 0.1, 0.2, \ldots, or 1</td>
</tr>
<tr>
<td>quality_start = 0.7</td>
<td>The quality of the object is 0.7 at the beginning and linearly increases to 0.8 at the end of the experiment.</td>
</tr>
<tr>
<td>quality_end = 0.8</td>
<td>The ratings from honest raters follow a Gaussian distribution with mean being the quality of the object and variance being 0.2.</td>
</tr>
<tr>
<td>good_Var = 0.2</td>
<td></td>
</tr>
</tbody>
</table>

Influenced

- \( A\_start = 30 \)
- \( A\_end = 44 \)
- \( \text{The unfair ratings arrive between day 30 and day 44} \)
- \( \text{biasshift}_1 = 0.2 \)
- \( \text{recruitpower}_1 = 0.3 \)
- \( \text{during the attack interval, 30\% raters increase their original ratings by 0.2} \)
- \( \text{biasshift}_2 = 0.15 \)
- \( \text{badVar} = 0.02 \)
- \( \text{recruitpower}_2 = 1 \)
- \( \text{type 2 collaborative ratings, whose arrival rate is 3, follow a Gaussian distribution with variance 0.02, and mean = object quality + 0.15} \)
Trust-based Reliable Rating Aggregation

**Raw Ratings**

- Ratings without collaborative raters
- Ratings with collaborative raters (red: type 1, green: type 2)
Majority Rule won’t work

Histogram of ratings (without collaborative rater)

Histogram of ratings (with collaborative rater)
Our Algorithm Worked!

Detection Ratio = 0.782;
False Alarm Ratio = 0.06.
Our algorithm worked for real-world data

Model errors for original data and data with collaborative ratings. (Dinosaur Planet, 2003.)
Trust-based Reliable Rating Aggregation

Motivation
Introduction
System
Algorithms
(1) Detection
(2) Trust in raters

Trust Manager

2. Calculating trust in raters

3. Find a good trust model for rating aggregation
Trust in Raters

- $n$: total number of ratings provided by this rater
- $n_f$: the number of ratings that are filtered out
- $n_s$: the number of ratings that are in suspicious interval
- $C_i$: the suspicious level, $i = 1, 2, \ldots, n_s$
- $b$: scaling factor between 0 and 1

$$F = n_f + b \sum_{i=1}^{n_s} C_i$$

$$S = n - n_f - n_s$$

$$TrustValue = \frac{(S + 1)}{(S + F + 2)}$$
Trust-based Reliable Rating Aggregation

Motivation
Introduction
System
Algorithms
(1) Detection
(2) Trust in raters
(3) Rating Aggregation

Rating Aggregation

Trust Relationship \{A: B, task\}

- \{rater: product, have a certain quality\} – Rating Value
- \{system: rater, provide fair ratings\} – Trust in Raters
- \{system: product, having a certain quality\} – aggregated ratings.
A Good Trust Model

- We have compared four popular trust models.
  - Simple averaging
  - Beta function based aggregation, without trust.
  - **Modified weighted average**
    \[
    R_{ag} = \frac{1}{\sum_{i:i \in R} \text{max}(T_i - 0.5, 0)} \sum_{i:i \in R} \text{max}(T_i - 0.5, 0) \cdot r_i
    \]
  - Beta-function based trust model
The rating scores have 10 levels
400 are reliable raters, 200 are careless raters and 200 are potential collaborative unfair raters. \( \text{good}_\text{var} = 0.2; \text{careless}_\text{var} = 0.3 \)

**collaborative rater**
- If recruited: *with a higher probability to rate*;
- If not recruited: behave as a reliable rater, but with lower probability to rate.

Rating 60 products during 360 days. In each month (30 days), the owner of 1 product recruit collaborative raters, who rate in 10 days.

The quality of the products is assumed to be uniformly distributed between 0.4 and 0.6.
System Performance Evaluation

- Mean of Rater’s Trust

![Graph showing Mean of Raters’ Trust over time (months)]
Trust-based Reliable Rating Aggregation

- Trust Values

Raters’ Trust in the 6th Month

For honest raters, false alarm rate is 1%
For careless raters, false alarm rate is 3%
For Collaborative raters, detection rate is 72%
Unfair rating detection ratio

No existing schemes are able to detect collaborative unfair raters that does not introduce a large bias and overpower honest raters in certain time intervals.
- Aggregated Rating

Rating Aggregation of Dishonest Products (Bias = 0.15)
Rating Challenge

- Real online rating data for 9 flat panel TVs.
- Participants control 50 biased raters.
- The participants’ goal is to boost the ratings of two products and reduce the ratings of another two products.
- The successfulness of the participants’ attack is determined by the overall manipulation power.
- The participants that can generate the largest MP value win the competition.

www.etanlab.com/rating