Spotting Fake Reviewer Groups in Consumer Reviews

Spotting Fake Reviewer Groups in Consumer Reviews

Arjun Mukherjee
Department of Computer Science
University of Illinois at Chicago
851 S. Morgan, Chicago, IL 60607
arjun4787@gmail.com

Bing Liu
Department of Computer Science
University of Illinois at Chicago
851 S. Morgan, Chicago, IL 60607
liub@cs.uic.edu

Natalie Glance
Google Inc.
4720 Forbes Ave, Lower Level
Pittsburgh, PA 15213
nglance@google.com

ABSTRACT

Opinionated social media such as product reviews are now widely used by individuals and organizations for their decision making. However, due to the reason of profit or fame, people try to game the system by opinion spamming (e.g., writing fake reviews) to promote or demote some target products. For reviews to reflect genuine user experiences and opinions, such spam reviews should be detected. Prior works on opinion spam focused on detecting fake reviews and individual fake reviewers. However, a fake reviewer group (a group of reviewers who work collaboratively to write fake reviews) is even more damaging as they can take total control of the sentiment on the target product due to its size. This paper studies spam detection in the collaborative setting, i.e., to discover fake reviewer groups. The proposed method first uses a frequent itemset mining method to find a set of candidate groups. It then uses several behavioral models derived from the collusion phenomenon among fake reviewers and relation models based on the relationships among groups, individual reviewers, and products they reviewed to detect fake reviewer groups. Additionally, we also built a labeled dataset of fake reviewer groups. Although labeling individual fake reviews and reviewers is very hard, to our surprise labeling fake reviewer groups is much easier. We also note that the proposed technique departs from the traditional supervised learning approach for spam detection because of the inherent nature of our problem which makes the classic supervised learning approach less effective. Experimental results show that the proposed method outperforms multiple strong baselines including the state-of-the-art supervised classification, regression, and learning to rank algorithms.

Categories and Subject Descriptors

H.1.2 [**Information Systems**]: Human Factors; J.4 [**Computer Applications**]: Social and Behavioral Sciences

Keywords

Opinion Spam, Group Opinion Spam, Fake Review Detection

1. INTRODUCTION

Nowadays, if one wants to buy a product, most probably, one will first read reviews of the product. If he/she finds that most reviews are positive, he/she is very likely to buy it. However, if most reviews are negative, he/she will almost certainly choose another product. Positive opinions can result in significant financial gains and fames for organizations and individuals. This, unfortunately, gives strong incentives for *opinion spamming*, which refers to human activities (e.g., writing fake reviews) that try to deliberately mislead readers by giving unfair reviews to some entities (e.g. products) in order to promote them or to damage their reputations. As more and more individuals and organizations are using reviews for their decision making, detecting such fake reviews becomes a pressing issue. The problem has been widely reported in the news[1].

There are prior works [14, 15, 23, 24, 31, 32, 34] on detecting fake reviews and individual fake reviewers or spammers. However, limited research has been done to detect fake reviewer (or spammer) groups, which we also call *spammer groups*. Group spamming refers to a group of reviewers writing fake reviews together to promote or to demote some target products. A spammer group can be highly damaging as it can take total control of the sentiment on a product because a group has many people to write fake reviews. Our experiments show that it is hard to detect spammer groups using review content features [31] or even indicators for detecting abnormal behaviors of individual spammers [24] because a group has more manpower to post reviews and thus, each member may no longer appear to behave abnormally. Note that by a group of reviewers, we mean a set of reviewer-ids. The actual reviewers behind the ids could be a single person with multiple ids (sockpuppet), multiple persons, or a combination of both. We do not distinguish them in this work.

Before proceeding further, let us see a spammer group found by our algorithm. Figures 1, 2, and 3 show the reviews of a group of three reviewers[2]. The following suspicious patterns can be noted about this group: (i) the group members all reviewed the same three products giving all 5 star ratings; (ii) they posted reviews within a small time window of 4 days (two of them posted in the same day); (iii) each of them only reviewed the three products (when our Amazon review data [14] was crawled); (iv) they were among the early reviewers for the products (to make a big impact). All these patterns occurring together strongly suggest suspicious activities. Notice also, none of the reviews themselves are similar to each other (i.e., not duplicates) or appear deceptive. If we only look at the three reviewers individually, they all appear genuine. In fact, 5 out of 9 reviews received 100% helpfulness votes by Amazon users indicating that the reviews are useful. Clearly, these three reviewers have taken total control of the sentiment on the set of reviewed products. In fact, there is a fourth reviewer in the group. Due to space limitations, we omit it here.

If a group of reviewers work together only once to promote or to demote a product, it is hard to detect them based on their collective behavior. They may be detected using the content of their reviews, e.g., copying each other. Then, the methods in [14, 23, 24, 31, 32, 34] are applicable. However, over the years, opinion spamming has become a business. People get paid to write fake reviews. Such people cannot just write a single review

WWW 2012, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1229-5/12/04.

[1] http://www.nytimes.com/2012/01/27/technology/for-2-a-star-a-retailer-gets-5-star-reviews.html
http://www.nytimes.com/2011/08/20/technology/finding-fake-reviews-online.html

[2] http://www.amazon.com/gp/pdp/profile/A3URRTIZEE8R7W
http://www.amazon.com/gp/pdp/profile/A254LYRIZUYXZG
http://www.amazon.com/gp/cdp/member-reviews/A1O70AIHNS4EIY

Arjun Mukherjee
Department of Computer Science
University of Illinois at Chicago

Bing Liu
Department of Computer Science
University of Illinois at Chicago

Natalie Glance
Google Inc.

WWW'12

Fake/Spam/Biased Reviews

- Online **reviews** play an important role in **Decision making** process
- Review Spamming, Motivations:
  - Fame
  - Financial gain
- To **promote** or **demote** other products
- Opinion spamming is now a business
  - people get paid to write fake reviews
  - So they **write many reviews about many products**, such collective behavior can give them away
- This study: Focusing on **Spammer Groups** instead of **individual** reviews/reviewers

Dataset

- Created a Labeled dataset for group opinion spam
- Refers to prior studies and argue that in absence of labeled data, best option is to create one based on human expert
- Dataset Stats:
    - Amazon Dataset from 2006 (updated on 2010)
    - Only manufactured products (53K reviewer with 110K reviews on 39K products)
    - Attributes: Title , content , star rating , posting date and helpful feedbacks

- 1) **Candidate groups**: Using Frequent Itemset Mining (FIM)
    - On list of reviewer ids per product
    - All groups with min_sup =3 and 2 items
        - Groups with at least 2 reviewers who worked at least on 3 products

Dataset

- 2) Opinion Spam signals:
  - Provided the list spam signals from prior research and websites:
    - (i) having zero caveats, (ii) full of empty adjectives, (iii) purely glowing praises with no downsides, (iv) being left within a short period of time of each other
  - Access to review Database

- Judges: employees from Rediff shopping (4) and eBay.in (4)
  - Spent 8 weeks to label 2431 groups.
  - **Spamicity Rate (SR)**
    - 1: spam , 0.5 borderline , 0: no-spam
      - 8 votes, average of all would be the SR for review.
      - average of reviews SR => group spamicity

SPAMMING BEHAVIOR INDICATORS

For modeling or learning, a set of effective spam indicators or features is needed

Spamming behavior indicators

1. Group spam behavior indicators
   - Group time window (GTW)

$$GTW(g) = \max_{p \in P_g}(GTW_P(g,p)),$$

$$GTW_P(g,p) = \begin{cases} 0 & \text{if } L(g,p) - F(g,p) > \tau \\ 1 - \dfrac{L(g,p) - F(g,p)}{\tau} & otherwise \end{cases},$$

   - Group Deviation (GD)

$$GD(g) = \max_{p \in P_g}(D(g,p)),$$

$$D(g,p) = \frac{|r_{p,g} - \bar{r}_{p,g}|}{4},$$

Spamming behavior indicators

- Group spam behavior indicators
  - Group Content Similarity (GCS)

$$GCS(g) = \max_{p \in P_g}(CS_G(g, p)),$$

$$CS_G(g, p) = \underset{m_i, m_j \in g, i<j}{avg}\left(cosine(c(m_i, p), c(m_j, p))\right),$$

  - Group Member Content Similarity (GMCS)

$$GMCS(g) = \frac{\sum_{m \in g} CS_M(g, m)}{|g|},$$

$$CS_M(g, m) = \underset{p_i, p_j \in P_g, i<j}{avg}\left(cosine(c(m, p_i), c(m, p_j))\right)$$

Spamming behavior indicators

- Group spam behavior indicators
  - Group Early Time Frame (GETF)

$$GETF(g) = \max_{p \in P_g}(GTF(g, p)),$$

$$GTF(g,p) = \begin{cases} 0 & \text{if } L(g,p) - A(p) > \beta \\ 1 - \dfrac{L(g,p) - A(p)}{\beta} & \text{otherwise} \end{cases},$$

  - Group Size Ratio (GSR)

$$GSR(g) = \underset{p \in P_g}{avg}(GSR_p(g, p)),$$

$$GSR_p(g,p) = \frac{|g|}{|M_p|},$$

Spamming behavior indicators

- Group spam behavior indicators
  - Group Size (GS)

$$GS(g) = \frac{|g|}{\max(|g_i|)}$$

  - Group Support Count (GSUP)

$$GSUP(g) = \frac{|P_g|}{\max(|P_{g_i}|)}$$

Spamming behavior indicators

2. Individual Spam Behavior Indicators

- Individual Rating Deviation (IRD):

$$IRD(m, p) = \frac{| r_{p,m} - \bar{r}_{p,m} |}{4},$$

- Individual Content Similarity (ICS)

$$ICS(m, p) = avg(cosine(c(m, p))$$

Spamming behavior indicators

2. Individual Spam Behavior Indicators

- Individual Early Time Frame (IETF)

$$IETF(m,p) = \begin{cases} 0 & \text{if } L(m,p) - A(p) > \beta \\ 1 - \dfrac{L(m,p) - A(p)}{\beta} & \text{otherwise} \end{cases},$$

- Individual Member Coupling in a group (IMC)

$$IMC(g,m) = \operatorname*{avg}_{p \in P_g}\left(\frac{|(T(m,p) - F(g,p)) - avg(g,m)|}{L(g,p) - F(g,p)}\right),$$

$$avg(g,m) = \frac{\sum_{m_i \in G - \{m\}} (T(m_i,p) - F(g,p))}{|g| - 1},$$

This behavior measures how closely a member works with the other members of the group. If a member m almost posts at the same date as other group members, then m is said to be tightly coupled with the group

Empirical Analysis

Statistical validation

- Spamicity threshold : 0.5 => 62% non-spam and 38% spam groups
- Feature effectiveness:
  - $$Eff(f) \equiv P(f > 0 \mid Spam) - P(f > 0 \mid Non-spam),$$

    $$P(f > 0 \mid Spam) = \frac{|\{g \mid f(g) > 0 \wedge g \in Spam\}|}{|\{g \mid g \in Spam\}|}$$

    $$P(f > 0 \mid Non-spam) = \frac{|\{g \mid f(g) > 0 \wedge g \in Non-spam\}|}{|\{g \mid g \in Non-spam\}|}$$
  - Using Fisher's exact test, it is reported that spam groups are more likely to exhibit feature.
    - null hypothesis rejected with $p < 0.0001$

Behavioral Distribution

- Position
  - for a given cumulative percentage cp, the corresponding feature value xn for non-spam groups is less than xs for spam groups
- Steep initial jumps
  - very few groups obtain significant feature values
- Gaps
  - The separation margin refers to the relative discriminative potency

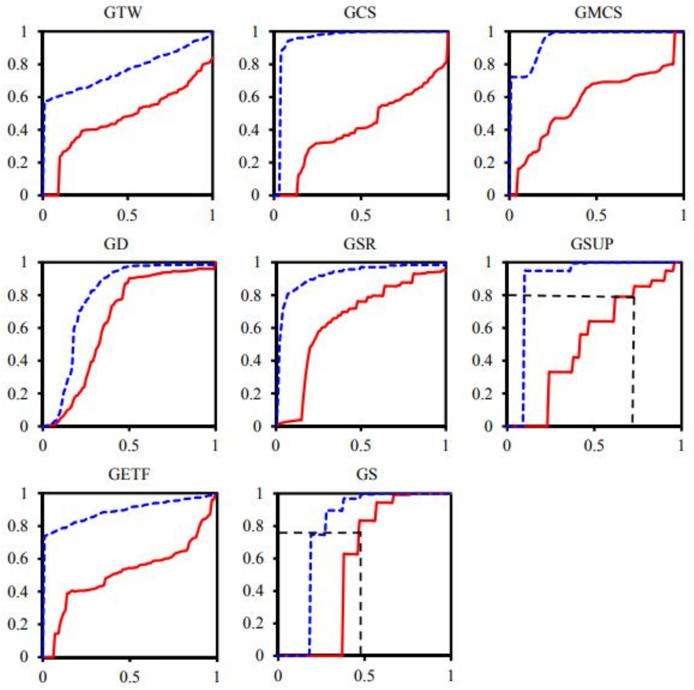


Figure 4: Behavioral Distribution. Cumulative % of spam (solid) and non-spam (dashed) groups vs. feature value

MODELING RELATIONS

MODELING RELATIONS

Better not to follow the classic approach : Classification

1. training and testing instances are not independently and identically drawn from some distribution (groups share members)
2. Group features only summarize the group behaviors (avg/sum)
   a. lead to loss of information
3. It is difficult to include the effect of Products!

So, they propose a more effective model to address the above concerns and also cover three binary relations:

Group Spam–Products
Member Spam–Products,
and Group Spam–Member Spam.

Group Spam–Products Model

- The relation among groups and products they target.
  - (i) spam contribution to p by each group reviewing p and
  - (ii) “spamicity” of each such group

$$w_1(p_i, g_j) = \frac{1}{5}[GTW_P(g_j, p_i) + D(g_j, p_i) + GTF(g_j, p_i) + CS_G(g_j, p_i) + GSR_P(g_j, p_i)],$$

$$W_{PG} = [w_1(p_i, g_j)]_{|P|\times|G|} \quad (16)$$

W_{PG} denotes the corresponding contribution matrix.

$$s(p_i) = \sum_{j=1}^{|G|} w_1(p_i, g_j) s(g_j); \quad V_P = W_{PG} V_G,$$

$$s(g_j) = \sum_{i=1}^{|P|} w_1(p_i, g_j) s(p_i); \quad V_G = W_{PG}^T V_P$$

Member Spam–Product Model

- IRD (individual rating deviation of m towards p)
- ICS (individual content similarity of reviews on p by m)
- IETF (individual early time frame of spam infliction by m towards p)

$$w_2(m_k, p_i) = \frac{1}{3}[IRD(m_k, p_i) + ICS(m_k, p_i) + IETF(m_k, p_i)],$$

$$W_{MP} = [w_2(m_k, p_i)]_{|M|\times|P|} \qquad (19)$$

We sum the individual contribution of each member w2, weighted by its spamicity:

$$s(m_k) = \sum_{i=1}^{|P|} w_2(m_k, p_i) s(p_i); \quad V_M = W_{MP} V_P$$

$$s(p_i) = \sum_{k=1}^{|M|} w_2(m_k, p_i) s(m_k); \quad V_P = W_{MP}^T V_M$$

Group Spam–Member Spam Model

- IMC (degree of m's coupling in g),
- GS (size of g with which m worked), and
- GSUP (number of products towards which m worked with g)

$$w_3(g_j, m_k) = \frac{1}{3}[IMC(g_j, m_k) + \boxed{(1 - GS(g_j))} + GSUP(g_j)],$$

$$W_{GM} = [w_3(g_j, m_k)]_{|G| \times |M|}$$

for large groups, the individual contribution of a member diminishes. Hence we use 1-GS(gj) to compute w3.

$$s(g_j) = \sum_{k=1}^{|M|} w_3(g_j, m_k) s(m_k); \quad V_G = W_{GM} V_M,$$

$$s(m_k) = \sum_{j=1}^{|G|} w_3(g_j, m_k) s(g_j); \quad V_M = W_{GM}^T V_G.$$

GSRank: Ranking Group Spam

Algorithm: GSRank

Input: Weight matrices W_{PG}, W_{MP}, and W_{GM}
Output: Ranked list of candidate spam groups

1. Initialize $V_G^0 \leftarrow [0.5]_{|G|}$; $t \leftarrow 1$;
2. Iterate:
   i. $V_P \leftarrow W_{PG} V_G^{(t-1)}$; $V_M \leftarrow W_{MP} V_P$;
   ii. $V_G \leftarrow W_{GM} V_M$; $V_M \leftarrow W_{GM}^T V_G$;
   iii. $V_P \leftarrow W_{MP}^T V_M$; $V_G^{(t)} \leftarrow W_{PG}^T V_P$;
   iv. $V_G^{(t)} \leftarrow V_G^{(t)} / \| V_G^{(t)} \|_1$;

   until $\| V_G^{(t)} - V_G^{(t-1)} \|_\infty < \delta$
3. Output the ranked list of groups in descending order of V_G^*

Complexity: linear in the number of candidate groups discovered by FIM

$$O(t(|G|(|M|+|P|) + |M||P|))$$

EXPERIMENTAL EVALUATION

- We first split 2431 groups:
  - The development set, D with 431 groups (randomly sampled) for parameter estimation
    - for GTW and GETF ,
      - using a greedy hill climbing search to maximize the log likelihood of the set D
      - $\tau = 2.87$

        the time interval beyond which members in a group are not likely to be working in collusion
      - $\beta = 8.86$

        denotes the time interval beyond which reviews posted are not considered to be “early” anymore
  - The validation set, V with 2000 groups for evaluation.
- All evaluation metrics averaged over 10-fold cross validation (CV)

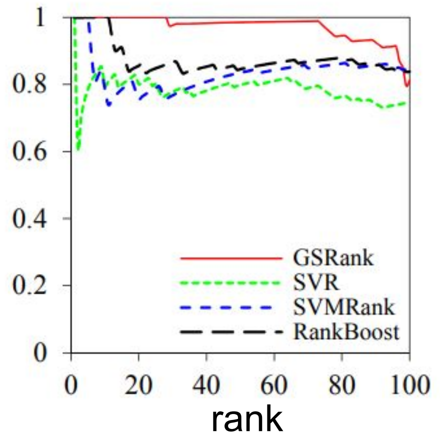
Ranking Experiments :: baselines

1. Using **regression**
    - The problem of ranking spammer groups can be seen as:
        - optimizing the spamicity of each group as a regression target
    - the support vector regression (SVR) system in SVMlight is used

2. and **Learning to Rank**
    - we treat each feature f as a ranking function
    - The rank produced by each feature is based on a certain spamicity dimension
    - None of the ranks may be optimal. A learning to rank method basically learns an optimal ranking function using the combination of f1…f8
    - Each group is vectorized with (represented with a vector of) the 8 group spam features

Ranking Experiments, cont.

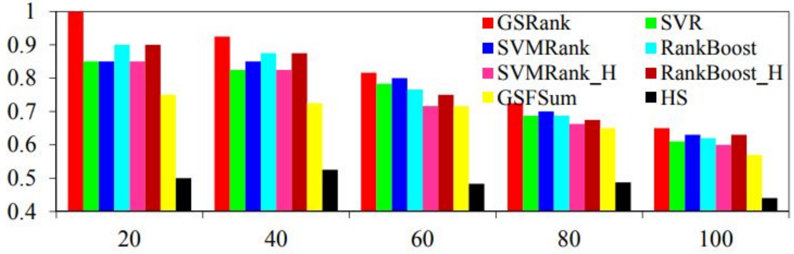
- Normalized Discounted Cumulative Gain (NDCG) as our evaluation metric
- GSRank performs the best at all top rank positions except at the bottom,
  - which are unimportant because they are most likely to be non-spam (since in each fold of cross validation, the test set has only 200 groups and out of the 200 there are at most 38% spam groups)



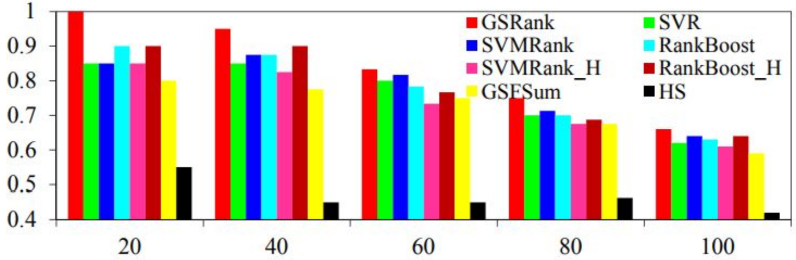
Ranking Experiments, cont.

- we also experimented with the following baselines:
  - Group Spam Feature Sum (GSFSum)
    - to rank the groups in descending order of the sum of all feature values
  - Helpfulness Score (HS)
    - HS uses the mean helpfulness score (percentage of people who found a review helpful) of reviews of each group to rank groups in ascending order of the scores
  - Heuristic training rankings (H)
    - three heuristic rankings using feature mixtures

$$h_1(g) : G \rightarrow \mathbf{R}^+,\ h_1(g) = GCS(g) + GMCS(g)$$
$$h_2(g) : G \rightarrow \mathbf{R}^+,\ h_2(g) = GS(g) + GSUP(g) + GTW(g)$$
$$h_3(g) : G \rightarrow \mathbf{R}^+,\ h_3(g) = GSR(g) + GETF(g) + GD(g)$$



(a) The spamicity threshold of $\xi = 0.5$



(b) The spamicity threshold of $\xi = 0.7$

Figure 6: Precision @ n = 20, 40, 60, 80, 100 rank positions.

All the improvements of GSRank over other methods are statistically significant at the confidence level of 95% based on paired *t*-test.

	$\xi = 0.5$	$\xi = 0.7$
Spam	38%	29%
Non-spam	62%	71%

Classification

- If a spamicity threshold is applied to decide spam and non-spam groups, supervised classification can also be applied
- features that we consider in learning:
    - Group Spam Features (GSF)
    - Individual Spammer Features (ISF)
    - Linguistic Features of reviews (LF)

      (word and POS (part-of-speech) n-gram features)

- AUC (Area Under the ROC Curve) is employed for classification evaluation

Classification

Feature Settings	SVM	LR	SVR	SVM Rank	Rank Boost	SVM Rank_H	Rank Boost_H	GS Rank
GSF	0.81	0.77	0.83	0.83	0.85	0.81	0.83	**0.93**
ISF	0.67	0.67	0.71	0.70	0.74	0.68	0.72	
LF	0.65	0.62	0.63	0.67	0.72	0.64	0.71	
GSF + ISF + LF	0.84	0.81	0.85	0.84	**0.86**	0.83	0.85	

(a) The spamicity threshold of $\xi = 0.5$

Feature Settings	SVM	LR	SVR	SVM Rank	Rank Boost	SVM Rank_H	Rank Boost_H	GS Rank
GSF	0.83	0.79	0.84	0.85	0.87	0.83	0.85	**0.95**
ISF	0.68	0.68	0.73	0.71	0.75	0.70	0.74	
LF	0.66	0.62	0.67	0.69	0.74	0.68	0.73	
GSF + ISF + LF	0.86	0.83	0.86	0.86	**0.88**	0.84	0.86	

(b) The spamicity threshold of $\xi = 0.7$

Table 2: AUC results of different algorithms and feature sets.

The End