A Unified Model for Unsupervised Opinion Spamming Detection Incorporating Text Generality*

Yinqing Xu Bei Shi Wentao Tian and **Wai Lam**
Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong, Hong Kong
{yqxu, bshi, wttian, wlam}@se.cuhk.edu.hk

Abstract

Many existing methods on review spam detection considering text content merely utilize simple text features such as content similarity. We explore a novel idea of exploiting text generality for improving spam detection. Besides, apart from the task of review spam detection, although there have also been some works on identifying the review spammers (users) and the manipulated offerings (items), no previous works have attempted to solve these three tasks in a unified model. We have proposed a unified probabilistic graphical model to detect the suspicious review spams, the review spammers and the manipulated offerings in an unsupervised manner. Experimental results on three review corpora including Amazon, Yelp and TripAdvisor have demonstrated the superiority of our proposed model compared with the state-of-the-art models.

1 Introduction

In many modern E-commerce websites such as Amazon, Yelp and TripAdvisor, online review texts and ratings have increasingly played an important role to help make purchase decisions for potential customers. Products with large proportion of positive reviews tend to attract more customers and bring significant financial gains, while large amount of negative review comments can also defame such product and cause sales loss. Due to the reputation and financial incentive, imposters may be hired to deliberately write fake or deceptive reviews to promote or demote the reputation for their target products or services. Such imposters are called review or opinion spammers and their review texts are called review spams [Jindal and Liu, 2008]. Definitely, this malicious activity would mislead the potential customers and damage the fairness of the market.

In the past few years, several supervised methods for detecting review spams or review spammers have been proposed. Unlike other forms of spamming, it is difficult to collect a large amount of gold-standard labels for reviews by means of manual effort. Thus, most of these methods [Mukherjee *et al.*, 2013; Li *et al.*, 2013a; Sun *et al.*, 2013] just rely on the ad-hoc or pseudo fake or non-fake labels for model training, such as the labels annotated by the Amazon anonymous online workers [Ott *et al.*, 2011; Li *et al.*, 2014]. On the other hand, some unsupervised methods have been proposed to detect the individual review spammer [Mukherjee *et al.*, 2013; Lim *et al.*, 2010; Wang *et al.*, 2011] and review spammer groups [Mukherjee *et al.*, 2012]. In addition, time series pattern [Xie *et al.*, 2012], rating distribution [Feng *et al.*, 2012], reviewer graph [Wang *et al.*, 2011], and reviewing burstiness [Fei *et al.*, 2013] have also been applied to identify the review spams in an unsupervised manner.

The features used in the previous works usually contain reviewer related features, product item related features and the review text content features. Though review texts can provide us rich features at both linguistic and semantic level, most of the previous works merely utilize simple text features such as content similarity, n-gram or the review length [Li *et al.*, 2011]. In this paper, we explore a novel idea of exploiting text generality in addition to the existing features for improving spam detection.

Intuitively, it is more likely that spammed review content tends to be general, and overly general review description has higher chance to be review spam[1]. This phenomenon frequently occurs in the domains such as hotels and restaurants. Review spammers for such domains are hard to write the comments in details when lack of truthful experience. On the contrary, for the domains like daily products or mobile phones, review spammers can learn the detailed related information from the advertisements or the experience of corresponding substitutes, so that they can easily fabricate a detailed and nearly genuine review text. In this paper, we only concentrate on the former domains.

To illustrate the idea of text generality for review spams, we provide two example reviews below,

1. The Chicago Hilton is a **great hotel** our stay there was **fantastic**. The hotel is placed in the heart of the city were you can find

*The work described in this paper is substantially supported by grants from the Research Grant Council of the Hong Kong Special Administrative Region, China(Project Codes: 413510 and 14203414) and the Direct Grant of the Faculty of Engineering, CUHK (Project Code: 4055034). This work is also affiliated with the CUHK MoE-Microsoft Key Laboratory of Human-centric Computing and Interface Technologies.

[1] http://www.moneytalksnews.com/3-tips-for-spotting-fake-product-reviews-%E2%80%93-from-someone-who-wrote-them/

it a **easy walk** to almost **everything** from the local colleges to the **great restaurant** that surround the city. The room itself was **clean** and the staff was **fun and helpful**. I cant really say anything bad about this place it was a **great time** and a **great place** to stay while we were there.

2. We stay at Hilton for **4 nights** last march. We got a large room with **2 double beds and 2 bathrooms,** The TV was Ok, **a 27' CRT Flat Screen**. The coincierge was very friendly when we need. The room was very cleaned when we arrived, **we ordered some pizzas from room service and the pizza was Ok also.** The breakfast is charged, **20 dollars,** kinda expensive. The internet access (WiFi) is charged, **13 dollars/day.** Tip: **When leaving the building, always use the Michigan Av exit. Its a great view.**

These two reviews are extracted from the gold-standard review spam dataset in [Ott *et al.*, 2011] where they hired Amazon workers to crowdsource fake reviews for the hotels in Tripadvisor. The review 1 is a fake review where the bolded general terms or phrases frequently appear, while the review 2 is a truthful review which describes each aspect in detail and even provides us the breakfast price as well as TV screen size. It is more likely that the reviewer of the review 2 has the genuine experience. Consequently, we propose a method that computes text generality and attempt to improve spam detection in addition to other useful features for the reviews in the domains such as hotels and restaurants.

Recently, [Li *et al.*, 2013b] focused on identifying the product or service offerings where fake reviews appear with high probability. They named this task as identifying **manipulated offerings**. Although several methods have been proposed to individually detect the suspicious review spams, the review spammers, and the manipulated offerings, no previous works have attempted to solve these three tasks in a unified model. Such unified approach is important and useful for the review portal operators, such as TripAdvisor and Yelp, to investigate and monitor the activities of highly suspicious reviews, reviewers (users) and offerings (items) in order to maintain a fair market. Although [Lim *et al.*, 2010] have reported that these three tasks are intimately related as solving one task can help solve another two, it is not appropriate to consider all suspicious review spammer's reviews as fake reviews. Some spammers are able to write fake reviews by his own truthful account or the truthful accounts of his friends, or some clever spammers may even hijack truthful accounts and begin providing malicious and misleading reviews [Beutel *et al.*, 2014]. Besides, to detect the review spammers or the manipulated offerings by the proportion of his or its fake reviews is also not appropriate since the spamming degree (called "spamicity" in this paper) of a review is dyadic involving the mutual interactions between the reviewer (user) and the offering (item). The spamicity of a review should be simultaneously influenced by the spamicity of the reviewer (user) and the offering (item). For example, faced with different product items, the same spammer would inevitably write the fake reviews with different spamicity since they have different background knowledge for these items. Similarly, the same item would receive reviews from different reviewers with various spamicity. Consequently, it would be more effective to estimate the spamicity of the review, the reviewer (user) and the offering (item) in a unified model rather than individually estimate the spamicity of the reviewer or the offering by the spamicity of corresponding reviews.

In this paper, we have proposed a unified probabilistic graphical model to detect the suspicious review spams, the review spammers, and the manipulated offerings in an unsupervised manner. Our idea of tackling these three tasks simultaneously is new. We formulate this unified task as a ranking problem. Our model takes the collection of review texts in a particular domain as input. The goal is to rank the review texts, the reviewers (users) and the offerings (items) based on their spamicity, which is modeled as a latent variable with other observed features. Besides, compared with previous unsupervised opinion spam detection models, our model is able to effectively exploit the text generality based on the topic hierarchy generated by the hierarchical Latent Dirichlet Allocation (hLDA) model [Blei *et al.*, 2010].

In order to evaluate our model, we have performed several experiments on the review corpora including Amazon, Yelp, and TripAdvisor. We borrow the indirect review classification method from [Mukherjee *et al.*, 2013] for evaluating the review spam ranking results without using any labeled data. Meanwhile, three human judges have been hired to evaluate the ranking lists for the reviewers and the offerings. The experimental results demonstrate that our model significantly outperforms the state-of-the-art unsupervised baseline methods.

2 Our Proposed Model

We investigate the task of jointly detecting the suspicious review spams, the review spammers, and the manipulated offerings in an unsupervised manner. To facilitate a better presentation, we call the review spammer as "suspicious user" while manipulated offering as "suspicious item". Given a review corpus $\mathcal{D} = \{d_1, d_2, ..., d_{|\mathcal{D}|}\}$, the goal is to simultaneously rank the review texts, users, and items based on the spamicity (degree of spamming in the range of $[0, 1]$) [Mukherjee *et al.*, 2013].

2.1 Model Description

We propose a unified probabilistic graphical model, called "Unified Review Spamming Model" (URSM), for detecting the suspicious review spams, the review spammers, and the manipulated offerings in an unsupervised manner. In order to infer the latent spamicity, URSM not only considers the observed abnormal features of reviews, users, and items, but also incorporates the text generality detected from the review text.

Differing with most of the previous works, URSM decomposes the spamicity into K levels. Each review, user, and item is thus modeled as a K-dimensional spam level distribution vector where the sum of each element equals one. For a particular user u, its spam level distribution vector g_u is assumed to be drawn from a Dirichlet distribution, and is responsible for generating the corresponding observed user abnormal features F^{ux} and user-related review abnormal features F^{uy}. As shown in [Mukherjee *et al.*, 2013], these abnormal features are the useful indicator for spamming. For example, a user's

reviews are all nearly duplicated, or he always gives extreme ratings. Sec. 2.2 will explain such features in detail. Specifically, for the review d, URSM will sample a user spam level assignment π_d^u from the Multinomial distribution of g_u which is the conjugate distribution with Dirichlet distribution. Given the parameters ϕ_{kf}^{ux} and ϕ_{kf}^{uy}, each spam level k is characterized by a Bernoulli distribution for each binary user feature f, i.e. $F_f^{ux} \sim Bern(\phi_{kf}^{ux})$, and a Beta distribution for each user-related review feature f in $[0, 1]$, i.e. $F_f^{uy} \sim Beta(\phi_{kf}^{uy})$. Consequently, these Bernoulli and Beta distributions are employed to generate the observed user features F^{ux} and F^{uy}. On the other hand, we perform the similar process to generate the observed item features F^{vx} and F^{vy}.

For the review d, we believe its corresponding spam level distribution vector s_d is influenced by three factors including the user spam level distribution vector g_u, the item spam level distribution vector h_v, and the text generality vector m_d, due to the fact that the review text is dyadic [Xu *et al.*, 2014] involving mutual interaction between users and items. On top of that, text generality can provide us an additional clue for review spams. In URSM, such generality is derived from the hierarchical Latent Dirichlet Allocation (hLDA) model which is a classical Bayesian nonparametric topic model [Blei *et al.*, 2010]. Provided a document collection, hLDA is able to extract a topic hierarchy tree with a fixed depth L where each node corresponds to a topic and each document will have a topic path c_d from the root topic to the leaf topic. Their topic levels correspond to 0 (root) to $L-1$ (leaf). Since the topic whose level close to the root (leaf) level indicates a relatively general (specific) topic, more words in the document assigned to a topic with higher level would indicate a "general" document. Thus, we can measure the text generality by the topic proportion θ_d for the review d. However, in hLDA, the topic proportion θ_d is generated by the stick breaking process based on the infinite topic space while our review spam level distribution vector s_d is in finite space. In order to perform efficient inference, we instead use the topic assignment z_d, i.e. the sample of topic proportion, for text generality measurement.

In URSM, we assume the depth of the topic tree L equals the number of spam level K. Given the counter n_{dl} referring to the number of words assigned to the level l in the review d, we can calculate the kth element of text generality m_d by the reverted normalized counter n_{dl} since the spam level 0 (K) indicates the least (most) suspicious level. For example, a large n_{d0} would indicate that most of the words in the review d are general terms leading to a large spamicity in the most suspicious level K.

$$m_{dk} = \frac{n_{d(K-k-1)}}{n_{d(\cdot)}}, k = 0, 1, ..., K-1 \tag{1}$$

where $n_{d(\cdot)}$ represents the number of words in the review d.

Consequently, s_d is assumed to be drawn from the Normal distribution with the mean as weighted sum of g_u, h_v and m_d.

$$s_{dk} \propto N(\omega^T \widetilde{s}_{dk}, \sigma^2) \tag{2}$$

where $\widetilde{s}_{dk} = [g_{uk}, h_{vk}, m_{dk}]$ and ω represents the weight.

Figure 1 illustrates the entire graphical model of URSM. We note that each user (item) feature is placed within the review plate since each user (item) feature is exhibited by all his/her (its) reviews, as similarly used in [Mukherjee *et al.*, 2013]. The generative process for the observed features and observed words is described below.

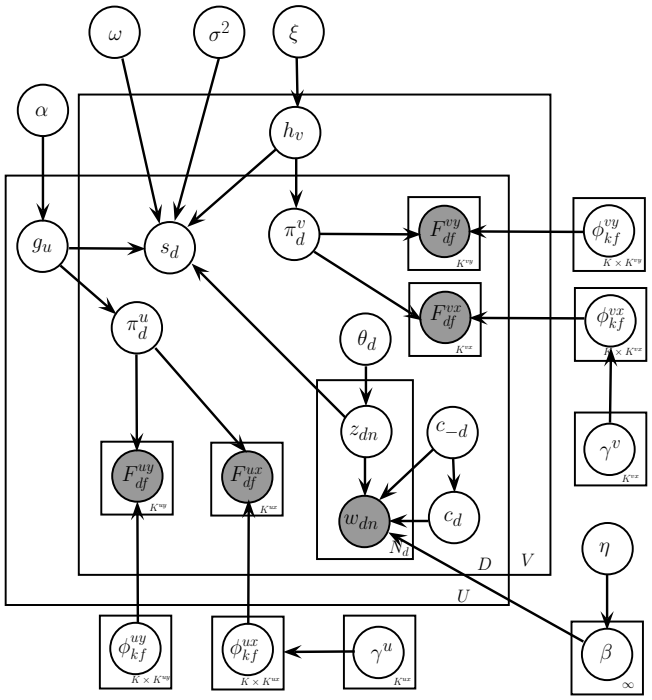

Figure 1: Our Proposed Model - URSM

- For each user $u \in \mathcal{U}$, choose $g_u \propto \text{Dir}(\alpha)$
- For each item $v \in \mathcal{V}$, choose $h_v \propto \text{Dir}(\xi)$
- For each user-related review abnormal feature f under the k spam level, choose $\phi_{kf}^{ux} \propto \text{Beta}(\gamma_k^u)$
- For each item-related review abnormal feature f under the k spam level, choose $\phi_{kf}^{vx} \propto \text{Beta}(\gamma_k^v)$
- For topic collection β, choose a new topic $\beta_i \propto \text{Dir}(\eta)$ if necessary
- For each review d written by the user u for the item v
  - draw $\pi_d^u \propto \text{Multi}(g_u)$, $\pi_d^v \propto \text{Multi}(h_v)$
  - draw $F_{df}^{ux} \propto \text{Bern}(\phi_{kf}^{ux})$, $f \in [0, K^{ux}-1]$
  - draw $F_{df}^{vx} \propto \text{Bern}(\phi_{kf}^{vx})$, $f \in [0, K^{vx}-1]$
  - draw $F_{df}^{uy} \propto \text{Beta}(\phi_{kf}^{uy})$, $f \in [0, K^{uy}-1]$
  - draw $F_{df}^{vy} \propto \text{Beta}(\phi_{kf}^{vy})$, $f \in [0, K^{vy}-1]$
  - for each word n in the review d
    - generate w_{dn} by the hLDA
  - draw $s_{dk} \propto N(\omega^T \widetilde{s}_{dk}, \sigma^2)$, $\widetilde{s}_{dk} = [g_{uk}, h_{vk}, m_{dk}]$ $k \in [0, K-1]$

The generative process for the words is similar to the original hLDA except an additional dependency on s_d. Due to space limit, we will not describe the word generative process in detail. The dependency relationship can be easily seen in the inference part.

2.2 Observed User and Item Abnormal Features

We present several observed abnormal features about reviews, users and items which are useful for spam detection. In [Mukherjee *et al.*, 2013], they proposed both user abnormal features such as content similarity, and user-related review abnormal features such as duplicate/near duplicate reviews. Since such features have been explained in [Mukherjee *et al.*, 2013], we just introduce the corresponding item abnormal features and item-related review abnormal features.

Item Abnormal Features: For the item v, such feature value F_f^{vy} is assumed in the range of $[0, 1]$ modeled by the Beta distribution. The value close to 1 (0) indicates spamming (non-spamming).

The first three abnormal features for items, i.e. Content Similarity $F_{f_0}^{vy}$, Maximum Number of Reviews Within a Time Window $F_{f_1}^{vy}$, and Reviewing Burstiness $F_{f_2}^{vy}$, are similar with the user abnormal features presented in [Mukherjee *et al.*, 2013]. The unique feature for the item v is the Ratio of Singleton Reviews $F_{f_3}^{vy}$. Singleton review is written by the reviewer who only has one piece of review. The study in [Xie *et al.*, 2012] reported that the singleton reviews tend to be involved with spamming since professional review spammers would usually register many new accounts and post a single review comment. We thus employ the ratio of singleton reviews for a item to be one of the item abnormal features.

$$F_{f_3}^{vy} = \frac{\#[\text{singleton reviews}]}{\#[\mathcal{D}_v]} \tag{3}$$

where # represents the counter.

Item-related Review Abnormal Features: Apart from the user-related review abnormal features F_f^{ux} in [Mukherjee *et al.*, 2013], we also propose two binary item-related review abnormal features F_f^{vx}. Value of 1 (0) refers to the spamming (non-spamming).

1. Extreme Rating under the J-shaped Rating Distribution: According to the study in [Feng *et al.*, 2012], manipulated items that hire spammers to write spam reviews would necessarily distort the natural rating score distribution of this particular item. Normally, the items involving spamming tend to have the bimodel (J-shaped) rating score distribution. The number of extreme rating (1 or 5 stars) is more than the number of 2 to 4 stars. Consequently, we believe that such feature value $F_{f_0}^{vx}$ for a review is 1 if exhibiting extreme rating score as well as commenting for a item with J-shaped rating distribution, and 0 otherwise.

2. Reviewing in the Burstiness Time Interval: Reviews posting within the burstiness time window of its corresponding item would be considered as review spams with high probability. Thus, we first detect the burstiness time interval by sliding a time window of fixed width τ_w^v. If the number of reviews exceed the given threshold n_r, we will assign such feature value $F_{f_1}^{vx}$ of the reviews in this interval as 1, and 0 otherwise.

2.3 Posterior Inference

Given the model parameters $\mathbf{\Omega} = \{\boldsymbol{g}, \boldsymbol{h}, \boldsymbol{s}, \boldsymbol{\omega}, \boldsymbol{\phi^{ux}}, \boldsymbol{\phi^{vx}}, \boldsymbol{\phi^{uy}}, \boldsymbol{\phi^{vy}}\}$ and the observed data $\mathbf{\Theta} = \{\boldsymbol{F^{ux}}, \boldsymbol{F^{vx}}, \boldsymbol{F^{uy}}, \boldsymbol{F^{vy}}\}$, the key step of applying the URSM model is to infer the posterior distribution of the latent variables $\mathbf{\Phi} = \{\boldsymbol{\pi^u}, \boldsymbol{\pi^v}, \boldsymbol{z}, \boldsymbol{c}, \boldsymbol{\beta}\}$ conditioned on the $\mathbf{\Omega}$ and $\mathbf{\Theta}$. We utilize the Gibbs-EM [Diao *et al.*, 2014; Hoffman *et al.*, 2012], a hybrid inference method alternating between collapsed Gibbs sampling and gradient descent in variational inference, to estimate latent variables $\mathbf{\Phi}$ and model parameters $\mathbf{\Omega}$. Differing with the traditional EM method, Gibbs-EM estimates the expectation of variational distribution of $\boldsymbol{z}$ and $\boldsymbol{c}$, i.e. $q^\star(\boldsymbol{z}, \boldsymbol{c})$, by the average value of the samples from Gibbs sampler in the E-step.

Specifically, we introduce a family of factorized variational distribution for $\mathbf{\Phi}$.

$$\begin{aligned} &q(\mathbf{\Phi}|\boldsymbol{\kappa^u}, \boldsymbol{\kappa^v}, \boldsymbol{\lambda}) \\ &= \prod_d \{q(\pi_d^u|\boldsymbol{\kappa_d^u}) q(\pi_d^v|\boldsymbol{\kappa_d^v}) q(\boldsymbol{z_d}, \boldsymbol{c_d})\} \prod_i q(\boldsymbol{\beta_i}|\boldsymbol{\lambda_i}) \end{aligned} \tag{4}$$

where $\boldsymbol{\beta_i}$ represents the ith topic.

The lower bound $\mathcal{L}$ of the log-likelihood for the review corpus can be approximated by the Jensen's inequality as follows. The goal of Gibbs-EM method is to maximize such lower bound by alternating between E-step and M-step.

$$\mathcal{L} = \mathbb{E}_q[\log p(\mathbf{\Phi}, \mathbf{\Theta}, \mathbf{\Omega})] - \mathbb{E}_q[\log q(\mathbf{\Phi})] \tag{5}$$

To maximize the lower bound $\mathcal{L}$ is equivalent to finding the optimal free parameters so that the approximated variational distribution is close to the true posterior by the KL divergence. Following [Bishop, 2006], we thus directly present the updating formula for the free parameters $\widehat{\boldsymbol{\kappa}}_d^u$, $\widehat{\boldsymbol{\kappa}}_d^v$ and $\widehat{\boldsymbol{\lambda}}_i$ below.

E-step:

$$\begin{aligned} \widehat{\kappa}_{dk}^u \propto & g_{uk} \prod_{f=0}^{K^{ux}} [\phi_{kf}^{ux}]^{F_{df}^{ux}} [1 - \phi_{kf}^{ux}]^{1 - F_{df}^{ux}} \\ & \cdot \prod_{f=0}^{K^{uy}} [F_{df}^{uy}]^{\phi_{kf,1}^{uy} - 1} [1 - F_{df}^{uy}]^{\phi_{kf,2}^{uy} - 1} \end{aligned} \tag{6}$$

where $\widehat{\kappa}_{dk}^u$ refers to the estimated probability for the $\pi_d^u = k$. Updating for the $\widehat{\kappa}_{dk}^v$ can be similarly derived.

$$\widehat{\lambda}_{ik} = \sum_d e_{diw} + \eta_w \tag{7}$$

where e_{diw} is a counter for the number of word w assigned to the topic i in the review d. For the $\boldsymbol{z_d}$ and $\boldsymbol{c_d}$, we estimate their expectation by the Gibbs sampling. Normally, in each E-step, we would save the average value of the sampled $\boldsymbol{z_d}$ and $\boldsymbol{c_d}$ after B burn-in sweeps, which has been done similarly in [Hoffman *et al.*, 2012].

$$\begin{aligned} &q^\star(z_{dn} = k|\boldsymbol{z_{-dn}}, \boldsymbol{c_d}) \\ &\propto [\frac{\delta_1 \delta_2 + e_{dk}}{\delta_2 + \sum_{i=k}^K e_{di}} \prod_{i=1}^{k-1} \frac{(1 - \delta_1)\delta_2 + \sum_{j=i+1}^K e_{dj}}{\delta_2 + \sum_{j=i}^K e_{dj}}] \cdot \\ &\exp[\Psi(\lambda_{c_{dk} w_{dn}}) - \Psi(\sum_w \lambda_{c_{dk} w}) + \sigma^2 (2 s_{dk}^T \widetilde{s}_{dk} - \widetilde{s}_{dk}^T \widetilde{s}_{dk})] \end{aligned} \tag{8}$$

where δ_1 and δ_2 represent the mean and variance of the GEM distribution in the hLDA model, respectively, and $\widetilde{s}_{dk} = [g_{uk}, h_{vk}, m_{dk}]$.

$$q^{\star}(\boldsymbol{c_d} = t|\boldsymbol{c_{-d}}, \boldsymbol{z_d}) \propto p(\boldsymbol{c_d}|\boldsymbol{c_{-d}}) \exp\{\sum_{n}[\Psi(\lambda_{c_{dk}w_{dn}}) - \Psi(\sum_{w}\lambda_{c_{dk}w})]\} \quad (9)$$

$$p(\boldsymbol{c_d}|\boldsymbol{c_{-d}}) = \begin{cases} \frac{\#[\boldsymbol{c_{-d}}=t]}{\#[\mathcal{D}]-1+\gamma}, \text{t is previously chosen path} \\ \frac{\gamma}{\#[\mathcal{D}]-1+\gamma}, \text{t is a new path} \end{cases} \quad (10)$$

where $\#[\boldsymbol{c_{-d}} = t]$ denotes the number of reviews assigned to the path t except the review d, $\#[\mathcal{D}]$ is the number of reviews in the entire corpus, and γ is the hyper-parameter in the hLDA to control the size of the topic hierarchy tree.

M-step:

In M-step, we estimate the model parameters in Ω by maximizing the lower bound $\mathcal{L}$ in Eq. 5. For the user and item spam level distribution vector g_u and h_v, we update their values by the projected gradient descent [Duchi *et al.*, 2008] since they should be constrained in a simplex. Since the items related parameters ($\boldsymbol{h}$, $\boldsymbol{\phi^{vx}}$, $\boldsymbol{\phi^{vy}}$) play the same role with user related parameters ($\boldsymbol{g}$, $\boldsymbol{\phi^{ux}}$, $\boldsymbol{\phi^{uy}}$), we only present the user related updating formula.

$$\nabla \frac{\mathcal{L}_{d\in\mathcal{D}_u}}{g_{uk}} = \frac{\sum_{d\in\mathcal{D}_u}\kappa^u_{dk} + \alpha - 1}{g_{uk}} + \sum_{d\in\mathcal{D}_u}\sigma^2\omega_g(s_{dk} - \widetilde{s}_{dk}) \quad (11)$$

where ω_g is the corresponding weight element for the g_u in ω. For the Bernoulli parameter ϕ^{ux}_{kf} of binary user related review features, we can compute its analytical updating formula by setting its gradient to zero.

$$\phi^{ux}_{kf} = \frac{\sum_d \kappa^u_{dk}F^{ux}_{df} + \gamma^u_{f,1} - 1}{\sum_d \kappa^u_{dk} + \gamma^u_{f,1} + \gamma^u_{f,2} - 2} \quad (12)$$

We update the Beta shape parameters using the method of moment which is commonly used in previous works. Besides, we update the spam level distribution vector $\boldsymbol{s_d}$ for the review d by the mean of its Normal Distribution, i.e. $\boldsymbol{s_d} = \widetilde{\boldsymbol{s}}_d$. In addition, we also perform the projected gradient descent for updating the weight $\boldsymbol{\omega}$ since it should stay in a simplex.

$$\nabla_{\omega_g}\mathcal{L} = \sum_d\sum_k \sigma^2 g_{uk}(s_{dk} - \widetilde{s}_{dk}) \quad (13)$$

In summary, after the initialization, we can iteratively perform the E-step and M-step until convergence. All the related parameters setting are described in Sec. 3.2.

3 Experiment

3.1 Data Sets and Preprocessing

We demonstrate the effectiveness of our model by conducting experiments on three popular review data sets including the Amazon audioCD data[2], the TripAdvisor hotel data[3], and the Yelp restaurant data[4]. As discussed in Sec. 1, the review spammers in these three domains are more likely to write general review comments. In particular, each dataset is a collection of review comments from a set of users for the product items.

Data Set	#Review	#User	#Item	Avg.words
Amazon-audioCD	36,774	8,937	10,087	63.6
TripAdvisor	22,635	2,069	935	72.4
Yelp	13,007	248	255	79.1

Table 1: Statistics of data sets

For the preprocessing, we first remove non-English reviews and reviews with less than 20 words, and compute the values of each abnormal features in Sec. 2.2. Then we only keep the users with more than 5 reviews and items with more than 5 reviews since the users or items with fewer reviews exhibit less behavior characteristics. Besides, in order to ensure our evaluated data set has review spammers and manipulated offerings, we randomly sample a proportion of reviews with the average value of corresponding user or item abnormal feature greater than 0.5. For the review text, we remove the stop words, and the terms whose count frequency is less than 5. Table 1 depicts the statistics for each data set.

3.2 Experimental Setup

We specify the similar thresholds for the user abnormal features mentioned in [Mukherjee *et al.*, 2013]. For the item abnormal features, we set $\tau_v = 70$, and $\tau^v_w = 5$, $n_r = 50$.

For the initialization of our model URSM, we first rank the review collection in each data set based on the sum of corresponding abnormal feature values, and then divide the ranked reviews into K segments with the same size of $\frac{\#[\mathcal{D}]}{K}$. The kth segment can be initially treated as the reviews with the kth spam level. We thus have the initial value of π^u_d and π^v_d, and then we can apply the M-step to initialize g_u, h_v, s_d and all the Beta, Bernouli distribution parameters. Additionally, we set $K = L = 3$, $K^{ux} = 5$, $K^{uy} = 4$, $K^{vx} = 2$, $K^{vy} = 4$ and use default values for the parameters related to the hLDA. The hyper-parameters are specified as follows, i.e. $\alpha = \xi = \frac{0.5}{K}$, $\sigma^2 = 1.0$, and uniformed priors for all γ. When performing our Gibbs-EM algorithm, we set the maximum iteration number as 1000, and the burn-in sweeps as 10.

3.3 Evaluation for Review Spam Detection

We employ the supervised review classification method to evaluate the spamicity-based ranking of suspicious review spams. This evaluation method has been used in [Mukherjee *et al.*, 2013]. The main idea is that supervised text classification using n-gram features have been proven quite effective for review spam detection [Ott *et al.*, 2011]. Thus, if our classification of the reviews using n-gram features performs better, it implies that our detected spam and non-spam labels for the reviews are more accurate. Moreover, if URSM is effective, the most (least) suspicious review would be ranked at the top (bottom). We can treat the top $k\%$ reviews as spams and the bottom $k\%$ reviews as non-spam reviews, and then perform a supervised classification for such labeled review set using the textual n-gram features. Since the URSM involves the

[2]http://liu.cs.uic.edu/download/data/

[3]http://www.cs.cmu.edu/ jiweil/html/hotel-review.html

[4]http://www.yelp.com/dataset_challenge

bag-of-words modeling in hLDA, i.e. unigram, we thus apply the bigram and trigram features for classification.

We compare our model URSM with the state-of-the-art unsupervised models. **ASM-HE** proposed by [Mukherjee *et al.*, 2013] is able to rank the suspicious reviews based on its spamicity in an unsupervised manner. Such model exploits user's abnormal behavior features and user-related review abnormal features, but does not consider the item abnormal features and ignore the useful review text generality information. **FSum-R**: intuitively ranking each review by the sum of all the review abnormal feature values. **URSM-IF**: in order to demonstrate the effectiveness of considering text generality factor, we further compare URSM with URSM-IF, a variant of URSM by removing the hLDA text generality modeling part. In URSM-IF, the spamicity of a particular review is merely influenced by the user spamicity and item spamicity.

Since [Ott *et al.*, 2012] reported that there are normally 8% to 15% spam rate for online review sites, we only report the results of k=5%, 10% and 15%. We conduct the supervised classification by the popular LIBSVM library [Chang and Lin, 2011] with a linear kernel and report the 5-fold cross validation results in Table 2. Due to the space limit, we only show the metrics of F1-score and accuracy. From the result, we note that our model URSM outperforms all the baseline methods for all data sets because the review spams in such domains tend to be general and the items can also provide helpful spamming clues. Specifically, URSM-IF performs better than ASM-HE and FSum-R indicating that our consideration for the item abnormal features is useful. Besides, we note that URSM outperforms URSM-IF after considering the text generality. Consequently, it is really useful to exploit item abnormal features and text generality for review spam detection.

k(%)	URSM		URSM-IF		ASM-HE		FSum-R	
	F1	A	F1	A	F1	A	F1	A
5	**76.7**	**78.0**	73.2	72.6	68.3	68.3	60.8	62.2
10	**75.5**	**76.6**	70.3	70.2	61.4	62.5	55.7	57.4
15	**70.4**	**71.7**	64.6	64.1	59.0	60.9	56.8	55.9

(a) Amazon-AudioCD

k(%)	URSM		URSM-IF		ASM-HE		FSum-R	
	F1	A	F1	A	F1	A	F1	A
5	**78.1**	**78.8**	74.2	74.0	72.4	72.5	64.9	65.9
10	**72.2**	**73.2**	70.3	69.2	64.5	65.5	60.2	60.3
15	**69.4**	**70.3**	66.4	67.1	63.5	63.9	52.6	53.4

(b) TripAdvisor

k(%)	URSM		URSM-IF		ASM-HE		FSum-R	
	F1	A	F1	A	F1	A	F1	A
5	**75.5**	**75.2**	72.6	71.9	68.0	70.5	67.9	68.3
10	**74.3**	**73.8**	68.0	68.1	65.5	66.3	64.0	63.9
15	**70.1**	**71.4**	66.7	67.0	64.9	64.6	55.1	56.9

(c) Yelp

Table 2: 5-fold SVM Review classification result for each data set. F1: F1-score A: Accuracy

3.4 Human Evaluation for Review Spammer and Manipulated Offering Detection

For the evaluation of the ranked users and items based on the spamicity, we resort to the human judgement method which is commonly used in the spamming evaluation. We employ 3 college student helpers to label the suspicious users and items. They are firstly required to read the related article on the spamming detection strategies[5], and then independently examine the user or item detailed profile and all his or its review texts to provide a label of spam or non-spam.

Due to the limited manpower, we just consider the largest Amazon-audioCD data set which has more users and items, and compare our model with the baseline models mentioned above. Note that although the original ASM-HE model can only rank the suspicious review and users, we can also use the proportion of review spams to measure the spamicity for a particular item, which is called **ASM-HE-V**. For the FSum approach, we can apply the sum of all the user or item abnormal feature values to rank the users or items. We name such methods as **FSum-U** and **FSum-V** respectively.

	URSM			URSM-IF			ASM-HE			FSum-U		
	TP	MP	BP	TP	MP	BP	TP	MP	BP	TP	MP	BP
J_1	84	12	0	77	14	0	71	17	2	66	20	3
J_2	80	9	1	70	11	2	66	16	3	62	19	3
J_3	78	9	1	73	9	1	65	13	3	68	17	1
$Avg.$	80.7	10.0	0.7	73.3	11.3	1.0	67.3	15.3	2.7	65.3	18.7	2.3
κ_F	0.70			0.72			0.71			0.71		

(a) Review Spammer Detection. **The size of each part is 100**

	URSM			URSM-IF			ASM-HE-V			FSum-V		
	TP	MP	BP	TP	MP	BP	TP	MP	BP	TP	MP	BP
J_1	43	3	1	40	7	2	31	10	3	34	12	2
J_2	40	5	0	38	5	1	35	11	5	35	9	2
J_3	39	4	0	35	7	1	30	12	3	35	10	4
$Avg.$	40.7	4.0	0.3	37.7	6.3	1.3	32.0	11.0	3.7	34.7	10.3	2.7
κ_F	0.73			0.72			0.76			0.74		

(b) Manipulated Offering Detection. **The size of each part is 50**

Table 3: Human evaluation result for the Amazon-audioCD data set. The larger value for TP indicates better performance while the smaller value for MP and BP indicates better performance.

Similar to the setting in [Mukherjee *et al.*, 2013], for the evaluation of users, we provide each helper with three parts of users for each method, i.e. Top Part (TP) contains the top 100 ranked users. Middle Part (MP) contains the middle 100 ranked users and Bottom Part (BP) contains the bottom 100 ranked users. For the evaluation of items, since the number of reviews for an item is relatively larger, we decrease the size of part to 50. Note that a particular item is labeled as a manipulated offering when its review spam proportion succeeds 0.5. Table 3 shows the human evaluation results as the count of users or items labeled as spam. We additionally measure the agreement of judges by the Fleiss multi-rater kappa [Fleiss, 1971] for each method. As you can see, our model URSM can detect more spammers or manipulated offerings within the

[5]http://consumerist.com/2010/04/14/how-you-spot-fake-online-reviews/

top part (TP) and leave less spammers within the middle part (MP) and the bottom part (BP). In other words, URSM has better ranking performance towards the users and the items based on the spamicity than other methods, and all judges have consensus in the spamming judgements based on the high κ_F value. Moreover, the URSM-IF method performs better than the ASM-HE-V and FSum-V further indicating the usefulness of considering item abnormal features. On the other hand, the URSM method with the additional consideration for the text generality is superior to URSM-IF, which further shows that it is effective to exploit the text generality as well as the item abnormal features for spamming detection.

4 Conclusion

In this paper, we have proposed a unified probabilistic graphical model URSM for detecting the suspicious review spams, the review spammers and the manipulated offerings in an unsupervised manner. Compared with previous works, we additionally consider the item abnormal features and text generality for spamming detection. Experimental results on three popular review data sets demonstrate the effectiveness of our proposed model.

References

[Beutel *et al.*, 2014] Alex Beutel, Kenton Murray, Christos Faloutsos, and Alexander J Smola. Cobafi: collaborative bayesian filtering. In *WWW*, pages 97–108. International World Wide Web Conferences Steering Committee, 2014.

[Bishop, 2006] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[Blei *et al.*, 2010] David M Blei, Thomas L Griffiths, and Michael I Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM (JACM)*, 57(2):7, 2010.

[Chang and Lin, 2011] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[Diao *et al.*, 2014] Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *KDD*, pages 193–202. ACM, 2014.

[Duchi *et al.*, 2008] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the l 1-ball for learning in high dimensions. In *ICML*, pages 272–279, 2008.

[Fei *et al.*, 2013] Geli Fei, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. Exploiting burstiness in reviews for review spammer detection. In *ICWSM*, 2013.

[Feng *et al.*, 2012] Song Feng, Longfei Xing, Anupam Gogar, and Yejin Choi. Distributional footprints of deceptive product reviews. In *ICWSM*, 2012.

[Fleiss, 1971] Joseph L Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

[Hoffman *et al.*, 2012] Matt Hoffman, David M Blei, and David M Mimno. Sparse stochastic inference for latent dirichlet allocation. In *ICML*, pages 1599–1606, 2012.

[Jindal and Liu, 2008] Nitin Jindal and Bing Liu. Opinion spam and analysis. In *WSDM*, pages 219–230. ACM, 2008.

[Li *et al.*, 2011] Fangtao Li, Minlie Huang, Yi Yang, and Xiaoyan Zhu. Learning to identify review spam. In *IJCAI*, volume 22, page 2488, 2011.

[Li *et al.*, 2013a] Jiwei Li, Claire Cardie, and Sujian Li. Topicspam: a topic-model based approach for spam detection. In *ACL*, pages 217–221, 2013.

[Li *et al.*, 2013b] Jiwei Li, Myle Ott, and Claire Cardie. Identifying manipulated offerings on review portals. In *EMNLP*, pages 1933–1942, 2013.

[Li *et al.*, 2014] Jiwei Li, Myle Ott, Claire Cardie, and Eduard Hovy. Towards a general rule for identifying deceptive opinion spam. In *ACL*, 2014.

[Lim *et al.*, 2010] Ee-Peng Lim, Viet-An Nguyen, Nitin Jindal, Bing Liu, and Hady Wirawan Lauw. Detecting product review spammers using rating behaviors. In *CIKM*, pages 939–948. ACM, 2010.

[Mukherjee *et al.*, 2012] Arjun Mukherjee, Bing Liu, and Natalie Glance. Spotting fake reviewer groups in consumer reviews. In *WWW*, pages 191–200. ACM, 2012.

[Mukherjee *et al.*, 2013] Arjun Mukherjee, Abhinav Kumar, Bing Liu, Junhui Wang, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. Spotting opinion spammers using behavioral footprints. In *KDD*, pages 632–640. ACM, 2013.

[Ott *et al.*, 2011] Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T Hancock. Finding deceptive opinion spam by any stretch of the imagination. In *ACL*, pages 309–319. Association for Computational Linguistics, 2011.

[Ott *et al.*, 2012] Myle Ott, Claire Cardie, and Jeff Hancock. Estimating the prevalence of deception in online review communities. In *WWW*, pages 201–210. ACM, 2012.

[Sun *et al.*, 2013] Huan Sun, Alex Morales, and Xifeng Yan. Synthetic review spamming and defense. In *KDD*, pages 1088–1096. ACM, 2013.

[Wang *et al.*, 2011] Guan Wang, Sihong Xie, Bing Liu, and Philip S Yu. Review graph based online store review spammer detection. In *ICDM*, pages 1242–1247. IEEE, 2011.

[Xie *et al.*, 2012] Sihong Xie, Guan Wang, Shuyang Lin, and Philip S Yu. Review spam detection via temporal pattern discovery. In *KDD*, pages 823–831. ACM, 2012.

[Xu *et al.*, 2014] Yinqing Xu, Wai Lam, and Tianyi Lin. Collaborative filtering incorporating review text and co-clusters of hidden user communities and item groups. In *CIKM*, pages 251–260. ACM, 2014.