Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

Detecting Review Spammer Groups

Min Yang
University of Hong Kong
myang@cs.hku.hk

Ziyu Lu
University of Hong Kong
zylu@cs.hku.hk

Xiaojun Chen
Shenzhen University
xjchen@szu.edu.cn

Fei Xu*
Chinese Academy of Sciences
xufei@iie.ac.cn

Abstract

With an increasing number of paid writers posting fake reviews to promote or demote some target entities through Internet, review spammer detection has become a crucial and challenging task. In this paper, we propose a three-phase method to address the problem of identifying review spammer groups and individual spammers, who get paid for posting fake comments. We evaluate the effectiveness and performance of the approach on a real-life online shopping review dataset from amazon.com. The experimental result shows that our model achieved comparable or better performance than previous work on spammer detection.

Introduction

For the purpose of profit or fame, people try to write fake reviews to promote or demote some target entities (e.g. events, services or products). This kind of problem has already been widely reported by the media.

Review spams usually look perfectly normal until one compares them with other reviews of the same products or compares them with other reviews written by the same user. Detecting review spam is a challenging task. In this paper, we focus on discovering review spammer groups as well as individual spammers. Since actual users behind different ids could be a single person with multiple ids (also known as sockpuppet), multiple people or combination of both, we do not distinguish them in our work.

In this paper, we propose a three-phase method to address the problem of identifying review spammer groups and individual spammers. We first identify duplicate or near duplicate reviews. Secondly, each user's interest entities (or topics) can be presented by a word distribution. With the generated user interest distribution, the similarities between different reviewers can be computed by consin similarity (or KL distance) based on interests vector. This step has the potential to detect the spammers who can manipulate their behaviors to act just like genuine reviewers (do not copy each other). Thirdly, there are some individual spammers that cannot be detected by the first two steps. To deal with these spammers, we use Amazon Webstore Browse Categories API[1] to retrieve each review's parent node. If a user review the products that belong to the same parent node within a small time window (e.g., 1 days) but make different comments, it is likely that this reviewer is a review spammer.

*Corresponding author

Related work

Recently, various methods have been proposed to detect deceptive opinion spam (Fei et al. 2013; Jindal and Liu 2008; Mukherjee, Liu, and Glance 2012). On the other hand, as author's interests showing increasing importance for the development of personalized and user-centric applications, variety of LDA (Blei, Ng, and Jordan 2003; Yang, Cui, and Tu 2015) extensions have been proposed to incorporate authorship information into the text (Rosen-Zvi et al. 2004).

Model Description

Duplicate and near duplicate reviews detection

Identifying whether a review is a spam is very difficult by manually reading the review separately. However, the reviews which contain the following types of duplicates (here, we assume the duplicates include near duplicates) are almost certainly opinion spam (Jindal and Liu 2008): a) Duplicates from different user ids on the same entity; b) Duplicates from the same user ids on the different entities; c) Duplicates from different user ids on different entities.

We don't treat the duplicates from the same user on the same product as spams because they could be due to clicking the submit button more than once. These duplicate reviews can be detected using the shingle method in (Broder 1997). The reviews with similarity score of at least ρ are regarded as duplicates. Here, ρ is a threshold defined as a hyper-parameter. On the other hand, for the duplicates from the same userids on the same entity, we only keep the last copy and remove the rest.

Recognizing such duplicate and near duplicate reviews help us identifying potential spammers. Thus, we treat these reviewers who write duplicate reviews as spam reviewers.

Reviewers interest similarity

The method used in the last subsection identifies certain types of spammers, i.e., those who post many similar reviews about one or more target entities. However, in reality,

[1] http://aws.amazon.com/asl/

there are also other kinds of spammers who can manipulate their behaviors to act just like genuine reviewers, and thus cannot be detected using the method in last subsection. One way to detect review spammer group is to find the reviewers who have similar interests (target entities and corresponding sentiment).

We employ author-topic model (Rosen-Zvi et al. 2004) to build an interest profile for each reviewer. The Author-Topic model naturally yields reviewer representations in the form of distributions over topics. By modeling the interests of users, we can address several task such as authorship attribution and community detection. In this work, we employ Author Topic model to develop user interest profile. With the generated user interest profiles, the similarities between different reviewers can be computed by consin similarity (or KL distance) based on interests vector. Once the distances between reviewers have been obtained, finding spammer candidate group is straightforward.

Individual spammer behavior

There are some types of spammers that cannot be detected by the first two steps. For instance, some reviewers are likely to review similar product (have the same parent category) within a short period while using different comments. To deal with these spammers, we use Amazon Webstore Browse Categories API[2] to retrieve each review's parent node. If a user review the products that belong to the same parent node within a small time window (e.g., 1 day) and use different comments, it is likely that this reviewer is a review spammer.

Experiments

Datasets

Amazon dataset: We use product reviews from Amazon.com as the experiment dataset. This dataset was originally made public and posted to the web by (Jindal and Liu 2008). For our experiment, we only use reviews of manufactured products (e.g. electronics, computers, etc), which comprised of 109,518 reviews, 53,469 reviewers and 39,392 products. Reviews in other categories can be studied similarly.

We perform data preprocessing before applying our algorithm to detect deceptive opinion spammer group. The texts are first tokenized using the natural language toolkit NLTK[3].

Baseline methods

In this paper, we evaluate and compare our approach with two state of the art methods: Fei's method proposed in (Fei et al. 2013) and Mukherjee's method proposed in (Mukherjee, Liu, and Glance 2012).

Experimental results

In this experiment, the threshold ρ is set to be $\{0.7, 0.8, 0.9\}$. We experimented with topic number from the set $\{10, 20, 50, 100, 200, 400, 600, 800, 1000\}$. By examining the obtained topic words, we set topic number 800 as it achieves the best performance. We choose hyperparameters of author topic model as α= 0.01, β = 0.05.

Class	Fei's	Mukherjee's	Ours (ρ=0.7, 0.8, 0.9)
Spammer	0.74	0.68	0.72 / 0.78 / 0.84
Non-Spammer	0.68	0.64	0.68 / 0.76 / 0.80
Overall	0.71	0.66	0.70 / 0.77 / 0.82

Table 1: The precision results with different ρ

To verify the the effectiveness of proposed algorithm, our evaluation is based on human expert judgment, which is commonly used in the research on anti-spams. Due to the large number of reviewers in the experimental dataset, it would have taken too much time for human judges to assess all the reviewers. Thus, we cannot perform recall evaluation. For precision evaluation, we randomly selected 50 reviewers from spammers and non-spammers detected by each method, and invited three Natural Language Processing (NLP) researchers to evaluate these reviewers. Table 1 shows the precision for each method. Our approach outperform another two methods when the threshold $\rho \geq 0.8$ on the overall performance. The advantages of our approach may come from its capability of detecting both spammer groups and individual spammers.

Conclusion and Future Work

This paper studied opinion spammer group detection. To the best of our knowledge, our paper is the first study which uses Author Topic model to represent user interest in spammer detection.

References

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *The Journal of machine Learning research* 3:993–1022.

Broder, A. Z. 1997. On the resemblance and containment of documents. In *Compression and Complexity of Sequences*, 21–29. IEEE.

Fei, G.; Mukherjee, A.; Liu, B.; Hsu, M.; Castellanos, M.; and Ghosh, R. 2013. Exploiting burstiness in reviews for review spammer detection. *ICWSM* 13:175–184.

Jindal, N., and Liu, B. 2008. Opinion spam and analysis. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, 219–230. ACM.

Mukherjee, A.; Liu, B.; and Glance, N. 2012. Spotting fake reviewer groups in consumer reviews. In *WWW*, 191–200. ACM.

Rosen-Zvi, M.; Griffiths, T.; Steyvers, M.; and Smyth, P. 2004. The author-topic model for authors and documents. In *UAI*, 487–494.

Yang, M.; Cui, T.; and Tu, W. 2015. Ordering-sensitive and semantic-aware topic modeling. In *AAAI*.

[2]http://aws.amazon.com/asl/

[3]http://www.nltk.org