Opinion Spam Detection in Web Forum: A Real Case Study

Yu-Ren Chen and Hsin-Hsi Chen
Department of Computer Science and Information Engineering, National Taiwan University
No. 1, Sec. 4, Roosevelt Road, Taipei, 10617 Taiwan
+886-2-33664888ext311
chenaren@gmail.com; hhchen@ntu.edu.tw

ABSTRACT
Opinion spamming refers to the illegal marketing practice which involves delivering commercially advantageous opinions as regular users. In this paper, we conduct a real case study based on a set of internal records of opinion spams leaked from a shady marketing campaign. We explore the characteristics of opinion spams and spammers in a web forum to obtain some insights, including subtlety property of opinion spams, spam post ratio, spammer accounts, first post and replies, submission time of posts, activeness of threads, and collusion among spammers. Then we present features that could be potentially helpful in detecting spam opinions in threads. The results of spam detection on first posts show: (1) spam first posts put more focus on certain topics such as the user experiences' on the promoted items, (2) spam first posts generally use more words and pictures to showcase the promoted items in an attempt to impress people, (3) spam first posts tend to be submitted during work time, and (4) the threads that spam first posts initiate are more active to be placed at striking positions. The spam detection on replies is more challenging. Besides lower spam ratio and less content, replies even do not mention the promoted items. Their major intention is to keep the discussion in a thread alive to attract more attention on it. Submission time of replies, thread activeness, position of replies, and spamicity of first post are more useful than content-based features in spam detection on replies.

Categories and Subject Descriptors
H.3.3 [**INFORMATION STORAGE AND RETRIEVAL**]: Information Search and Retrieval –*Information filtering.*

General Terms
Algorithms, Design, Experimentation, Human Factors.

Keywords
Fake Web Review; Opinion Spam Detection; Web Forum.

1. INTRODUCTION
Mining opinions from heterogeneous resources attract much attention due to their practical applications in various domains. More and more customers refer to opinions spreading on the web before purchasing items, reserving hotel rooms, and so on. To affect customers' buying decisions, fake opinions are generated for purpose to promote special targets and/or denounce their competitors. How to filter out untrustful information becomes an important issue in opinion mining.

To prepare datasets for the study of opinion spam detection is indispensable. However, it is more difficult than the preparation of the datasets for the other types of spam detection tasks such as email spams and web spams due to the subtlety nature of opinion spams. Crowdsourcing is introduced to annotate opinion spams. The major problem is the context cannot be fully rebuilt for the workers to make correct annotations.

In this paper, we study a real case: Samsung probed in Taiwan over 'fake web reviews', reported by BBC on 16 April 2013 [1]. This case happened in *Mobile01*, a web forum in Taiwan, which mainly features discussion about mobile phones, hand-held devices, and other consumer electronics. In April 2013, a poster submitted a thread in which several confidential documents of a covert marketing campaign that had been conducted under the table were disclosed. The campaign instructed hired writers and designated employees to post disingenuous comments on some web forums including *Mobile01*. This revelation created a big stir at that time, since it was the first time such strong evidence supporting what most folks had considered as a "conspiracy theory" came to light.

Generally speaking, web forums provide platforms for people with similar interests to interact and share experiences with each other. Since people normally believe posts on legit forums to be based on genuine personal opinion and experience, it is considered unethical to use them to promote things for personal gain without disclaimer, and take advantage of the inherent mutual trust between forum users. As a matter of fact, such marketing malpractice violated the fair trade law, and the company in charge of the campaign was fined by the *Fair Trade Commission* (*FTC*) in Taiwan, after the investigation was completed.

The confidential documents, along with relevant articles describing the campaign, are listed on Taiwansamsungleaks.[2] In this campaign, hired posters were asked to promote a certain brand and denounce its rivals on web forums such as *Mobile01*, while disguised as normal consumers. Among the disclosed documents, there are two spreadsheets that appear to be internally-kept records of the spam posts incurred by the

WWW 2015, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3469-3/15/05.
http://dx.doi.org/10.1145/2736277.2741085

[1] http://www.bbc.com/news/technology-22166606.
[2] http://taiwansamsungleaks.org/.

campaign from 2011 to 2012. Each row in these spreadsheets is a record of an incentivized forum post and consists of the poster's username, the time of posting, the url to the post, the product that was discussed in the post, and some other details.

In this research, we leverage the spreadsheets to generate ground truth of deceptive forum spams. To the best of our knowledge, this is the first real case study of opinion spam detection in web forum. The real spammers' behaviors in a thread consisting of first post and replies are investigated. This paper is organized as follows. Section 2 surveys the related work of opinion spam detection. Section 3 gives a detail description of the real dataset. Section 4 analyzes the dataset from various perspectives. Section 5 presents spam opinion detection in first post and replies in threads. Section 6 concludes the remarks and shows some future works.

2. RELATED WORK

2.1 Opinion Spam

Spam, whose definitions usually center on the concept of *unsolicited message* [8], has been bothering Internet users for a long time. Email spam is one of the most prevalent types of spams that could be dated back to long ago [1]. Web spam is another form of spams whose objective is to game the ranking algorithm of a search engine in order to get an undeserved high ranking [5]. As social media gained its popularity in recent years, social network spam came along [15]. One of the variants involves throwaway accounts created in batch to somehow bait regular users to clicks certain link for personal gain.

Opinion spam is different from the above types of spams from various perspectives. One of the most prominent differences is that opinion spam is arguably the most "subtle" kind of spams. This is because it is not only completely ineffective, but also very harmful to the reputation of the target that it promotes, when it gets caught. Therefore, opinion spammers would generally try their best to disguise their opinion spams as genuine opinions. Carefully-written opinion spams have caused great challenges in manually identifying the spams and annotating the ground truth, which is in concert with the finding that human are poor judge of deception [20]. Ott et al. [17] reported a very low annotator agreement score when annotating the opinion spams from a review corpus. In contrast, most of the email spams, web spams, or social network spams are fairly easy to spot by an experienced user of the respective platform.

One of the earliest researches on opinion spam is Jindal and Liu [10]. They attempt to detect fake product reviews on *Amazon*. Since then, this topic has been drawing increasing attention. Jindal et al. [11], Lim et al. [14], Mukherjee et al. [16], Wang et al. [21] and Xie et al. [22], are some of the researches.

2.2 Targets of Detection

The task of opinion spam detection can be seen as a binary classification problem. A detection model aims at detecting whether a given instance is a spammy or not. Naturally, an instance would be a post. Spam and non-spam post would be the two classes. Alternatively, an instance could be a user account, where spammer and non-spammers would be the classes. In this paper, we attempt to deal with the former type of targets in web forum with a real case study.

In spam post detection, the goal of a detection model is to identify whether a forum post, a product review, a store review, etc. is a spam post. Many of the previous researches on opinion spam aimed at detecting spam reviews, which can be seen as a type of post [7, 10, 11, 17]. Nonetheless, even if the target of detection is spam, we can still utilize features derived from information about the corresponding spammers, and vice versa.

Lim et al. [14] and Wang et al. [21] are two of the previous researches that focused on identifying spammers, while Mukherjee et al. [16] considered a variation by making groups of spammers who worked together to write fake reviews as their detection target.

2.3 Features Used

A number of features have been proposed with some common supervised learning models such as *SVM* (*Support Vector Machine*), or alternatively, in some ad-hoc models designed for the specific purpose. Most of these features fall into two categories: the ones derived from the textual contents of opinion spams, and the ones not directly related to them.

For the features derived from textual contents, Jindal and Liu [10] counted the percentage of opinion-bearing words, brand name mentions, numerals, capitals, etc. Mukherjee et al. [16] computed content similarity between reviews to examine if there are duplicate or near-duplicate reviews, which are suspicious of being spam reviews. Ott et al. [17] used bag-of-n-grams and slightly improved the performance with psychologically meaningful dimensions in *LIWC* [19]. Harris [7] took cues such as word diversity, proportion of first person pronouns and mention of brand names. Feng et al. [3] went a step further by adopting deep syntactic features, which were derived from the production rules involved in parsing the contents based on the PCFG, in addition to the basic bag-of-words.

Speaking of features not directly related to textual contents, Lim et al. [14] and Feng et al. [3] both made extensive use of various characteristics of user rating patterns on *Amazon*. Mukherjee et al. [16] derived features from bursts in the amount of reviews, how early the reviews was post, and rating deviation, with respect to either groups or individuals. Wang et al. [21] iteratively computed the trustiness of reviewers, honesty of reviews and reliability of stores based on a graph model which utilized non-content-centric features such as average rating and number of reviews.

2.4 Ground Truth Acquisition

One of the major obstacles in studies of opinion spam is the difficulty in acquiring ground truth, since spammers try their best to keep it secret, and manual annotation is ineffective because of the subtlety nature of opinion spam mentioned in Section 2.1.

A lot of efforts had been put into obtaining ground truth in studies of opinion spam. Jindal and Liu [10] assumed near-duplicate reviews were likely to be spam and followed this heuristic to build an annotated dataset. More recently, collective annotations using crowdsourcing platform like *Amazon Mechanical Turk* had become a popular approach. Gokhman et al. [4] discussed various techniques of obtaining ground truth in studies of deception, and argued that realistic deceptive contents could be generated from crowdsourcing, if the context of deception in practice is replicated on the crowdsourcing platform. On the other hand, it is ineffective

to annotate existing deceptive contents. In fact, one of the quality indicators of fabricated opinion spams is that they should not be recognizable by crowdsourced annotations. Ott et al. [17] scraped truthful opinions from *TripAdvisor* and synthesized deceptive opinion with the help of *Amazon Mechanical Turk*.

Thus far, most of the previous researches on opinion spams adopted some sort of approximations of the actual ground truth. On the contrary, we study a real case in this paper. We extract ground truth from the confidential records leaked directly from a covert advertising campaign.[3] The Fair Trade Commission (FTC) in Taiwan, the central competent authority in charge of competition policy and Fair Trade Act in Taiwan, announced a decision on October 31, 2013 based on the leaked information and other documents, "the Samsung Taiwan, OpenTide Taiwan, and Sales & Profit International Co. concealed their identity and pretended to be regular private citizens to market their products on the Internet by making comparisons with and comments on the products of other enterprises. The deceptive conduct was able to affect trading order in violation of Article 24 of the Fair Trade Act. In addition to ordering the three companies to immediately cease the unlawful act, the FTC also imposed on them administrative fines of NT$10 million, NT$3 million and NT$50,000 respectively."[4]

3. DATASET

3.1 Leaked Spreadsheets

The leaked spreadsheets *HHP-2011.xlsx* and *HHP-2012.xlsx* keep the histories of the opinion spam posts made in 2011 and 2012, respectively. Several discussion platforms were spammed, but for simplicity, we consider only the opinion spams and the corresponding spammers on *Mobile01*, which make up the majority of the records contained in the spreadsheets.

From the spreadsheets, urls to the spam posts and usernames of the spammers[5] are extracted. Some typos and inconsistent ways of presenting the usernames are manually checked and fixed. Furthermore, urls linked to pages on *Mobile01* might appear in different forms. To reliably match the posts we scraped later, a 3-tuple (*fid*, *thid*, *pnum*) are extracted from each url, where *fid*, *thid* and *pnum* refers to forum id, thread id and page number, respectively. These 3-tuples serve as unique identifiers of a page in a thread on *Mobile01*.

Since we regard a user who has ever posted a spam post as a spammer, any account contained in the spreadsheets is considered to be spammer. Thereafter, we have a set of 2-tuples, each consisting of a spammer's username and a nested 3-tuples identifier leading to the page containing one or more spamming posts of the spammer.

3.2 Mobile01 Corpus

A large portion of past related studies used dataset scraped from product or store review websites such as *Amazon* or *TripAdvisor*, whereas our corpus is scraped from a **web forum**. Another difference is that the contents on *Mobile01* are mostly written in Traditional Chinese, with little bit of English phrases scattered around, rather than predominantly written in English as in previously used corpora.

Since more than 70% of the recorded spams were submitted to the *Samsung* (*Android*) board on *Mobile01*, we decide to focus our analysis on this board. We fetched all the threads along with the contained posts accessible by a regular member on the *Samsung* (*Android*) board on May, 2014. In addition, profiles of users who have ever posted in this board are also retrieved.

Table 1 lists the statistics of POSTS, THREADS and PROFILES, which correspond to the post messages, the threads they belong, and the posters who post the messages. Total 632,234 messages belonging to 41,759 threads were posted by 58,531 posters. Of these, there are 300 spammers and 3,116 spam posts. Tables 2-4 show the attributes of a post, a thread, and a poster, respectively. Note that the data we scraped from *Mobile01* is the May-2014 version, while the spam activities we investigate happened during 2011 and 2012. Ideally, a snapshot at the end of the 2012 may suit our need best. When we collected the dataset, some posts could have been edited or removed. Besides, profiles could have evolved with time. That will affect if users still stay active. The dataset is available at http://nlg.csie.ntu.edu.tw/m01-corpus/.

Table 1. Statistics of *Mobile01* corpus

Table	Total Instances	Spammer/Spam
PROFILES	58,531	300
POSTS	632,234	3,116
THREADS	41,759	Non-applicable

Table 2. Attributes of POSTS

Attribute	Description
thid	id of the thread to which the post belong
time	submission time of the post
uid	id of the poster who made the post
uname	username of the poster
nfloor	position relative to other posts in the thread
pnum	page number on which the post is
content	structured content in *HTML*

Table 3. Attributes of THREADS

Attribute	Description
thid	id of the thread
fid	id of the forum (board) in which the thread is
title	title of the thread
pages	number of pages in the thread
clicks	number of clicks (views) on this thread
time	submission time of the thread (=first post time)

Table 4. Attributes of PROFILES

Attribute	Description
uid	id of the user
reg_time	time of registration on the site
login_time	last time the user logged in
n_threads	number of threads initialized by the user
n_eff_posts	number of effective posts
n_posts	number of all posts
n_replies	number of replies, i.e., n_posts - n_threads
score	karma given by other users to the threads
p_phone	proportion of posts made on the smart phone

[3] http://taiwansamsungleaks.org/.

[4] http://www.ftc.gov.tw/internet/english/doc/docDetail.aspx?uid=179&docid=13035

[5] Whenever the word **spammer** or **user** is used hereafter without further details, we are talking about **spammer account** or **user account**, respectively, since we cannot tell out who is the actual human poster behind a user account.

3.3 Product Information

Because the spams related to cell phones, we scrape product information of all cell phones and tablets of the top brands. People use a wide variety of aliases to refer to cell phone or tablets products on *Mobile01*. To be able to match as many product mentions as possible, we take unigrams and bigrams from the full name as the aliases. If the name contains Roman numerals, we convert them to the respective Arabic numbers.

Many products of the same brand share aliases. For example, people often use *Note* to refer to a *Samsung* product, but there are a dozen of *Samsung* products with such alias. Since we care about only the brand the product belongs to, we just deem it as a mention of an arbitrary *Samsung* product of the *Note* series.

3.4 Data Splitting

We split the posts into training set and test set for spam detection based on their temporal orders. Posts submitted between Jan 2011 and Dec 2011 are placed into training set, and those submitted between Jan 2012 and May 2012 are put into test set.

The posts in test set may be posted by the same user in training set. To avoid the possible effects of the writing style of the spammers on spam detection, we remove all the posts by user accounts whose posts are included in training set from the test set, and call the resulting set "test set*". As a result, there do not exist any posts submitted by the same user account between training set and test set*. Table 5 shows the statistics of training set, test set and test set* for spam detection.

Table 5. Training, test set and test set* for spam detection

	#spam posts	#all posts	spam ratio
training set	1,883	159,432	1.12%
test set	1,233	92,552	1.33%
test set*	414	32,932	1.26%

4. CORPUS ANALYSIS

4.1 Subtlety Property

One of the properties we observed is that most of the spam posts do not really look suspicious, which echoes the discussion in Section 2.1. Spammers usually deliver their opinions about brands in a subtle way. Some spam posts (mostly replies) do not even carry opinion about any brand. They may only keep the discussion alive to attract more attention to the specific topics of the thread that meet the goal of the campaign. Replying to a thread would bump it to the top place of a board, since threads are ordered by the time of the last reply on most web forums. In this paper, we still deem these spam posts as opinion spam, but we probably should come up with an appropriate name for such spam posts.

Figure 1 shows a real thread in Chinese. The first post, i.e., "#1: *can NDSL stylus be shared with GALAXY NOTE*", intended to initiate a discussion about a *Samsung* product. It does not contain any opinion about any brand. The subsequent posts translated in English as follows are replies to the thread initiated by spam post.

#5: *Who says girls must pretend to be cute! I prefer to buy GALAXY R!*

#42: *The best way to celebrate her birthday is to send her a SII and push her down the valley!*

#24: *Please sign in if you are eager to get Galaxy Nexus.*

#24: *Through i9100 lens, we share a little happiness of life.*

#7: *Do White SII + Pink SGP protector glass look good?*

These posts only serve the purpose of heating up the discussion or just keep it alive. Since the intended messages may already be delivered by the first post or some other replies in the thread, the spammers avoid stating any strong opinions about the brands directly to make these posts even less suspicious. Such carefully written spam posts may make the automatic detection very challenging, because the content-centric features could be ineffective.

Figure 1. An example thread containing spam posts

4.2 Spam Post Ratio

Spammers are those posters who had submitted any spam post recorded in the leaked spreadsheets, as defined in Section 3.1. When we inspect the posts from the training set specified in Section 3.4, only about 33% of the posts by spammers are recorded as spam. Figure 2 shows the distribution of the spam post ratio among spammers. It demonstrates that a large number of spammers actually rarely spammed.

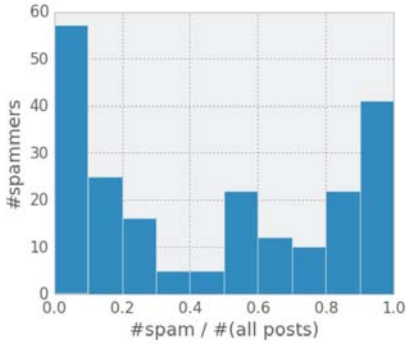

Figure 2. Distribution of the spam post ratios

4.3 Spammer Accounts

There are two possible spammer accounts in this dataset: reputable accounts and throwaway accounts. Figure 3 plots the spam post ratio vs. total number of threads made by the spammers.

We can see that most of the spammers with a high #spams/#posts ratio have initialized very few threads. It could be a clue that these are the throwaway accounts created for the sole purpose of spamming. On the other hand, accounts with a lower ratio show a larger variance in #threads. Some of them are likely to be reputable posters who usually make lots of threads.

Throwaway accounts are often created in mass within a short period of time, as it takes much more effort and patience to spread out the daunting task of registering throwaway accounts, especially if a large number of them are needed. To test if this postulation is applicable in our dataset, we adopt a simple heuristic to categorize accounts initiating less than 35 threads as throwaway accounts, and reputable accounts otherwise. Figure 4 shows the number of spammer accounts registered within each two weeks after 2009.

Indeed, there are three short periods in which large numbers of throwaway spammer accounts were registered, i.e., January 2010, April 2011 and October 2011. On the other hand, registration time for reputable accounts spreads quite evenly.

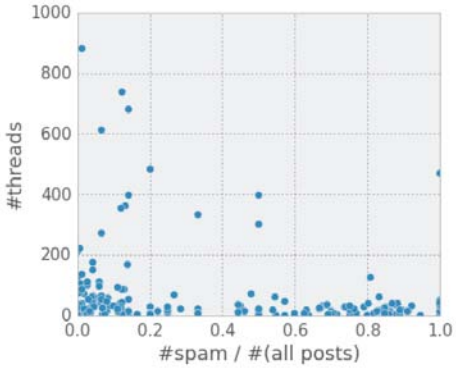

Figure 3. Spam posts ratio vs. number of threads

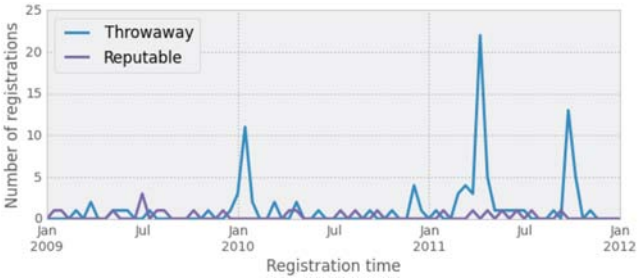

Figure 4. Number of spam accounts created in each 2 weeks

4.4 First Post vs. Replies

In the online web forums, first post, also known as original post, in a thread is written by the user submitting the thread. First posts are relatively richer in content as they serve the critical role in initializing a discussion on a specific topic. On the other hand, replies are often quite concise, and sometimes do not carry any opinion.

First posts and replies in threads display different characteristics in many aspects. Figure 5 shows that first posts tend to contain more characters. Moreover, at least one image is embedded in 19.2% of first posts, but only in 4.1% of replies.

Table 6 shows the spam counts and proportions for first posts and replies in training set. The ratio of spams in first posts is as high as 4.99%. In other words, for every 20 threads in the training set, one of them is created for covert marketing! In contrast, %spam is much lower for replies, i.e., 0.90%.

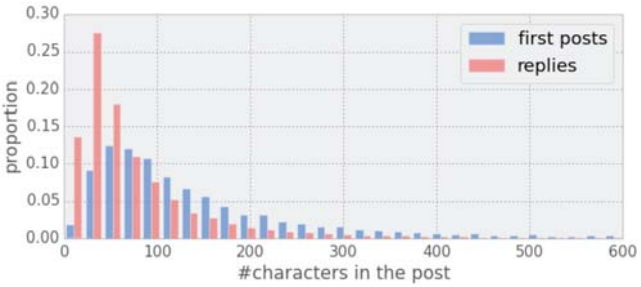

Figure 5. #Characters in first posts and replies

Table 6. Spams in first posts and replies

Type	#posts	#spams	%spams
First posts	10,951	546	4.99%
Replies	148,481	1,337	0.90%
All posts	159,432	1,883	1.18%

4.5 Submission Time of Posts

Because making spam posts is a job rather than a leisure activity for spammers, we postulate that a higher percentage of spam posts would be submitted during work time, compared to non-spam posts.

To verify our postulation, we plot the distribution of submission time of spam and non-spam. In Figure 6, the submission time of non-spam posts distributes pretty evenly over each day of week, whereas the amount of spam posts drops drastically on Saturday, and has a moderate decrease on Sunday. In Figure 7, we can observe that more spam posts are submitted during work hour, especially between 10 a.m. and 11 a.m., while non-spam posts are more often made during the spare hours. Hence, we see there is more or less a trend that spam posts are more often made during work time than leisure time, in comparison with non-spam posts.

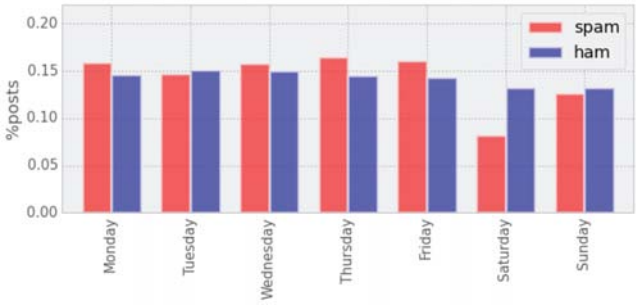

Figure 6. Proportion of spams submitted throughout a week

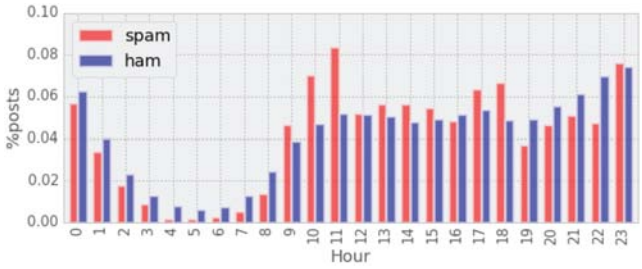

Figure 7. Proportion of spams submitted throughout a day

4.6 Activeness of Threads

The threads started by spam first posts are expected to be more active, since those are written to draw attention and exposure,

while non-spam threads may or may not be created with such intent in mind.

One intuitive way to measure the activeness of a thread would be counting the total number of posts in the thread, which is equal to 1 (first post) + #replies. Figure 8 plots the numbers of posts in spam and non-spam threads. Clearly, spam threads tend to attract more replies, which could be either spam replies or non-spam replies.

Another way to measure the activeness of a thread is the number of clicks. Figure 9 shows that spam threads tend to get more clicks in comparison with normal threads.

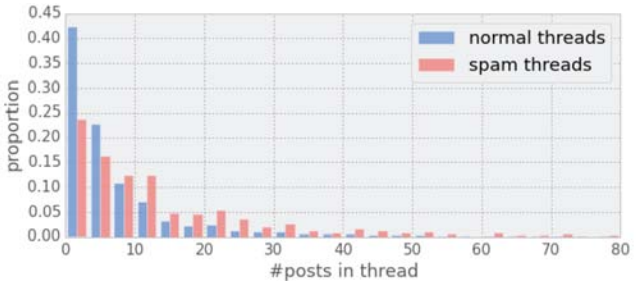

Figure 8. #posts in spam threads vs. normal threads

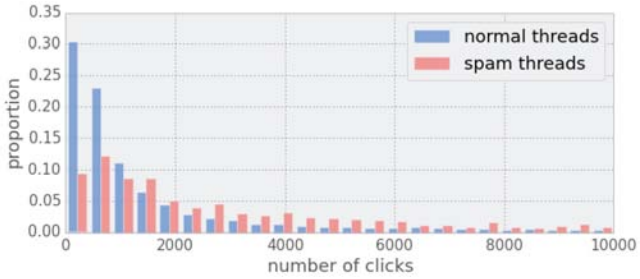

Figure 9. #clicks in spam threads vs. normal threads

4.7 Collusion among Spammers

Looking into the leaked spreadsheets, we note that a few threads contain multiple spam posts submitted by different accounts. It may be an indication of collusion going on among multiple spammers. These spammers would usually express similar opinions in the same thread to reinforce the credibility, or it could be just a result of multiple spammers bumping the same thread in an attempt to attract more attention to it.

Sometimes, it could be just the same person submitting posts with different spammer accounts in a thread, but it can still be regarded as collusion between multiple spammer accounts on the surface. In Figure 10, we could observe that there are threads containing 2 or more spam posts. In fact, 67% of the spam posts occur in threads containing at least 2 spam posts.

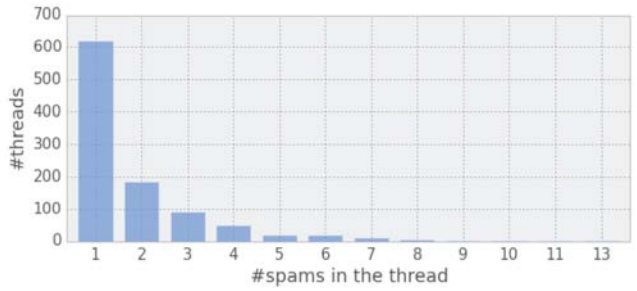

Figure 10. #threads containing specific number of spams

5. SPAM DETECTION

5.1 Models for First Post

At first, we will train a detection model specifically for first posts in thread. Thus we only use first posts from the training set and test set of posts introduced in Section 3.4. Counts and ratios of spam for first posts are listed in Table 7 for reference. Due to the low ratio of spam posts in training set, we randomly remove 60% of the non-spam posts from training set beforehand.

Table 7. Training and test set for spam detection for first posts

	#spam posts	#all posts	spam ratio
training set	546	10,951	4.99%
test set	208	5,870	3.54%
test set*	70	3,035	2.30%

In our dataset, spam posts ratio is quite low. Therefore, accuracy is not a good metric since it is dominated by the majority non-spam class. High precision on the spam class is desired because we do not want to falsely incriminate an innocent forum post as a spam; high recall on the spam class is also desired because we would like to find as many opinion spams as possible. Depending on the application, precision could be more important than recall, or vice versa. For instance, in an application where the detection system is used in an initial filtering stage narrowing down the set of suspicious posts for a later stage of manual classification, high recall might be preferred since misclassifying a spam post as normal one completely rules out the possibility of identifying the instance, while identifying a normal post as spam could still be corrected in the later stage of the pipeline. Because no particular application is aimed at, we do not have a prior preference on either precision or recall. Therefore, our evaluation metric would be the harmonic mean of precision and recall, also known as F-measure.

We adopt *Scikit-Learn* library [18] and explore various learning algorithms such as *Logistic Regression*, *SVM with linear kernel*, *SVM with RBF kernel*, *SVMperf*, etc. Most of the time, *SVM with RBF kernel* seems to win out by a non-negligible margin. *SVMperf* claims to somehow directly optimize the F-measure [12], but the F-measure in our experiments is not better than that using *SVM with RBF kernel*, Besides, *SVMperf* takes much longer time to train a model. Therefore, we decide to stick with *SVM with RBF kernel* from *Scikit-Learn*, which is actually a *Python* wrapper for the widely-used LibSVM [2], to conduct our experiments. As suggested in Hsu et al. [9], we scale each feature to zero mean and unit variance before feeding it to *SVM*.

There are two primary hyperparameters C and γ to be tuned in *SVM with RBF kernel*. For this purpose, whenever a model is to be learned, we first run 5-fold cross-validation multiple times on the training set to facilitate a grid search on C and γ with F-measure as the metric to optimize. The grid to search is represented below.

$$(C, \gamma) \in \{10^x | -3 \leq x \leq 3, x \in \mathbb{Z}\} \times \{10^y | -5 \leq y \leq 2, y \in \mathbb{Z}\}$$

Various feture combinations were explored in the experiments. Tables 8 and 9 summarize the experimental results using test set and test set*, respectively. Precision (**P**), Recall (**R**), and F-measuere (**F1**) are listed. The detail of each model will be discussed in the following sections.

Table 8. Spam detection for first post using test set

Model	P	R	F1
M1: random baseline	0.0343	0.4904	0.0642
M2: content bag-of-words	0.6289	0.4808	0.5450
M3: content + title bag-of-words	0.5912	0.5144	0.5501
M4: M3+content characteristics	0.7305	0.4952	0.5897
M5: M4+time+thread	**0.7237**	**0.5288**	**0.6111**
M6: M5+sentiment on brands	0.7097	0.5288	0.6061

Table 9. Spam detection for first post using test set*

Model	P	R	F1
M1: random baseline	0.0252	0.5571	0.0482
M2: content bag-of-words	0.5000	0.5143	0.5070
M3: content + title bag-of-words	0.5616	0.5857	0.5734
M4: M3+content characteristics	0.6491	0.5286	0.6032
M5: M4+time+activeness	**0.6667**	**0.5714**	**0.6154**
M6: M5+sentiment on brands	0.6557	0.5714	0.6107

5.1.1 Random Baseline

As an absolute baseline, the model M1 predicts whether a first post is spam based on the result of flipping a fair coin. As expected, the precision is about equal to ratio of the spams, which is 0.0354 and 0.0230 on test set and test set*, respectively. The recall is around 0.50, which reflects that fact that there is a half chance we correctly identify a spam as such by flipping a fair coin.

5.1.2 Bag-of-Words

After Chinese word segmentation on the HTML-stripped cleansed content from each post with *Jieba* (https://github.com/fxsjy/jieba), we count the occurrence of each word in training set, and construct a vocabulary. Next, rare words with less than 5 occurrences are removed from the vocabulary, since these would be the sparse bag-of-words features and might cause overfitting. On the other hand, words appearing in over 30% of the posts are also removed, as these are likely to be stop words or the like. After the vocabulary is set up, we represent each post as a vector of occurrences of each word in the vocabulary, where the occurrences are normalized by the length of the post.

In bag-of-words, each word in the vocabulary corresponds to a feature. Since the high number of features could slow down the training process significantly and may cause overfitting, we apply randomized PCA [6] on the #posts×#words bag-of-words matrix to reduce the word dimension. The desired number of dimension to reduce with PCA is tuned by looking at the average F-measure from 5-fold cross-validation on the training set, as plotted in Figure 11.

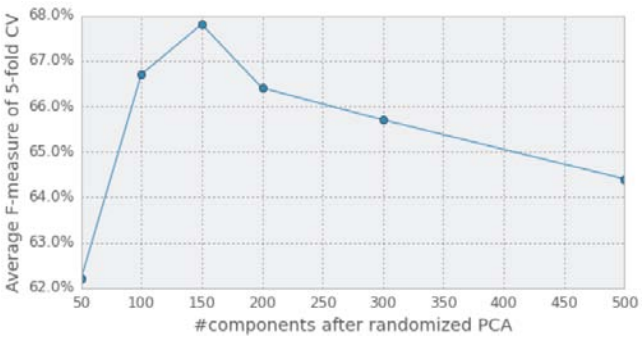

Figure 11. F-measures as #components in PCA changes

As shown in the plot, reducing to 50 components may cause too much information loss and thus deteriorating the average F-measure. On the other hand, too many components may cause some degree of overfitting which also worsens the performance. The average F-measure is the highest when the bag-of-words is reduced to 150 components, so we adopt it to train our model on the whole training set and see how it performs on test set.

Referring to M2 in Tables 8 and 9, the performance is surprisingly good. Our observation on subtlety of the spam posts in Section 4.1 gives us a hunch that the contents of the posts might not give strong clues about whether a post is spam, since the contents of spam posts are well-disguised. We are curious about what happens under the hood. To dive deeper into it, we would like to get the importance of each feature in order to observe what types of words are the decisive factors in the model predictions. However, for a non-linear model like *SVM with RBF kernel*, there is no simple way of computing importance of each feature. Nevertheless, by falling back to linear kernel, the model suffers around 10% performance loss in F-measure on test set, but we are able to see the relative importance of each word by looking at the coefficients after inverse-transformed with PCA.

Figure 12 is a word cloud containing the words with the highest coefficients, that is, words that are the strongest spam indicators, where the font size of a word is positively correlated with its weight. Figure 13 shows word cloud for words with the lowest weights, that is, words that are the strongest non-spam posts indicators. We can observe the distinctive difference between the two word clouds at the first glance. The first one is mainly about Samsung's top products (*galaxy*, *nexus*, *note*, *sii*) and the user experiences in terms of words such as "體驗" (experience), "看到" (see), and "覺得" (feel), while focusing on the multimedia aspect such as "照片" (photo), "拍照" (photograph), and "影片" (movie). On the other hand, the second word cloud is more about seeking help with words "問題" (problem), "解決" (solve), and "無法" (unable), and involves more polite words such as "謝謝" (thanks), "大大" (big brother and big sister), "and "小弟" (I) and technicalities such as rom, "開機" (boot), and "設定" (set up).

Figure 12. Words with the highest weights

Figure 13. Words with the lowest weights

The previous bags-of-word features were based on only contents of the posts. There is also much information lying in the titles of the threads, so we create another 50 dimension-reduced bags-of-

word features based on the titles, and combine these with the content parts to yield 200 features. We prefer not to have them mixed together because title and content may have distinct groups of spam keywords.

M3 in Tables 8 and 9 shows that the addition of title bag-of-words improves F-measure further. The dimension-reduced bags-of-word features turned out to be surprisingly helpful. The model is able to accomplish over 0.55 in F-measure while the ratio of spam is only around 3% on the test sets for first posts. Compared to the random baseline, it boosts the F-measure by as much as 0.45, which implies that the contents of posts actually give some strong clues about whether a first post is spam. Although each spam post looks rather unsuspicious on its own, spam posts put more emphasis on certain topics, in comparison with non-spam posts. Our model trained with bag-of-words features is able to exploit this distinction.

5.1.3 Content Characteristics

A set of features derived from basic characteristics of the contents of the post is introduced in Table 10. In regard to the naming of these features, the *n_* prefix means *number of*, while the *p_* prefix means *proportion of*, i.e., divided by the number of characters in the post. Most features are self-explanatory.

Table 10. Description of content characteristics

Attribute	Description
n_all	number of characters used in the post
n_words	number of words in the post
n_lines	number of lines in the post
n_hyperlinks	number of hyperlinks in the post
n_img	number of images added to the post
n_emoticon	number of emoticons used in the post
n_quote	number of quotations from previous posts
p_digit	proportion of digits
p_english	proportion of English characters
p_punct	proportion of punctuation characters
p_special	proportion of non-alphanumeric characters
p_wspace	proportion of white space characters
p_immediacy	proportion of first person pronouns
p_ntusd_pos	proportion of positive words in *NTUSD*
p_ntusd_neg	proportion of negative words in *NTUSD*
p_emoticon_pos	proportion of positive emoticons
p_emoticon_neg	proportion of negative emoticons

We compute symmetric KL divergence to find out which features exhibit the most different distributions between spams and hams. The formula of symmetric KL divergence is defined as follows.

$$D_{KL}(P_{spam}(f)\|Q_{ham}(f)) + D_{KL}(Q_{ham}(f)\|P_{spam}(f))$$

where P_{spam} and Q_{ham} are the distributions of the features under all spam and all non-spam first posts, respectively. The higher the symmetric KL divergence is, the more different the two distributions are. That makes the features more useful in discriminating between spam and non-spam first posts.

Figure 14 shows symmetric KL divergence between spams and hams. The top four features that distinguish spam and non-spam first posts best are *n_all*, *n_imgs*, *n_words* and *n_lines*, which are all related to the quantity of content. This is not surprising because many of the spam first posts are essentially advertisements in disguise, e.g., unboxing posts and positive experience with *Samsung* product posts, they would generally use lots of words and pictures to showcase *Samsung* products in an attempt to impress people.

On top of the bag-of-words features, we add these 17 numerical features that characterize the contents of the first posts. The resulting performance is shown in M4 in Tables 8 and 9. F-measure increases by about 0.03 on both test set and test set*, so these features do provide extra information that helps detect spam first posts.

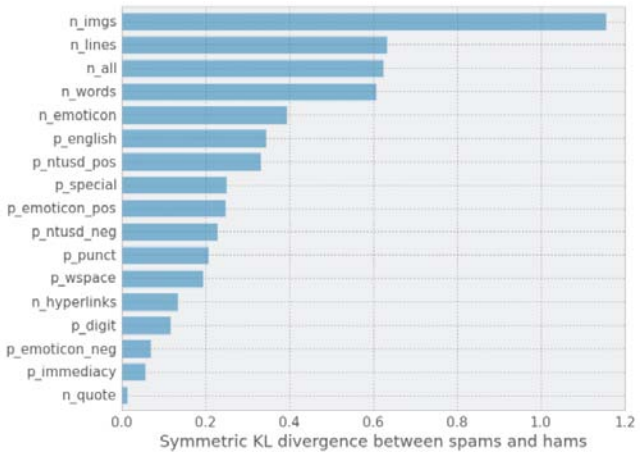

Figure 14. Content characteristics features

5.1.4 Submission Time and Thread Activeness

Besides content-centric features, we also incorporate some non-content-centric features. As discussed in Section 4.5, spam posts have a tendency of being submitted more often during work time. To make use of this observation, we add a binary feature for each hour in a day and each day in a week, in total 24+7=31 new features. If the first post was submitted during the hour or the day a feature corresponds to, then its value is 1; otherwise it is 0.

Moreover, we use number of posts in the thread started by the first post as another feature, which can serve as a measure of the activeness of the thread, as discussed in Section 4.6. Tables 8 and 9 show that incorporating these non-content-centric features (i.e., M5) further improves the F-measure on both test set and test set*.

5.1.5 Sentiment Scores toward the Brands

The main objective of the covert marketing campaign is to promote a certain brand and sometimes denounce its competitor's brands in order to give it an unfair edge. Hence, we expect spam posts to show a positive attitude when it comes to *Samsung*, and possibly a negative attitude toward the competitors. We devise a simple method to capture the sentiment toward brands in posts. Basically, we just add up the polarity of sentiment words in NTU sentiment dictionary (*NTUSD*) [13] and emoticons near mention of a brand or a product.

M6 in Tables 8 and 9 shows that the F-measure of the approach with the sentiment scores toward brands dropped a little on both test set and test set*. The possible reasons why the polarity of our estimated sentiment score fails to reflect the true opinion polarity are listed as follows. First, as discussed in Sections 2.1 and 4.1, the spam posts are carefully written to subtly deliver the messages, so they might avoid using sentiment words to some degree. Second, *NTUSD* has been released for some years, while the community on *Mobile01* may give some words new meanings,

and even invent new words in their subculture. For example, the following post "orz 只能說 S2 真的是怪物" (orz We can only say S2 is really a monster) used negative emoticon such as orz, and word like 怪物 (monster) to compliment a *Samsung* product in a dramatic manner. Third, sarcasm is heavily used on *Mobile01*. The following is a first post in Chinese. The corresponding English translation is also shown. In this example, *Note* and *XL* are products of *Samsung* and *HTC*, respectively. They are competitors in cell phone domains.

我比較推薦 Note
(*I comparatively recommend Note*)
因爲 htc 好像把 XL 當精品賣..哈
(*Because htc seems to sell XL as boutique .. Ha*)
現在單核心的手機還敢賣那麼貴的..而且還很多人讚賞
(*Now single-core phone is sold so expensive bravely.. and a lot of people are appreciated*)
真的只有 HTC
(*Really only HTC*)
XL 真的不錯啦
(*XL is really good*)

5.2 Models for Replies

Following the discussion for first posts, we consider spam detection for replies. In the experimental setup, we remove all the replies in a thread containing any posts (first post and replies) in the training set from the test set in Table 5. Table 11 shows the spam counts and ratios for replies. The ratio of spam posts for replies is even lower than that for first posts. Thus, we randomly remove 90% of the non-spam post for downsampling. Tables 12 and 13 list the experimental results of different models with test set and test set*. The detail will be discussed in the following sections.

Table 11. Training and test set for spam detection for replies

	#spam posts	#all posts	spam ratio
training set	1,337	148,841	0.90%
test set	1,020	67,025	1.52%
test set*	343	25,165	1.36%

Table 12. Spam detection for replies using test set

Model	P	R	F1
M1: random baseline	0.0147	0.4833	0.0285
M2: content bag-of-words	0.1731	0.2451	0.2029
M3: content + title bag-of-words	0.1560	0.2647	0.1963
M4: M3+content characteristics	0.1770	0.2520	0.2079
M5: M4+time+activeness+position	0.1966	0.3098	0.2406
M6: M5+spamicity of the first post	0.2559	0.2961	0.2745

Table 13. Spam detection for replies using test set*

Model	P	R	F1
M1: random baseline	0.0138	0.5015	0.0268
M2: content bag-of-words	0.1185	0.1983	0.1483
M3: content + title bag-of-words	0.1028	0.2216	0.1405
M4: M3+content characteristics	0.1232	0.1983	0.1520
M5: M4+time+activeness+position	0.1465	0.2507	0.1849
M6: M5+spamicity of the first post	0.2110	0.2682	0.2362

5.2.1 Random Baseline

The performance of random baseline for replies, i.e., M1, is worse in comparison with the random baseline for first posts. It reflects the fact that spam ratio is much lower for replies, as observed in Section 4.4.

5.2.2 Bag-of-Words

We repeat the procedure in Section 5.1.2. The optimal number of dimensions to reduce the bag-of-words features is 250, according to the F-measure of 5-fold cross-validation. The evaluation result shown in M2 in Tables 12 and 13 does not looks as nice as bag-of-words for first posts. Besides, the performance on test set is significantly better than that on test set*, so the writing style of the users might partially contribute to the F-measure on test set.

We explain such discrepancy between the performance with bag-of-words features for first posts and for replies as follows. In addition to the fact that spam ratio is lower and content is less for replies, as observed in Section 4.4, many of the spam replies are the vacuous ones with the sole intention of keeping the discussion in the thread alive to attract more attention to the thread, which is probably started by a spam first post, as mentioned in Section 4.1. They are concise and contain little opinion on the brands, so it would be very difficult to distinguish them from non-spam posts. On the other hand, because first posts are the posts that initiate the threads, it cannot be used for such "keeping a thread alive" purpose obviously. In this case, adding title bag-of-words features shown in M3 in Tables 12 and 13 does not help. It is probably due to the fact that title is per thread (also per first post), rather than per reply. A title is shared by all the replies in the thread.

5.2.3 Content Characteristics

As in Section 5.1.3, we compute the symmetric KL divergence of the content characteristics features for replies to find out which of them are the most useful. Figure 15 shows the content characteristics features. Interestingly, replies seem to use more emoticons in general. Spam replies often have either positive or negative attitude explicitly expressed to side with the previous "pro-Samsung" posts (first post in the thread or some previous replies, compared to non-spam replies). Tables 12 and 13 show the performance is slightly improved with the addition of content characteristics features, i.e., M4.

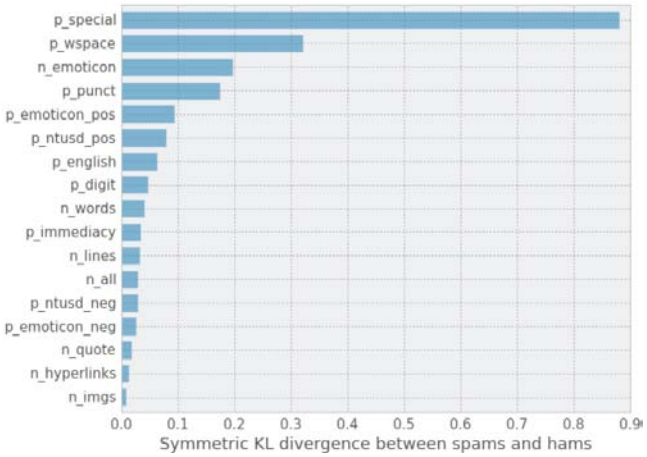

Figure 17. Content characteristics features

5.2.4 Non-Content-Centric Features

As in Section 5.1.4, non-content-centric features indicating the hour/day of the submission time, and the number of posts in the thread in which the reply is, are added. Moreover, the two attributes about the position of a post (reply) in the thread, i.e., *nfloor* and *pnum*, as described in Table 2, are also incorporated as non-content-centric features. M5 in Tables 12 and 13 shows these non-content-centric features are quite helpful in spam detection for replies. F1 is increased 0.0327 and 0.0329 on the test set and test set*, respectively.

5.2.5 Spamicity of the First Post in the Thread

When a thread is started by a spam first post, we could envision more spam activities to follow (as spam replies to the thread), due to the collusion discussed in Section 4.7. To measure the spamicity of first post in the thread in which the reply occur, we leverage our best model for spam detection for first posts, i.e., M5 in Tables 8 and 9. We include its probabilistic prediction on the first post in the thread as an additional feature. M6 in Tables 12 and 13 show its effectiveness. F1 is further increased 0.0339 and 0.0513 on the test set and the test set*, respectively

6. CONCLUSION AND FUTURE WORKS

In this paper, we conduct a real case study of opinion spams with a dataset containing the "true" ground truth. We obtain decent results on opinion spam detection for first posts with features derived from the contents of first posts. Seemingly contradictory to the observation that spam posts are carefully written to avoid getting caught (Section 4.1), our investigation demonstrated that spam first posts tend to put more focus on certain topics that are not that suspicious per se (Section 5.1.2), and we also saw the unusually rich contents of spam first posts could also be a giveaway (Section 5.1.3). Non-content based features such as submission time and thread activeness are useful clues (Section 5.1.4). On the other hand, performance has more room for improvement for spam detection for replies. Spamicity of first post has significant effect on identifying spams from replies.

In our study, the sentiment postulation, i.e., spam posts might show certain attitudes toward their competitors, does not have clear effect. The coverage of sentiment dictionary and the uses of sarcasm are some of major reasons. How to successfully grasp sentiment score toward the brands need to be further investigated.

The interaction between forum posters is also interesting. In comparison to product reviews sites like *Amazon* or *TripAdvisor*, the activities of users on a web forum involve more interactions one another. It may be in an explicit form such as quoting a previous post directly or mentioning a poster's username, or in an implicit form that requires deep natural language understanding. The ways in which spammers interact with each other or with other posters may be worthy of studying to improve the performance of spam detection for replies.

Opinion spammer detection is another interesting task. On the one hand, we can utilize the maximum spamicity of first posts by the poster as a feature, where the spamicity is estimated by the model we devised for spam first post detection. On the other hand, we can also use the spamicity of the posters to help in detection of spam posts. A unified model to integrate spam and spammer detection together can be explored.

7. ACKNOWLEDGMENTS

This research was partially supported by National Taiwan University under grants NTU-ERP-103R890858 and NTU-ERP-104R890858, and Ministry of Science and Technology, Taiwan, under grant MOST 102-2221-E-002-103-MY3.

8. REFERENCES

[1] Enrico Blanzieri and Anton Bryl. A survey of learning-based techniques of email spam filtering. *Artificial Intelligence Review*, 29(1):63–92, 2008.

[2] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[3] Song Feng, Ritwik Banerjee, and Yejin Choi. Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume* 2, pages 171–175. Association for Computational Linguistics, 2012.

[4] Stephanie Gokhman, Jeff Hancock, Poornima Prabhu, Myle Ott, and Claire Cardie. In search of a gold standard in studies of deception. In *Proceedings of the Workshop on Computational Approaches to Deception Detection*, pages 23–30. Association for Computational Linguistics, 2012.

[5] Zoltan Gyongyi and Hector Garcia-Molina. Web spam taxonomy. In *Proceedings of First International Workshop on Adversarial Information Retrieval on the Web* (*AIRWeb* 2005), 2005.

[6] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[7] Christopher G. Harris. Detecting deceptive opinion spam using human computation. In *Proceedings of AAAI Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*, pages 87-93, 2012. AAAI, 2008.

[8] Pedram Hayati, Vidyasagar Potdar, Alex Talevski, Nazanin Firoozeh, Saeed Sarenche, and Elham A Yeganeh. Definition of spam 2.0: New spamming boom. In *Proceedings of 2010 4th IEEE International Conference on Digital Ecosystems and Technologies* (*DEST*), pages 580–584. IEEE, 2010.

[9] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. A practical guide to support vector classification, 2003.

[10] Nitin Jindal and Bing Liu. Opinion spam and analysis. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 219–230. ACM, 2008.

[11] Nitin Jindal, Bing Liu, and Ee-Peng Lim. Finding unusual review patterns using unexpected rules. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1549–1552. ACM, 2010.

[12] Thorsten Joachims. A support vector method for multivariate performance measures. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 377–384. ACM, 2005.

[13] Lun-Wei Ku, Hsiu-Wei Ho and Hsin-Hsi Chen. Opinion mining and relationship discovery using CopeOpi opinion analysis system. *Journal of the American Society for Information Science and Technology*, 60(7):1486–1503, 2009.

[14] Ee-Peng Lim, Viet-An Nguyen, Nitin Jindal, Bing Liu, and Hady Wirawan Lauw. Detecting product review spammers using rating behaviors. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 939–948. ACM, 2010.

[15] M McCord and M Chuah. Spam detection on twitter using traditional classifiers. In *Proceedings of the 8th International Conference on Autonomic and Trusted Computing* (*ATC* 2011), pages 175–186. Springer, 2011.

[16] Arjun Mukherjee, Bing Liu, Junhui Wang, Natalie Glance, and Nitin Jindal. Detecting group review spam. In *Proceedings of the 20th International Conference on World Wide Web*, pages 93–94. ACM, 2011.

[17] Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T Hancock. Finding deceptive opinion spam by any stretch of the imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 309–319. Association for Computational Linguistics, 2011.

[18] Fabian Pedregosa, Ga¨el Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikitlearn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[19] James W Pennebaker, Cindy K Chung, Molly Ireland, Amy Gonzales, and Roger J Booth. The development and psychometric properties of liwc2007. Austin, TX, LIWC. Net, 2007.

[20] Aldert Vrij, Ronald Fisher, Samantha Mann, and Sharon Leal. A cognitive load approach to lie detection. *Journal of Investigative Psychology and Offender Profiling*, 5(1-2):39–43, 2008.

[21] Guan Wang, Sihong Xie, Bing Liu, and Philip S Yu. Review graph based online store review spammer detection. In *Proceedings of 2011 IEEE 11th International Conference on Data Mining* (*ICDM*), pages 1242–1247. IEEE, 2011.

[22] Sihong Xie, Guan Wang, Shuyang Lin, and Philip S Yu. Review spam detection via temporal pattern discovery. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 823–831. ACM, 2012.