PACIS 2014 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

2014

DISCRIMINATIVE TOPIC MINING FOR SOCIAL SPAM DETECTION

Long SONG
City University of Hong Kong, song.long@my.cityu.edu.hk

Raymond Y.K. Lau
City University of Hong Kong, raylau@cityu.edu.hk

ChunXiao Yin
Information Systems, USTC-CityU Joint Advanced Research Center, Suzhou, Jiangsu, China., yincx@mail.ustc.edu.cn

Recommended Citation

SONG, Long; Lau, Raymond Y.K.; and Yin, ChunXiao, "DISCRIMINATIVE TOPIC MINING FOR SOCIAL SPAM DETECTION" (2014). *PACIS 2014 Proceedings*. 378.
http://aisel.aisnet.org/pacis2014/378

DISCRIMINATIVE TOPIC MINING FOR SOCIAL SPAM DETECTION

Long SONG, Department of Information Systems, City University of Hong Kong, Hong Kong, song.long@my.cityu.edu.hk

Raymond Y.K. Lau, Department of Information Systems, City University of Hong Kong, Hong Kong, raylau@cityu.edu.hk

Chunxiao YIN, USTC-CityU Joint Advanced Research Center, yincx@mail.ustc.edu.cn

Abstract

In the era of Social Web, there has been an explosive growth of user-contributed comments posted to various online social media. However, increasingly more misleading and deceptive user comments found at online social media have also been a great concern for consumers and merchants, and social spam have been brought to the attention by the legal circle in recent years. Social spam can cause tremendous loss to both consumers and merchants, and so there is a pressing need to design effective methodologies to detect social spam to maintain the hygiene of online social media. The main contribution of this paper is the illustration of a novel social spam detection methodology which combines word-, topic-, and user-based features to combat social spam. In particular, the proposed methodology is underpinned by the Labeled Latent Dirichlet Allocation (L-LDA) model, a kind of probabilistic generative model. A series of experiments conducted based on the social comments posted to YouTube show that our proposed methodology can achieve a detection accuracy of 91.17%. The business implication of our research is that merchants can apply our methodology to filter spam so as to extract accurate market intelligence from online social media. Moreover, social media site owners can leverage the proposed methodology to maintain the hygiene of their sites.

Keywords: Social Spam, Social Media, Topic Model, Labeled LDA, Support Vector Machine, Latent Semantics, Machine Learning, Spam Detection.

1 INTRODUCTION

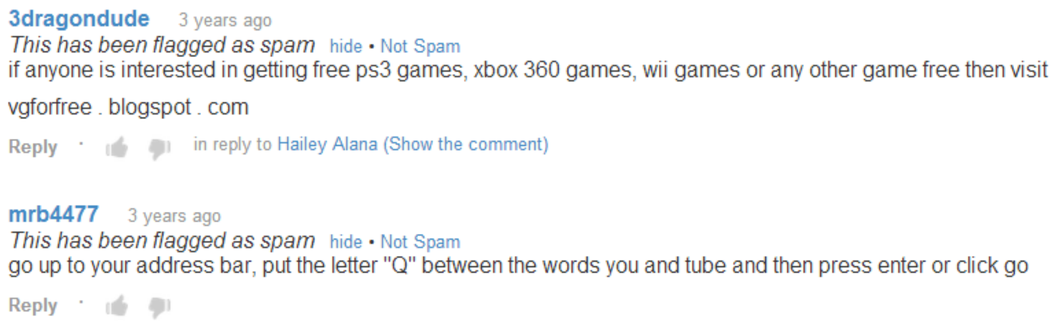
Recently, social media has been playing an increasingly more important role in people's daily life (van Marle 2011). Andreas Kaplan and Michael Haenlein (2010) give the definition of social media as "a group of Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of user-generated content." Besides, the features of social media are mainly summarized as their support for collective action and social interaction, their grassroots nature and decentralized governance, and the flexibility and portability of their technological platforms (Nevo, Benbasat and Wand 2012). The attributes of social media, including its creation and exchange of user-generated contents, and its support for highly collective actions and diverse social interactions, definitely make it an indispensable tool for retailers in e-commerce to promote their products and services. Furthermore, social media may become a new "social CRM" tool if retailers choose to import all the user profiles into their social media sites and keep close contacts with users on the corresponding platforms. The success of Starbucks is a good example. Starbucks always has a good insight of social media and first jumps into the social media stage by creating its own version of social network—My Starbucks Idea[1], on which the customers can share and discuss anything related to Starbucks. In 2010, Starbucks already had over 705,000 followers at Twitter and over 5,428,000 fans at Facebook.

Unfortunately, the power of social media also provides unprecedented opportunities for online fraudsters or spammers, who take advantage of social media platforms to perform deceptive acts (Chandramouli 2011), conduct unfair trading activities (Wu et al. 2010, Yoo and Gretzel 2009), and even make illegal profits (Toneguzzi 2007). The most typical type of deceptive behavior in social media is posting social spam. Social spam is some information of low quality that users do not ask for or specifically subscribe to in social networks (Wang, Irani and Pu 2011). It is usually used to make phishing attacks (Jagatic et al. 2007), promote adverse websites (Lin et al. 2007), distribute malwares (Boyd and Heer 2006), and spread adverse messages (Brown et al. 2008, Zinman and Donath 2007). Social spam can highly mislead online users since it contains contextual information (Brown et al. 2008, Felt and Evans 2008). For social spam, some URLs are often included which will direct users to some advertisement websites, malware websites, or pornographic websites. Obviously, the damages of social spam are various. On one hand, the information embedded in a spam is totally irrelevant and trashy to those users who intend to locate useful opinion or information in social media. In the long run, users' patience and satisfaction will be compromised gradually.

Figure 1 shows an example of social spam for the videos of Starbucks archived at YouTube. In this example, we can see the spam definitely affect the company's normal promotion and common communication with its customers. If the spam appears in a large amount, it will hinder Starbucks' managers to extract customer opinions from online social media to develop accurate market intelligence. On the other hand, from the perspective of online social media sites, social spam consumes system resources such as the bandwidth and disk space. Besides, if a user happens to click the links embedded in social spam, it will direct them to other dangerous websites which may result in personal information leakage and even huge monetary losses. According to Grier et al. (2010), 8% of 25 million URLs posted at Twitter pointing to phishing, malware and scams listed in popular blacklists. Yet, Twitter is such a successful website that attracts billions of users to view spam pages with a click rate of 0.13%. According to the report from the FBI and the American National White Collar Crime Center, the monetary losses caused by scam websites reached $240 million in 2008. Therefore, there is a pressing need to develop effective methodologies to fight against various social spam in order to reduce consumers' losses and create a clean environment for all online users and merchants. Moreover, since business managers and marketers have paid increasingly more attention to leverage online social

[1] http://mystarbucksidea.force.com/apex/ideaHome

media for e-Marketing and business transformation, it is essential to filter out social spam before accurate market and business intelligence can be extracted from online social media.



3dragondude 3 years ago
This has been flagged as spam
if anyone is interested in getting free ps3 games, xbox 360 games, wii games or any other game free then visit
vgforfree . blogspot . com
in reply to Hailey Alana (Show the comment)

mrb4477 3 years ago
This has been flagged as spam
go up to your address bar, put the letter "Q" between the words you and tube and then press enter or click go

Figure 1. A Snapshot of Social Spam about Starbucks found at YouTube[2]

In this paper, we design a novel computational methodology that combines word-, topic-, and user-based features for automated detection of social spam. In particular, our computational methodology is underpinned by Labeled Latent Dirichlet Allocation (Ramage et al. 2009) that can mine latent topics characterizing the inherent semantics of social spam. Labeled Latent Dirichlet Allocation (L-LDA) is a supervised variant of Latent Dirichlet Allocation (LDA), a kind of topic modeling method which was first proposed in Blei et al.'s paper (2003). The assumption of topic models is that a document is a mixture of various topics and each topic can be regarded as a group of words that have a high probability of co-occurrence. By running the topic models, we can automatically extract latent topics in the documents. Furthermore, these latent topics can be transformed as topic-based features for classification which are quite different from the word-based features like *tf-idf*. By adopting L-LDA model instead of LDA model, we can identify those discriminative topics in the documents as long as topic labelling can be performed. In this paper, in order to extract topic-based features we use the occurrences of the most discriminative words generated from Chi-Square test as topic labels for each document. Then, a comparison between three kinds of features—word-, topic-, and user-based features is made through rigorous empirical experiments to examine the effectiveness of these features toward social spam detection.

In summary, we frame our contributions as follows:

- We propose a novel method to extract topic-based features from the comments in social media by adopting L-LDA model.
- We prove that the topic-based features extracted by using our method can largely improve the performance of social spam classifiers by comparing with word- and user-based features.
- We also examine the effectiveness of word- and user-based features like *tf-idf* weights, average time interval of posting (ATI) and the average similarity (AS) for social spam detection.

We organize the remainder of the paper as follows. Section 2 presents some related studies about social spam detection. Section 3 illustrates the proposed computational methodology for social spam detection. Section 4 describes our experiments that were conducted based on a YouTube social spam dataset. Section 5 discusses our experimental results, followed by conclusions and future research in Section 6.

[1] http://www.youtube.com/all_comments?v=lMmA5dm-Q_w

2 RELATED WORKS

Of all the methods to tackle the social spam problem, the classification-based method is the most common one. The general steps for this method is to define and extract features from the spam, use the extracted features to train the classifier, and finally do the classification via the trained classifier. Usually, the features vary in the different research articles. In Markines et al.'s work (Markines, Cattuto and Menczer 2009), six features which have precisely captured the properties of the social spam are proposed, and a 98% accuracy rate and a near 2% false positive rate are achieved via an AdaBoost classifier. The six features include Tagspam (detect some special tag), TagBlur (capture the degree of independence of tags in a post), DomFp (compute the similarity of the source webpage and the existed spam pages), NumAds (compute the number of Ads in a source page), Plagiarism (detect automatically generated pages), ValidLinks (compute the percentage of valid resources post by a user). In another paper (Lee, Caverlee and Webb 2010), the authors deploy honeypots to monitor spammers' behavior and log their information. Multiple features like tweets similarity, material status, number of friends are extracted from the spam files harvested by the honeypots and models like the bag-of-words model and the sparse bigrams model are established to characterize the text-based features.

For the articles discussed above, the context is limited to social networks like Twitter, MySpace, etc., however, social spam also exist in the video websites like YouTube (Benevenuto et al. 2009). As a video website like YouTube, it's a good platform to put some product ads and promote them in order to increase the sales of the products. When video providers want to promote the video which means improving the rank of the video to attract more attention, they are most likely to generate a lot of irrelevant comments under the videos. In Benevenuto et al.'s work, their aim is to detect spammers who may post unrelated things to a popular video, and content promoters who post a large number of responses in order to gain visibility to a specific video, in online social networks like YouTube. Benevenuto et al. built a dataset using the real data of YouTube users and extracted 42 attributes from three attribute sets including video attributes, user attributes and social network attributes to discriminate spammers and promoters from legitimates. By using Information gain and χ^2 test to assess their discriminating power, they find that total number of views is the most discriminative feature. Besides, another paper by Sureka (2011) also dealt with the YouTube spam comments, and by mining user's comment activity logs and extracting patterns an effective method for spam detection was proposed. In O'Callaghan et al.'s paper (2012b), network motif profiling was adopted to track the spam campaigns over time on YouTube. This method successfully revealed two distinctive spam campaign strategies and tracked two corresponding active campaigns on the evaluation dataset.

As topic models, especially the variants of Latent Dirichlet Allocation (LDA) model, become more and more popular in different area including information retrieval (Brody and Elhadad 2010), genetics (Liu et al. 2010) and image classification (Wang, Blei and Li 2009), etc., there is an increasing number of papers adopting LDA to perform spam detection and opinion mining. In Bíró et al.'s paper (2008), a modification of LDA, the novel multi-corpus LDA was applied to web spam classification. Bíró et al. assumes that spam contents and legitimate contents (i.e., ham) have different number of topics and trained the spam topics and ham topics separately. In Bíró et al.'s another paper (2009), another extension of LDA is developed in which the linkage information between the documents is considered while training. What's more, LDA can be used to track the trend of some online opinion or extract some opinion from the documents. In Yan and Zheng's paper (2010), they used incremental Gibbs algorithm train the LDA model and the previous posterior of topic-word distribution was introduced to train the LDA model in the next time-slice. In this way, the trends of some topic can be tracked and observed.

However, as far as I know, none of the prior researchers have used the-state-of-the-art topic models to generate topic-based features to identify social spam and the current features adopted in the previous mentioned papers only include user-based, word-based, or network-based features. In our paper, we combine the word, topic, and user-based features together and incorporate the-state-of-the-art Labeled

LDA (L-LDA)—a variant of LDA—with the classical classification algorithm Support Vector Machine (SVM) to test the performance of the combinations of the three kinds of features.

3 THE PROPOSED METHODOLOGY

In this section, we will begin with our proposed methodology for social spam detection shown in Figure 2. From our proposed methodology, three kinds of features are generated in our paper—word-, topic-, and user-based features. For word-based features, *tf-idf* scheme and Chi-Square test were adopted for feature extraction and selection. Besides, the discriminative words with high Chi-Square scores were treated as the topic labels which were a part of the input of L-LDA model. The function of topic model—L-LDA is to identify what kind of topics one certain comment talks about. Then the normalized topic frequencies are generated as our topic-based features. For user-based features, user behavior analysis was performed for each user by computing his/her average time interval of posting (ATI) and the average similarity (AS) of two adjacent comments. Finally, SVM classifiers were adopted to perform the classification for spam and ham.

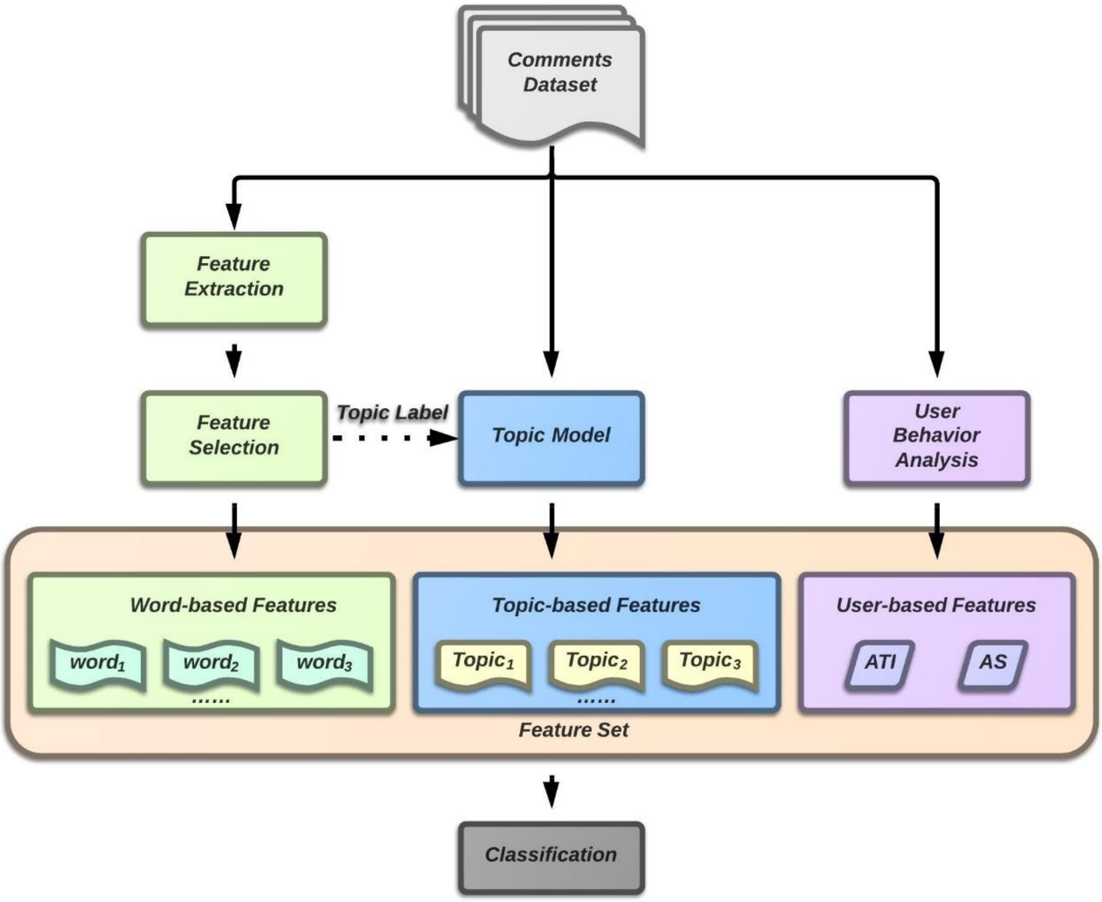


Figure 2. Proposed Methodology for Social Spam Detection

3.1 Word-Based Features

3.1.1 Feature Extraction

As *tf-idf* scheme (Salton and McGill 1986) offers a brief representation of the documents by computing the weights of the words in the documents, but also the *tf-idf* word-based features turn to be discriminative and powerful in various classifiers like KNN, SVM, and Rocchio (Joachims 1996, Soucy and Mineau 2005). Therefore in this paper for the word-level features, we decide to use *tf-idf* weights as the base of our entire feature set. Here, we perform a maximum normalization in the computation of term frequency, which turns out to be a very effect variant of *tf* functions. The normalized *tf* value of word *i* in document *d* is computed by

$$ntf_{i,d} = \alpha + (1 - \alpha)\frac{tf_{i,d}}{\max_i tf_{i,d}} \tag{1}$$

where α is a value between 0 and 1. Thus, the weight of word *i* in document *d* is denoted by the *tf-idf* scheme as follows.

$$w_{i,d} = ntf_{i,d} \times \log\frac{N}{df_i} \tag{2}$$

Here, *N* is the total number of documents in a collection and df_i is the document frequency of word *i*.

3.1.2 Feature Selection

However, although *tf-idf* scheme is adopted to give an effect representation of the documents, the size of the vocabulary of the corpus is usually large, which usually makes the vector sparse. From the view of text classification or clustering, these sparse vectors will make the task not that efficient and meanwhile the potential noisy features may decrease the accuracy of classification directly. Thus, to perform the classification or clustering on the corpus, feature selection methods are necessary to be used.

To select the features, we adopted Chi-Square test to perform feature selection in the word set. Chi-Square test is one of the most popular metrics for feature selection in text classification area (Alexandrov, Gelbukh and Lozovoi 2001, Dunham and Ming 2003, Forman 2003). Chi-Square test can be used as “(a) a goodness-of-fit test between a group of data and a specific probability distribution, or (b) a test for the degree of dependence or association between two factors or variables” (Al-Harbi et al. 2008). The Chi-Square test formula is relevant to feature selection functions in information-theoretic area which “try to capture the intuition that the best terms for the class *c* are the ones distributed most differently in the sets of positive and negative examples of *c*” (Mesleh , Sebastiani 2002). The basic formula for Chi-Square test is as follows.

$$\chi^2 = \sum\frac{(O - E)^2}{E} \tag{3}$$

In the formula above, the *O* represents the observed values which are instantiated as $|D_{i,positive}|$, $|D_{i,negative}|$, $|D_{positive}|$ - $|D_{i,positive}|$, $|D_{negative}|$ - $|D_{i,negative}|$ in our experiment. *D* is our whole corpus set, $D_{positive}$ is the spam set and $D_{negative}$ is the ham set. For each word *i*, $|D_{i,positive}|$ means how many times the word *i* appears in spam set and similarly $|D_{i,negative}|$ means the times that word *i* appears in ham set; Also, $|D_{positive}|$ and $|D_{negative}|$ are the total number of spam and ham. And *E* is the expected value of the observed values. In order to illustrate this in details, we give a contingency table as follows.

	Spam	Ham	total
Contain word i	$\lvert D_{i,positive}\rvert$ = A	$\lvert D_{i,negative}\rvert$ = B	A + B
Do not contain word i	C	D	C + D
total	$\lvert D_{positive}\rvert$ = A + C	$\lvert D_{negative}\rvert$ = B + D	N

Table 1. A Contingency Table

With the values given above, we could compute the expected value of *A* to *D*. For example, for *A* its expected value is

$$E_A = (A + C) \times \frac{A + B}{N} \tag{4}$$

For *B*, *C* and *D*, the computation is much similar. After putting all the values into (3), finally we can get the Chi-Square score χ^2 as follows.

$$\chi_i^2 = \frac{N(AD - BC)^2}{(A + C)(A + B)(C + D)(B + D)} \tag{5}$$

The value of χ_i^2 represents the association or dependency of word *i* with the spam set. The higher the value is, the stronger the discrimination power of the word *i* is. In the case that the size of the vocabulary of the corpus is very large, we will select the most distinctive words based on the Chi-Square score as our baseline feature set.

3.2 Topic-Based Features

Indeed, *tf-idf* has a few attractive advantages and the feature selection methods could help fix one of its shortcomings that the representation length is a little long for each document, but another drawback of this method still exists. That is, it fails to reveal the intra-document statistical structure and by adopting the “bag-of-words” assumption the documents are just viewed and decomposed from the micro word level, which may lead to the loss of semantic information among the words. To address this problem, a more macro level of viewing the documents is needed.

According to systemic functional linguistic theory (SFLT) which provides a mechanism to represent text information, language has three meta-functions—ideational, interpersonal, and textual (Halliday and Matthiessen 2004). For ideational meta-function, it provides a theory of human experience and is about the aspects of the “mental word” including attitudes, desires, etc. (Fairclough 2003). The ideational meta-function can be performed as several information types such as topics, emotions, opinions, etc. (Abbasi and Chen 2008). If the representation of text information can reach this level, semantic information among the words can be retained in a large extent. A significant step forward in this direction was made by Blei (2003), who proposed one of the earliest probabilistic topic models—Latent Dirichlet Allocation (LDA), which can identify how certain topic patterns are mixed in a document. LDA assumes the whole corpus has *k* number of topics, and the content of each document is focusing on these *k* topics. The document is considered as a mixture of topics with different probabilities for each and for each topic there exists a distribution for all the words.

Later on, Labeled LDA (L-LDA) (Ramage et al. 2009), a supervised variant of LDA was proposed. L-LDA is also a probabilistic graphical model, but different from LDA, L-LDA can automatically learn the latent topics in a document from the training set based on the given topic label and then predict the occurrences of the defined topics in a previously unseen document in the test set. Assume we estimate that there are *k* topics in total in this corpus; for each document *d*, its topic label $\Lambda_d = (l_1, l_2, \ldots, l_k)$ will be generated in some way. Here each $l_k \in \{0,1\}$ represents whether this document is related to the kth topic or not. By changing the value of the topic label for each document, we can alter the relevance between the document and the topics we are concerned with. Thus, the topics obtained from running L-LDA are definitely those we are interested in. To help interpret the meaning of topic label, we can compare the topic label to a few centroids and the L-LDA model will automatically cluster the words around these centroids to form the interesting topics. Figure 3 illustrates the probabilistic graph of L-LDA model.

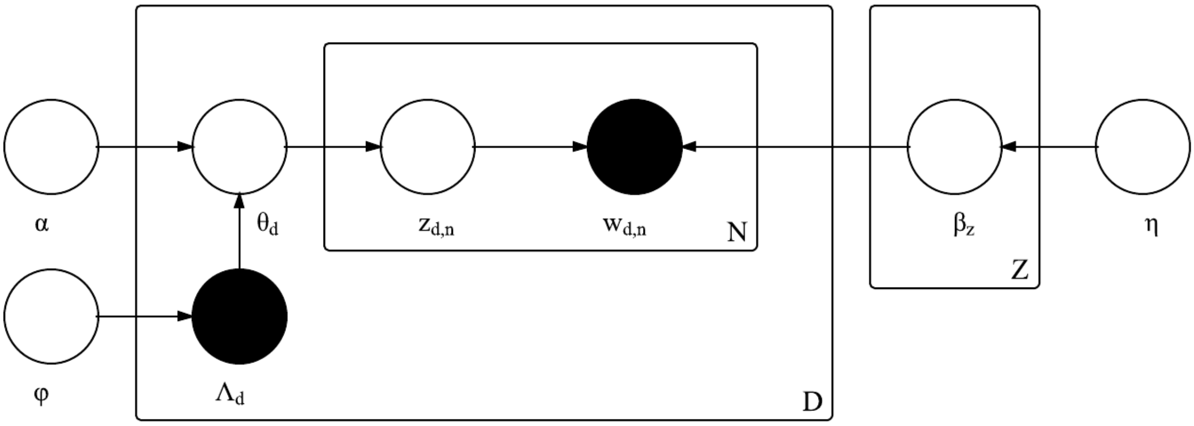


Figure 3. Graphical Model of L-LDA (Ramage et al. 2009)

For L-LDA model, the document is considered as a mixture of topics with different probabilities for each and for each topic there exists a distribution for all the words. Thus there are two distributions—document-topic and topic-word distributions. And these two distributions are all assumed to have a Dirichlet prior. For every document $d \in D$, the distribution θ_d on topic set Z is sampled from Dir(α). And for each topic $z \in Z$, the distribution β_z on vocabulary set V is sampled from Dir(η). Therefore, for the n^th word in document d represented as $w_{d,n}$, a topic assignment $z_{d,n}$ will be iteratively calculated via Gibbs-sampling (Casella and George 1992). After getting the topic assignment z, the document-topic distribution $\theta_{d,z}$ and topic-word $\beta_{z,w}$ distribution can be estimated as follows.

$$\theta_{d,z} = \frac{N_{d,z} + \alpha}{\sum_{z=1}^{|Z|} N_{d,z} + |Z|\alpha} \tag{6}$$

$$\beta_{z,w} = \frac{N_{z,w} + \beta}{\sum_{w=1}^{|V|} N_{z,w} + |V|\beta} \tag{7}$$

Here, $N_{d,z}$ represents the number of words assigned to topic z in the document d and $N_{z,w}$ represents the frequency of word w assigned to topic z in the corpus.

However, how to generate a topic label Λ_d for each document and how to define the topics that we are concerned with still remain unsolved. In our opinion, those topics we are concerned with should be those that can differentiate spam from ham in our corpus. That is, these topics must have a certain level of discrimination power. In 3.1.2, we mentioned the Chi-Square score χ^2 could be treated as a measure of discrimination power of a word. Thus, the topic label of a document can be generated based on the occurrences of the most discriminative words in that document. That is, for a document d, each l_n in its topic label $\Lambda_d = (l_1, l_2, \ldots, l_k)$ can be

$$l_n = \begin{cases} 1, & \forall \mathrm{i} \in \mathrm{d}, \mathrm{n} = \mathrm{Rank}(\chi_i^2) \leq \mathrm{k} \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

where Rank(χ_i^2) means the rank of the Chi-Square score of word i in document d. One advantage for this practice is that since the higher-ranked words are the most discriminative ones thus we can expect that the latent topics generated by the L-LDA also tend to be discriminative for spam detection. It is worth noting that not all the selected words are discriminative for spam and some are discriminative for ham. But once the learned latent topics are transformed as topic-based features, no matter their labels are discriminative for spam or ham, the generated topic-based features are powerful for spam detection.

By regarding the discriminative words as the topic labels for each document, a multi-labelled corpus can be successfully generated. And this multi-labelled corpus can be the input of L-LDA model. After running the L-LDA model, it will output the topic assignment $z_{d,n}$ for each word *i* in document *d* and further get $N_{d,z}$ –the number of words assigned to topic *z* in the document *d*. Here, for a topic *z*, its frequency in a document *d* can be represented by $N_{d,z}$. Similarly as 3.1, here we selected the normalized topic frequency as the feature for spam detection, and for document *d* the normalized frequency of topic *z* can be computed as

$$t_{d,z} = \tau + (1 - \tau) \frac{\mathrm{N}_{d,z}}{\max\limits_{z \in Z} \mathrm{N}_{d,z}} \tag{9}$$

where τ is a value between 0 and 1. Finally, for a document *d*, its topic-based feature is $T_d = [t_{d,1}, t_{d,2}, \ldots, t_{d,k}]$.

3.3 User-Based Features

What's more, recently more and more researchers found that the online spammers turn to generate spam at a short time interval and the content of these spam tend to be the same (Duan, Gopalan and Yuan 2007, Sureka 2011). This is largely because most spam messages are generated by bot accounts which automatically keep posting the same content. In order to detect this kind of spam comments, we added average time interval of posting (ATI) and the average similarity (AS) of two adjacent comments posted by the same user to our feature set, since for the bot accounts they usually repeat posting the comments of the same content continuously. For user *u* who posted N_u comments, the ATI of user *u* is computed as

$$\mathrm{ATI(u)} = \frac{\sum_{k=1}^{N_u} (t_{u,k} - t_{u,k-1})}{N_u} \tag{10}$$

where $t_{u,k}$ represents the posting time of the k^{th} comment of user *u*. Similarly, the AS of user *u* is as

$$\mathrm{AS(u)} = \frac{\sum_{k=1}^{N_u} sim(c_{u,k}\,, c_{u,k-1})}{N_u} \tag{11}$$

where $c_{u,k}$ is the k^{th} comment of user u and function *sim()* is a similarity function computing the similarity between two comments. For computing the similarity of two comments, we utilize the approximate string matching algorithm which roughly works by looking for the smallest number of edits to change one string into the other (Myers 1986, Ukkonen 1985).

4 EXPERIMENTAL EVALUATION

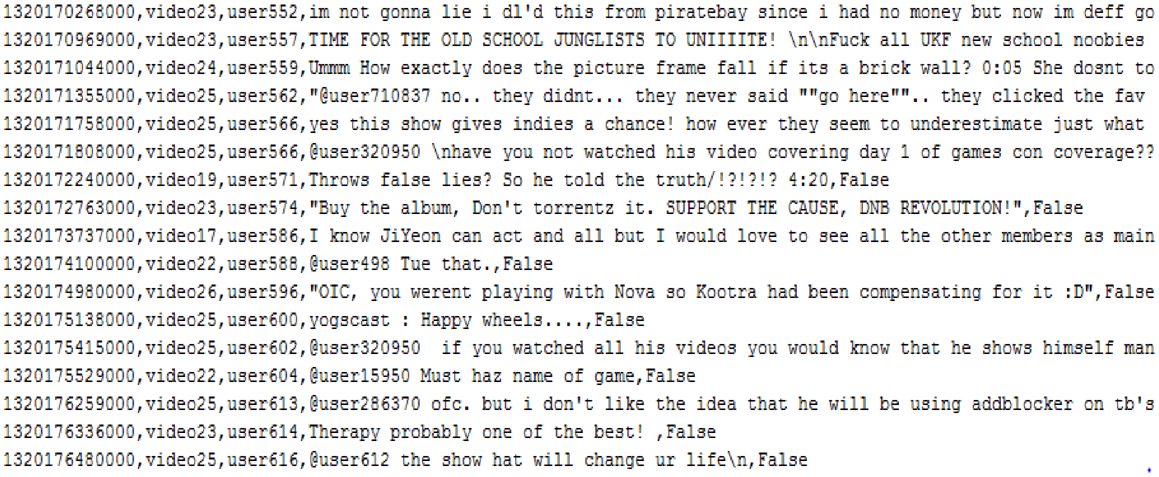
4.1 Dataset

This dataset used for experimental evaluation contains millions of YouTube comments, and these comments are labeled as true spam or not either by the spam filter or manually with the "Flag for spam" button available above each comment posted on a video page on YouTube. Based on these labeled comments, we elaborately extract the features mentioned above to classify the comments into spam or ham by utilizing a Support Vector Machine (SVM) classifier. The data collection began on October 31st, 2011, and we downloaded the dataset from the data source mentioned in the papers (O'Callaghan et al. 2012a, O'Callaghan et al. 2012b). Some statistics of the dataset are shown in Table 2.

Properties	Values
Videos	6,407
Total comments	6,431,471
Comments marked as spam	481,334
Total users	2,860,264
Spam comment users	177,542

Table 2. Some Properties of the Dataset

The most interesting feature of this dataset is its spam label generated by the spam filter or manually with the "Flag for spam" button available above each comment posted on a video page on YouTube. What we want to clarify is that these labels may not be 100% accurate. Due to its occasional inaccuracy, sometimes the innocent comments will be marked as spam while the true spam ones escape from the spam filter. And similarly, some spam comments are not tagged as spam due to infeasibility of manual verification of large volumes of comments. In the paper by Sureka (2011), this problem of the inaccurate labels also appeared. In another paper by Benevenuto et al. (2009), in order to avoid this situation manual annotation was performed before the YouTube data was put into the classifier. And most importantly, the spam tag is usually initialized by other users based only on the content-level instead of the user-label. That's why in our experiment we introduced various types of features to detect the spam automatically. At this stage, we adopt the approximate annotation of spam as our training label; in the future, we may consider improve the accuracy of the spam annotation of the YouTube comments.



```
1320170268000,video23,user552,im not gonna lie i dl'd this from piratebay since i had no money but now im deff go
1320170969000,video23,user557,TIME FOR THE OLD SCHOOL JUNGLISTS TO UNIIIITE! \n\nFuck all UKF new school noobies
1320171044000,video24,user559,Ummm How exactly does the picture frame fall if its a brick wall? 0:05 She dosnt to
1320171355000,video25,user562,"@user710837 no.. they didnt... they never said ""go here"".. they clicked the fav
1320171758000,video25,user566,yes this show gives indies a chance! how ever they seem to underestimate just what
1320171808000,video25,user566,@user320950 \nhave you not watched his video covering day 1 of games con coverage??
1320172240000,video19,user571,Throws false lies? So he told the truth/!?!?!? 4:20,False
1320172763000,video23,user574,"Buy the album, Don't torrentz it. SUPPORT THE CAUSE, DNB REVOLUTION!",False
1320173737000,video17,user586,I know JiYeon can act and all but I would love to see all the other members as main
1320174100000,video22,user588,@user498 Tue that.,False
1320174980000,video26,user596,"OIC, you werent playing with Nova so Kootra had been compensating for it :D",False
1320175138000,video25,user600,yogscast : Happy wheels....,False
1320175415000,video25,user602,@user320950  if you watched all his videos you would know that he shows himself man
1320175529000,video22,user604,@user15950 Must haz name of game,False
1320176259000,video25,user613,@user286370 ofc. but i don't like the idea that he will be using addblocker on tb's
1320176336000,video23,user614,Therapy probably one of the best! ,False
1320176480000,video25,user616,@user612 the show hat will change ur life\n,False
```

Figure 4. A Screenshot of the Dataset Content

Figure 4 shows the content of the dataset. From the first column to the last, the items are the time, the video number, the user number, the comment, and the label.

4.2 Experimental Design

4.2.1 Pre-processing

In our experiment, we mainly focus on the English spam comments detection. Therefore, the first step the pre-processing stage is to filter those non-English comments. After finishing filtering those non-English comments, the total number of the comments is 3,621,379 including true spam comments 392,964 and ham comments 3,228,415.

In the following, a part of users' comments were removed from the dataset because the total number of comments they post is smaller than a threshold. Here, we set the threshold as 6, that is, if a user just posted fewer than 6 comments in our dataset, then all his comments would not be considered in our experiment. The reason why we filtered these comments is that the features we extracted involve the comparison of two adjacent comments posted by the same user. To some degree, these features reflect the user's long term activity. In that way, if the user posted too few comments, the discrimination power of this feature will be weakened. Since we adopted machine learning method to classify the spam, a separation of training set and test set should be done. In our experiment, we separated the two sets based on the time sequence of each user posting the comments, which means we extracted first 2/3 of each user's comments as the training set and the remaining 1/3 as the test set based on the time stamp on each comment. After these two stages of pre-processing, a contrast of the original dataset information and the filtered dataset information is shown in Table 3.

Dataset	**Original**	**Filtered**	**Training set**	**Test set**
Total comments	6,431,471	1,055,375	724,569	330,806
Comments marked as spam	481,334	210,283	142,965	67,318
Comments marked as ham	5,951,037	845,092	581,604	263,488

Table 3. A Contrast of the Original Dataset and the Filtered Dataset

Before generating the feature set, tokenization and stemming should also be performed. Stemming is a very common technique in information retrieval area to eliminate simple variations of some words. In linguistic morphology, stemming is the process for reducing inflected (or sometimes derived) words to their stem, base or root form—generally a written word form and the efficiency of content-based spam filter can be significant improved if adopting stemming (Ahmed and Mithun 2004). The stemming process can effectively rule out the influence of the different forms of the same word when we are performing word statistics.

4.2.2 Feature Extraction and Selection

According to our statistics after performing Chi-Square test the cardinality of the word set in the dataset is 211,278. And most of them are misspelled words and appear less than 10 times in one million pieces of comments. Besides, among all the words, only a small number of them are discriminative and the rest have little contribution to the spam comments detection. Therefore, a minimum frequency for each word is set and if the frequency of one word is fewer than the minimum frequency it will be removed from the vocabulary. For our convenience, we set the minimum frequency to 50. In the end, the size of our vocabulary decreased to 7921. That means the length of the word-based feature equal to 7921. Furthermore, if the number of the latent topics to be learned in L-LDA model is set to *k*, plus the user-based feature ATI and AS, in total we have *k*+7923 features in our feature set.

4.2.3 Classification

Considering the efficiency and convenience, we chose Liblinear package—an implementation of support vector machines—to classify the comments. Liblinear is a very popular classifier which can classify a large dataset quickly with a high accuracy by using the linear kernel. And the objective function with L2-regularized L2-loss form was selected in our experiment.

To better evaluate the performance of each type of feature, four different combinations of features were created—(1) only word feature (W), (2) word+user features (WU), (3) word+topic features (WT), (4) word+topic+user features (WTU). From the experimental results, we were expecting to see how the performance would be improved after adding certain features. Besides, not only the combinations of features affect the results, the number of latent topics to be learned will also have an influence on the results. Thus, to run the experiment we set the number of latent topics *k* = 6, 10, 20, 50, 80, 100. Finally except W and WU group, there were 2×6 experiment groups and 12 SVM classifiers were trained and tested.

4.2.4 Evaluation Metrics

The evaluation metrics involved in our experiment include precision (PRE), accuracy (ACC), recall (REC), F_1-measure (F1), receiver operating characteristic curve (ROC) together with the area under the curve (AUC). These are standard metrics to evaluate the performance of such models.

5 EXPERIMENT RESULTS AND ANALYSIS

After training and testing for 14 groups of SVM classifiers, the overall performance of each group is shown in Table 4. As mentioned before, five basic evaluation metrics including accuracy, precision, recall, F_1-measure, AUC value are applied here. Taking a rough glance, the when topic quantity k = 10 or k = 20, the results tend to be better. And in terms of accuracy and precision, the groups containing topic-based features will outperform those only with word and user-based features. But for recall the situation is different, those groups without topic-based features tend to capture more spam and achieve higher recall than the other groups and the top recall value 84.20% is obtained in group with both word and user-based features. This is reasonable because for a topic model when it generates the topic-based features for a comment assigning topics for each word is based on the current topic assignment of all the other words in the same comment. Assuming that in this comment there are a lot of previously unseen words and only a minority of learned words, then the topic inference for these unknown words will be inaccurate, and as the majority of the words in the vocabulary tend to be ham discriminative and appear more in the normal comments these previously unknown words are inclined to be assigned a ham topic. Therefore, less spam will be captured in the groups with topic-based features and recall certainly decreases.

However, generally the groups with topic-based features usually obtain a higher precision. This is because topic-based features reveals how the latent sematic structure in the comment and this kind of structure judges the spam at a higher confidence level than the user-based features since there exist situations that user-based features are too external and not reliable, *e.g.* some real spammers' accounts may post some normal comments when they stop spamming and take a break.

		Metrics				
Topic Quantity (k)	**Feature Set**	ACC	PRE	REC	F1	AUC
N/A	W	87.45	65.42	81.33	72.51	86.64
	WU	88.05	66.23	**84.20**	74.14	87.19
6	WT	90.54	77.37	75.63	76.49	86.80
	WTU	90.71	76.44	78.60	77.50	87.34
10	WT	90.64	78.36	74.60	76.43	86.57
	WTU	**91.17**	77.93	78.94	**78.43**	**87.75**
20	WT	90.69	**78.59**	74.57	76.53	86.66
	WTU	91.03	77.52	78.80	78.15	87.67
50	WT	89.83	73.69	77.83	75.70	87.05
	WTU	90.41	74.49	80.43	77.35	87.73
80	WT	88.00	67.36	79.63	72.98	86.46
	WTU	89.82	72.31	80.95	76.39	87.42
100	WT	86.65	63.58	80.56	71.07	85.91
	WTU	89.11	69.71	82.24	75.45	87.37

Table 4. Performance Results (%) for All Experiment Groups

As mentioned before, there is a difference between the performances of the classifiers with different feature sets, which is also proven in Figure 5. When topic quantity k = 10 which seems the optimal value, the ROC curves of the classifiers with different feature set and same topic quantity k =10 are generated in Figure 5. Among all four feature sets, WTU perform best in terms of the ROC curve. And there is a huge gap between the classifiers with topic-based features and those without them since both the group of W and WU and the group of WT and WTU are close within groups but dissimilar between groups.

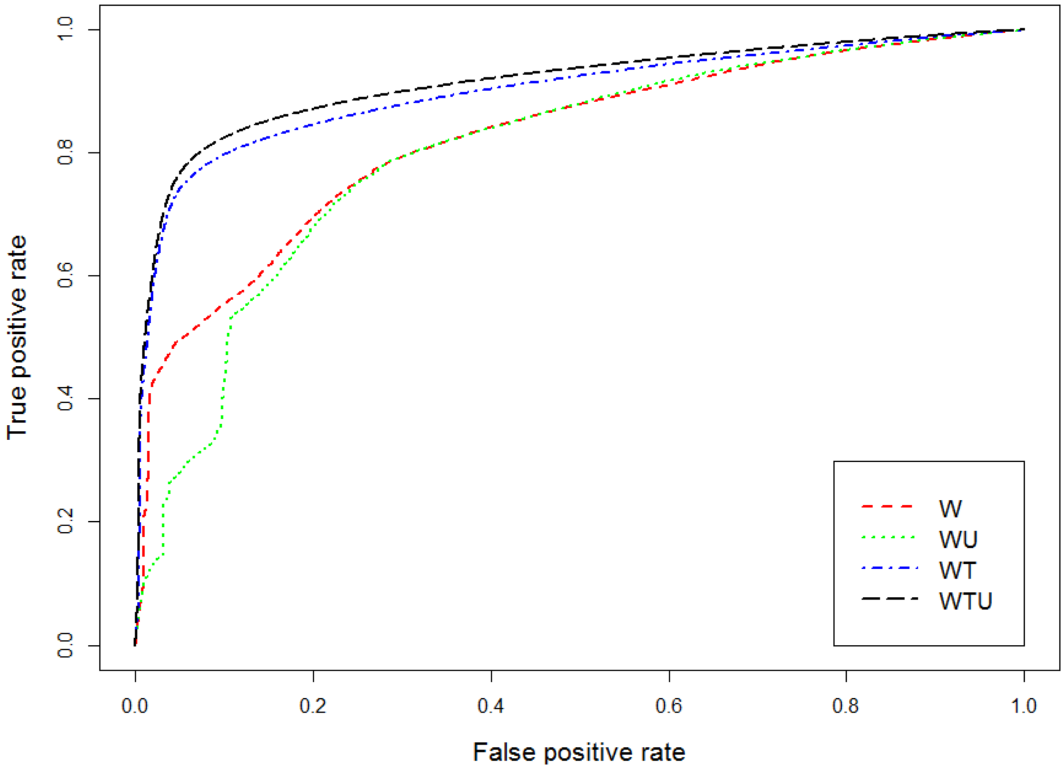


Figure 5. ROC Curves for SVM Classifiers with Topic Quantity k = 10

6 CONCLUSIONS AND FUTURE RESEARCH

In this research, starting from the abuse problem of user-generated contents in online social media, a novel social spam detection methodology is designed and implemented. To achieve a better detection performance, a comparison of the performance of detectors with different combinations of word-based feature, topic-based feature, and user-based feature is made by leveraging the L-LDA model and SVM classifier. For the three types of features examined in our research, word-based feature is generated by the *tf-idf* term weighting scheme, topic-based feature is mined by the L-LDA model, and user-based feature is extracted based on the average time interval of posting (ATI) and the average similarity of two adjacent comments (AS). Chi-Square test is adopted to perform feature selection for word-based features as well as generating the labels of latent topics mined by the L-LDA model in our research. After conducting a series of experiments based on a YouTube social spam dataset, the proposed methodology achieved an accuracy of 91.17% by leveraging all the aforementioned features.

As part of our future work, we will refine the existing topic-based features since some latent topics mined by the L-LDA model may be quite similar to each other. Further clustering of the initial latent topics seems to be a good way to further improve the quality of topic-based features. Besides, additional types of features such as network-based features will be explored in future research. In addition, crowdsourcing method will be explored to improve the quality of the current YouTube social spam dataset because we found occasional inaccuracy of the “Spam Hint” judgments. Finally, large scale of empirical experiments that involve social spam detection for multiple online social media will be conducted.

References

Abbasi, A. & H. Chen (2008) CyberGate: a design framework and system for text analysis of computer-mediated communication. *MIS Quarterly,* 32**,** 811-837.

Ahmed, S. & F. Mithun. (2004). Word Stemming to Enhance Spam Filtering. In *CEAS*. Citeseer.

Al-Harbi, S., A. Almuhareb, A. Al-Thubaity, M. S. Khorsheed & A. Al-Rajeh. (2008). Automatic Arabic Text Classification. In *Proceedings of The 9th International Conference on the Statistical Analysis of Textual Data*.

Alexandrov, M., A. F. Gelbukh & G. Lozovoi. (2001). Chi-Square Classifier for Document Categorization. In *CICLing'01*, 457-459.

Bíró, I., D. Siklósi, J. Szabó & A. A. Benczúr. (2009). Linked latent dirichlet allocation in web spam filtering. In *Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web*, 37-40. ACM.

Bíró, I., J. Szabó & A. A. Benczúr. (2008). Latent dirichlet allocation in web spam filtering. In *Proceedings of the 4th international workshop on Adversarial information retrieval on the web*, 29-32. ACM.

Benevenuto, F. i., T. Rodrigues, V. i. Almeida, J. Almeida & M. Gon\ccalves. (2009). Detecting spammers and content promoters in online video social networks. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 620-627. New York, NY, USA: ACM.

Blei, D. M., A. Y. Ng & M. I. Jordan (2003) Latent dirichlet allocation. *the Journal of machine Learning research,* 3**,** 993-1022.

Boyd, D. & J. Heer. (2006). Profiles as Conversation: Networked Identity Performance on Friendster. In *System Sciences, 2006. HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on*, 59c-59c.

Brody, S. & N. Elhadad. (2010). An unsupervised aspect-sentiment model for online reviews. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 804-812. Association for Computational Linguistics.

Brown, G., T. Howe, M. Ihbe, A. Prakash & K. Borders. (2008). Social networks and context-aware spam. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, 403-412. New York, NY, USA: ACM.

Casella, G. & E. I. George (1992) Explaining the Gibbs sampler. *The American Statistician,* 46**,** 167-174.

Chandramouli, R. (2011). Emerging social media threats: Technology and policy perspectives. In *Cybersecurity Summit (WCS), 2011 Second Worldwide*, 1-4.

Duan, Z., K. Gopalan & X. Yuan. (2007). Behavioral characteristics of spammers and their network reachability properties. In *Communications, 2007. ICC'07. IEEE International Conference on*, 164-171. IEEE.

Dunham, M. H. & D. Ming. (2003). Introductory and Advanced Topics. Prentice Hall.

Fairclough, N. (2003). *Analysing discourse: Textual analysis for social research*. Routledge.

Felt, A. & D. Evans. (2008). Privacy Protection for Social Networking Platforms. In *Proceedings of W2SP 2008: Web 2.0 Security and Privacy*.

Forman, G. (2003) An extensive empirical study of feature selection metrics for text classification. *the Journal of machine Learning research,* 3**,** 1289-1305.

Grier, C., K. Thomas, V. Paxson & M. Zhang. (2010). @spam: the underground on 140 characters or less. In *Proceedings of the 17th ACM conference on Computer and communications security*, 27-37. New York, NY, USA: ACM.

Halliday, M. A. & C. M. Matthiessen (2004) An introduction to functional grammar.

Jagatic, T. N., N. A. Johnson, M. Jakobsson & F. Menczer (2007) Social phishing. *Commun.ACM,* 50**,** 94-100.

Joachims, T. (1996). A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. DTIC Document.

Kaplan, A. M. & M. Haenlein (2010) Users of the world, unite! The challenges and opportunities of Social Media. *Business horizons,* 53**,** 59-68.

Lee, K., J. Caverlee & S. Webb. (2010). Uncovering social spammers: social honeypots + machine learning. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, 435-442. New York, NY, USA: ACM.

Lin, Y.-R., H. Sundaram, Y. Chi, J. Tatemura & B. L. Tseng. (2007). Splog detection using self-similarity analysis on blog temporal dynamics. In *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, 1-8. New York, NY, USA: ACM.

Liu, B., L. Liu, A. Tsykin, G. J. Goodall, J. E. Green, M. Zhu, C. H. Kim & J. Li (2010) Identifying functional miRNA–mRNA regulatory modules with correspondence latent dirichlet allocation. *Bioinformatics,* 26**,** 3105-3111.

Markines, B., C. Cattuto & F. Menczer. (2009). Social spam detection. In *Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web*, 41-48. New York, NY, USA: ACM.

Mesleh, A. M. d. A. (2007). Chi Square Feature Extraction Based Svms Arabic Text Categorization System. In *ICSOFT (PL/DPS/KE/MUSE),* eds. J. Filipe, B. Shishkov & M. Helfert, 235-240. INSTICC Press.

Myers, E. W. (1986) An O(ND) Difference Algorithm and Its Variations. *Algorithmica,* 1**,** 251-266.

Nevo, D., I. Benbasat & Y. Wand (2012) Understanding Technology Support for Organizational Transactive Memory: Requirements, Application, and Customization. *Journal of Management Information Systems,* 28**,** 69-98.

O'Callaghan, D., M. Harrigan, J. Carthy & P. a. Cunningham (2012a) Identifying Discriminating Network Motifs in YouTube Spam. *CoRR,* abs/1202.5216.

--- (2012b) Network Analysis of Recurring YouTube Spam Campaigns. *CoRR,* abs/1201.3783.

Ramage, D., D. Hall, R. Nallapati & C. D. Manning. (2009). Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1,* 248-256. Association for Computational Linguistics.

Salton, G. & M. J. McGill (1986) Introduction to modern information retrieval.

Sebastiani, F. (2002) Machine learning in automated text categorization. *ACM computing surveys (CSUR),* 34**,** 1-47.

Soucy, P. & G. W. Mineau. (2005). Beyond TFIDF weighting for text categorization in the vector space model. In *IJCAI*, 1130-1135.

Sureka, A. (2011) Mining user comment activity for detecting forum spammers in youtube. *arXiv preprint arXiv:1103.5044*.

Author. (2007). Theft, fraud cost retailers $8 million a day: study. *The Ottawa Citizen*.

Ukkonen, E. (1985) Algorithms for approximate string matching. *International Conference on Foundations of Computation Theory,* 64**,** 100-118.

van Marle, D. (2011). IP telephony shifts from unified communications to social media. In *FITCE Congress (FITCE), 2011 50th*, 1-4.

Wang, C., D. Blei & F.-F. Li. (2009). Simultaneous image classification and annotation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 1903-1910. IEEE.

Wang, D., D. Irani & C. Pu. (2011). A social-spam detection framework. In *Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, 46-54. New York, NY, USA: ACM.

Wu, G., D. Greene, B. Smyth & P. a. Cunningham. (2010). Distortion as a validation criterion in the identification of suspicious reviews. In *Proceedings of the First Workshop on Social Media Analytics*, 10-13. New York, NY, USA: ACM.

Yan, C. & L. Zheng (2010) LDA-based model for online topic evolution mining. *Computer Science,* 11**,** 042.

Yoo, K.-H. & U. Gretzel. (2009). Comparison of Deceptive and Truthful Travel Reviews. In *Information and Communication Technologies in Tourism 2009,* eds. W. Höpken, U. Gretzel & R. Law, 37-47. Springer Vienna.

Zinman, A. & J. Donath. (2007). *Is Britney Spears spam?* Mountain View, CA.