Chen C, Wu K, Srinivasan V *et al.* The best answers? Think twice: Identifying commercial campagins in the CQA forums. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 30(4): 810–828 July 2015. DOI 10.1007/s11390-015-1562-x

The Best Answers? Think Twice: Identifying Commercial Campagins in the CQA Forums

Cheng Chen (陈 诚), *Student Member, ACM, IEEE*, Kui Wu (吴 逵), *Senior Member, IEEE, Member, ACM*
Venkatesh Srinivasan, and Kesav Bharadwaj R

Department of Computer Science, University of Victoria, Victoria, V8P 5C2, Canada

E-mail: {cchenv, wkui, srinivas}@uvic.ca; rkesav1990@gmail.com

Received January 26, 2015; revised March 17, 2015.

Abstract In an emerging trend, more and more Internet users search for information from Community Question and Answer (CQA) websites, as interactive communication in such websites provides users with a rare feeling of trust. More often than not, end users look for instant help when they browse the CQA websites for the best answers. Hence, it is imperative that they should be warned of any potential commercial campaigns hidden behind the answers. Existing research focuses more on the quality of answers and does not meet the above need. Textual similarities between questions and answers are widely used in previous research. However, this feature will no longer be effective when facing commercial paid posters. More context information, such as writing templates and a user's reputation track, needs to be combined together to form a new model to detect the potential campaign answers. In this paper, we develop a system that automatically analyzes the hidden patterns of commercial spam and raises alarms instantaneously to end users whenever a potential commercial campaign is detected. Our detection method integrates semantic analysis and posters' track records and utilizes the special features of CQA websites largely different from those in other types of forums such as microblogs or news reports. Our system is adaptive and accommodates new evidence uncovered by the detection algorithms over time. Validated with real-world trace data from a popular Chinese CQA website over a period of three months, our system shows great potential towards adaptive detection of CQA spams.

Keywords CQA forum, anomaly detection, paid poster, online detection system

1 Introduction

Web 2.0 social websites are playing an increasingly important role in the Internet by utilizing the wisdom of crowds. One such example is the Community Question and Answer (CQA) portals on which users can post and answer questions, such as Yahoo! Answers[①], Naver[②], and Baidu Zhidao[③]. Some CQA websites like Quora[④] attract users by offering professional answers, most of which come from verified people in reality. These websites gain popularity and trust by providing a sense of interaction between the questioner and the masses. With millions of archived Q&A sessions, CQA forums have become a major source of advice for many Internet users.

As a large knowledge base of crowds, the archived Q&A sessions have been used for automatic question answering and recommendation. Nevertheless, the quality of user-generated content in the Q&A sessions varies drastically. For instance, some answers do not

Regular Paper
Special Section on Data Management and Data Mining
This research was partially supported by the Natural Sciences and Engineering Research Council of Canada under Grant No. 195819339 and the Globalink Internship of Mathematics of Information Technology and Complex Systems (MITACS) of Canada.

①https://answers.yahoo.com/, March 2015.

②http://www.naver.com/, March 2015.

③http://zhidao.baidu.com/, March 2015.

④http://www.quora.com/, March 2015.

match the questions and even contain spam and rude words. In recent years, tremendous efforts have been made to locate better answers and remove spam from the archived questions and answers resource. Techniques such as analysis of text, user-question-answer's link relationship, and user feedback features have been used in tools like PageRank to identify high-quality web pages[1-3].

Existing techniques, however, may not work well in the presence of the so-called Internet water army, a large crowd of hidden posters who get paid to generate artificial content in the social media for commercial profits. Paid posters have become popular with the booming of crowd-sourcing marketing. As confirmed in [4], crowd-sourcing systems such as Amazon's Mechanical Turk, Zhubajie (a similar Chinese crowd-sourcing site), have been broadly used for commercial campaigns. Due to their popularity, the CQA forums have become the targets of those campaigns that create untruthful Q&A sessions for commercial purpose. Consider the following example:

Question: I tried several methods to lose weight but all failed. What should I do? Please give me some advice!

Best answer: Don't worry, I have experienced the same pain as you. Firstly, you have to keep a healthy diet. Be careful about the nutrition in your food and never eat fast food. Secondly, don't sit too long in front of a computer. Finally, perform physical exercise everyday. What's more, you can also try a product named X. This product contains ingredients such as ... and can help you lose weight without any risks.

The above Q&A session is actually generated by paid posters. The answer provides very practical advice at first and then give suggestions on the product which needs to be promoted. The practical advice part is to earn the trust of the users. We have observed that fake answers generated by paid posters are often long enough and quite relevant to the questions, and some paid posters involved in the fake Q&A sessions are ranked high according to the website's reputation system.

Based on textual similarities, previous work [5-7] is likely to treat the above answer as of high quality due to the high relevance of textual features between the answer and the question content. As a result, the output may contain commercial spam, resulting in a credibility problem. Therefore, additional strategies, such as writing templates, public calls for commercial campaigns, and a poster's track reputation, should be integrated for the effective detection of paid posters. Furthermore, most existing work relies on offline analysis, while end users demand for instant help and should be warned of potential commercial campaigns when they browse a CQA forum. The call for a real-time response system that can detect potentially fake Q&A sessions on the fly is strong.

We tackle the above two challenges in this paper by designing an adaptive detection system tailored specifically for CQA forums. Our contributions are as follows.

• We discover that the behavioral features of paid posters are different in CQA forums when compared to other types of forums such as microblog (also called Weibo, a Twitter-like service in China) and news reports. We identify the special features of paid posters in CQA forums that are useful in the detection.

• Based on the identified special features, we design a detection method which uses machine-learning techniques and assigns credibility scores to each of the best answers by using semantic analysis and user features, such as users' history data.

• We implement an adaptive detection system which automatically analyzes the hidden patterns of commercial spams and raises alarms instantaneously to end users whenever a potential commercial campaign is detected. Our system is adaptive and accommodates new evidence gathered by the detection algorithms over time.

2 Related Work

Our research is mostly related to work on spam detection and recognizing experts or authoritative users and trustworthy content in the social media. These topics have become crucial to many online services, especially the question and answer communities, whose contents are generated by millions of users. We discuss prior work on two aspects.

2.1 Retrieving High-Quality Answers in CQA Sites

A lot of research has been done on finding high-quality content in CQA sites. However, we have not seen any paper which explicitly solved the credibility problem introduced in our work. Usually, researchers treated the best answers as the high-quality answers which has the risk of being defeated by the paid posters. In our work, we explicitly consider the credibility issues about the best answers.

Jeon *et al.*[1] attempted to predict the quality of answers in a community-based question answering service with only non-textual features, such as answerer's acceptance ration, answer length and user's recommendation. They assumed the user feedback was a reliable source for the evaluation. Jurczyk and Agichtein[2] presented a study of link structure of Yahoo! Answers. They adopted a variant of the HITS algorithm[8] for finding experts in the Q&A portal. Their research was also based on the assumption that the user feedback could be used to assign weights on the edges of their graph representing user relationships.

Liu *et al.*[5] applied their automated summary technique to summarize answers for questions which ask for opinions. They used cosine similarity to cluster topic-oriented answers and eliminated irrelevant ones. Bian *et al.*[6] tried to use both relevance between questions and answers and the quality of answers to retrieve good answers for a user query. Both textual features and statistical features such as user ratings are used in their approach. Later, in another work by Bian *et al.*[9], they explicitly considered the effect of several vote spam attacks. Such activities involve malicious voting for specific answers to improve their ranking and to decrease the ranking of competitors at the same time.

Agichtein *et al.*[3] studied the basic elements of social media and combined three features of the social media (Yahoo! Answers) to facilitate the task of identifying high-quality content, namely intrinsic content quality, interactions between users and content usage statistics. Traditional link analysis algorithms are used to calculate the hubs and authorities scores (as in HITS algorithm[8]), and PageRank scores[10]. In addition, usage statistics such as the number of clicks of the Q&A session are used to complement the link-based analysis.

Pera and Ng[11] developed a CQA refinement system that could retrieve top-ranked answers for a user query based on similarity scores and the length of the answers.

Fichman[12] conducted a comparative study of answer quality on multiple Q&A websites, Yahoo! Answers, Wiki Answers⑤, Askville⑥ and the Wikipedia Reference Desk⑦. Accuracy, completeness and verifiability were used as the quality measures for cross platform comparison. Fichman found that the quality of answers was significantly improved only in terms of answer completeness and verifiability, rather than the answer accuracy.

Sakai *et al.*[13] proposed system evaluation methods for the task of selecting or ranking answers for a given question. They noticed that the asker-selected best answers might be biased and even if they were not, there might be other good answers besides the best ones. In order to overcome the bias problems of BA-based evaluation, they hired four assessors to independently assess every answer for the Q&A answers. Their experiments showed that their methods found substantial difference between systems that would have been overlooked by BA-based evaluation. In our point of view, we announce that the best answers not only are biased, but also could be unreliable or fake commercial campaign.

2.2 Other Research Work About Crowd-Sourcing Spams in Different Realms

Previous research has also investigated the crowd-sourcing spam in other areas. Jindal and Liu[14], Ott *et al.*[15] and Mukherjee *et al.*[16] attempted to detect fake review or opinion spam in the online shopping stores, like Amazon's online store. Similar to research on CQA websites, they also used textual similarity features and user-oriented features, like ratings and history records. Huang *et al.*[17] developed a regression model with features suggesting quality-biased short text in Microblogging service, Twitter. They judged the quality of tweets based on relevance, informativeness, readability, and politeness of the short content and assigned different scores from 1 to 5. However, they did not explicitly present how they define a spam-like tweet. Huang *et al.*[18] conducted a similar study of commercial spam on blogging sites. They showed that the propaganda of some products in the comment of a blog post was crucial in detecting the malicious comments. The propaganda appeared in the form of URL, phone number, E-mail address, MSN numbers, etc.

3 Data Collection and Labeling

3.1 Data Collection

Users who register on Baidu Zhidao participate in various Q&A sessions, as either question askers or repliers. Since we already know that paid posters who ac-

⑤http://wiki.answers.com/, March 2015.

⑥http://askville.amazon.com/Index.do, March 2015.

⑦http://en.wikipedia.org/wiki/Wikipedia:Reference_desk, March 2015.

cept missions from crowd-sourcing sites create a variety of Q&A sessions on the site for product propaganda, the collecting process can be targeted directly to the product campaigns. In addition, since the readers tend to pay more attention to the best answers and also due to the manner in which online paid posters are supposed to work, we only collected the best answers and ignored other ones. This is to avoid collecting a large amount of irrelevant information for this study.

In order to collect campaign Q&A sessions, we first visited the crowd-sourcing websites, where the paid posters apply for campaign tasks and get paid, as stated in Section 1. From the campaigns calling for paid posters, we selected 11 closed requests because the paid posters who worked for the 11 products had finished the tasks. We extracted keywords for the 11 products and searched for Q&A sessions with them on Baidu Zhidao. We used a crawler to visit and download the web pages associated with searching results. These sessions included not only the campaign sessions, but also the normal sessions containing the keywords. After parsing all the collected web pages, we obtained a group of target users, including both paid posters and normal users, as well as the links to the users' homepages hosted by Baidu Zhidao.

By following the users' homepages, we could find useful information for our research. For example, a user's homepage provides the Q&A sessions where this user posted his/her answers (the question answering records). The question-answer history provides a good knowledge on the multiple campaigns that a potential paid poster might have been involved. Having obtained the initial dataset of IDs and links, we then visited each user's homepage, and retrieved every Q&A session that the user participated in. We only collected the closed Q&A sessions (i.e., the best answer determined). A closed Q&A session implies that users can no longer post new answers to the question, but they can click the "Like" button to support the posted answers, including the best answer and other answers.

From those Q&A sessions, we finally extracted information used in our analysis. The recorded information from those web pages includes questioner ID, answer ID, time, title, question content, answer content, user feedbacks (visited times, ratings). For text information (Q&A title and question/answer content), we have removed stopwords from the raw data.

From the Q&A website, Baidu Zhidao, we collected 6 462 users' question-answer history records accumulated during a three-month period from October to December in 2011. For each user, we built a list of history information, showing the question, answer, participated user IDs, and other features. Associated with the 6 462 user IDs, we have 75 200 Q&A sessions in total, all having the best answer.

In the following, we describe a solicitation example of Q&A campaign.

3.1.1 A Solicitation Example

Mission title: a_brand's_name (brand *A*): Baidu Zhidao, 5 RMB (0.8 USD) per valid Q&A.

The title indicates that this is a Q&A campaign mission for brand *A*, conducted on Baidu Zhidao CQA website. The payment is 5 RMB (0.8 USD) for each approved Q&A session.

General requirements:

1) Normally, it takes three days to complete a Q&A campaign. Day 1: post the question. Day 2: use a different IP address and login account to answer the question. Day 3: select the answer as the best one. The Q&A will be invalid (you will not get paid) if the answer is not selected as the best one, or the best answer is deleted within 72 hours.

2) One account can only be used to post/answer one question regarding the same solicitation.

3) If you answer a question posted by yourself, you must change your IP address and the account.

4) You must mention brand *A* in your answer.

5) Once you complete a Q&A, you need to send the link to the mission supervisor for evaluation. You will get paid once it is approved.

Keywords: detergent for car washes, car washing plant.

Question template: What is a good detergent for car washes?

Answer template: There are many different detergents, such as brand *A*, brand *B* and brand *C*. The detergent of brand *A* is better because it does not need wiping and it takes only seven minutes to clean a car. Of course, you will need some washing equipment. If you want to open a car washing plant, I highly recommend brand *A*.

The question by paid poster 1: I recently bought a car. It gets dirty after some driving. I would like to know which detergent works well.

The answer by paid poster 1: Cleaning is necessary to keep a vehicle in good shape. Important electrical connectors should be protected before cleaning. Then you can use the detergent of brand *A* to wash every individual part. Do not use high pressure washer.

The question by paid poster 2: Which detergent is good for car wash?

The answer by paid poster 2: I always use the detergent of brand *A* in my car washing plant because customers are very satisfied with brand *A*. You do not need wiping. With the help of washing equipment, you can finish washing a vehicle in seven minutes.

Note that this example only shows one of many possible working patterns of campaign Q&A. In practice, paid posters do not have to post questions by themselves. They could find related questions posted by regular questioner and answer them according to campaign templates.

3.2 Manual Data Labeling

To get a sample dataset for feature analysis, campaign sessions should be differentiated from the normal ones. By reading the best answers and *cross-checking* the Q&A templates from the crowd-sourcing websites such as Zhubajie[8] and Tiancaicheng[9], we manually label the Q&A sessions in the dataset. We summarize the applied techniques below.

1) Since we have collected a list of 11 products which were hyped in the Baidu Zhidao, we could compare the Q&A content with the campaign templates. If the product's name is in the 11 initial samples and the contents match the templates, such as the descriptive words and the organized pattern of sentences, we labeled it as a campaign Q&A session. We stress that there is difference between our work and related research which needs to judge the quality of answers. The evaluation of quality of answers is usually based on question-answer relevance, length of the texts, grammar correctness, politeness, and so on. To obtain a reliable dataset, researchers often rely on multiple assessors and are faced with the difficulty of reaching an agreement among the multiple evaluation results. Our labeling method differs from the above and largely avoids the annotation difficulty, because we know exactly the name of the hyped product and how paid posters would write the Q&A sessions.

2) When we encountered new products not in the list of 11 initial samples, we recorded the product's name and searched it in the crowd-sourcing websites. If we found the template of this product, we use the above method to compare their contents.

3) If a new product is listed in the campaign websites but the template is not available, we followed some special features normally found in Email spam to make a decision. For example, a spam may use different fonts to write the telephone numbers and insert special characters between the product's name. This type of operations is usually used to escape detection by the filter system. We labeled the session as campaign if the product's name is in a campaign list and the best answer has special features similar to Email spam.

4) If we could not find the new product in the campaign websites, we then tried to identify potential templates used in the same category of products and special features obvious in an Email spam. If none of those could be identified, we labeled the session as a normal session.

Up to now, we have labeled 4 998 samples in our dataset. Among these, 2 147 samples are campaign Q&A sessions and the other 2 851 samples are normal ones. The sample size is large enough for our current study. Since we selected 11 campaigns, which were posted on the crowdsourcing websites, as the seeds of our crawler, and we further encountered new products involved in campaigns, the proportion of campaign sessions is relatively high in the dataset.

When we manually labeled our datasets, we carefully read the contents of a user's post. The meaning can be understood by human but is hard to use in machine learning based classification. Even with the above template based labeling method, it is not easy to write an algorithm to automatically identify a campaign session because a poster may re-phrase the template in his/her own words. Due to these reasons, we need to search for statistical features that can be effectively used towards building a detection system.

4 Analysis of Statistical Features

4.1 Insufficiency of Existing Statistical Features

Here, we demonstrate the limitations of the features used in our previous work[19] on the detection of Internet water army in news report towards addressing the problem we study in this paper.

4.1.1 Interval Post Time

In [16], Mukherjee *et al.* defined several spamming indicators for modelling the behaviour of fake review

[8] http://www.zhubajie.com/, March 2015.

[9] http://www.tiancaicheng.com/, March 2015.

writers. They found that spammers of a spam group tend to post reviews during a short time interval. This feature has been shown to be a good indicator to detect Internet water army in news report websites[19].

In our work, we consider two timestamps for a Q&A session: one is the time when the questioner posts the question topic (the ask time), and the other one is the time when the best answer is posted by a replier (the best answer posted time). We define *interval post time* as the latter timestamp minus the former one.

In Fig.1, we show the approximated probability distribution of interval post time with dot-dashed lines for campaign sessions and solid lines for non-campaign sessions. The x-axis is drawn by log scale.

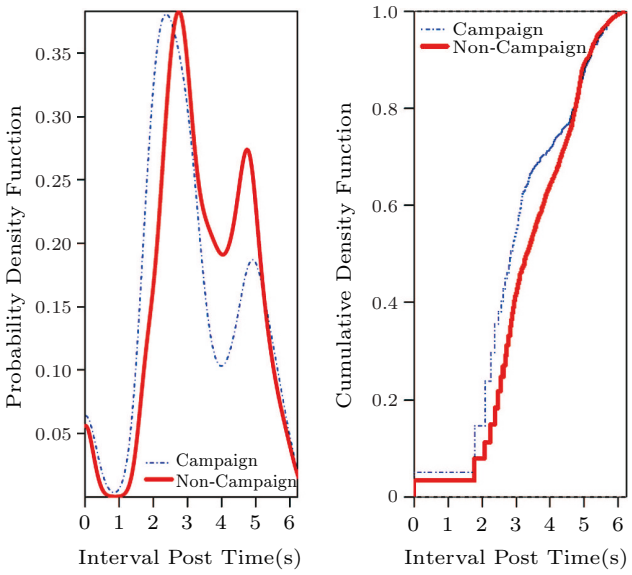

Fig.1. PDF and CDF of the interval post time.

From Fig.1, we find it difficult to tell the difference between campaign and non-campaign Q&A sessions. Two reasons may contribute to the above phenomenon. There are many normal users who spend much time on the Q&A website and try to post answers to *open* questions, especially those questions associated with some *rewards points*. These people are known as *bounty hunters*. Most bounty hunters post very good answers because they want to get more rewards points. On the other hand, online paid posters, before they post and choose the best answer, normally wait for some random time for other answers appearing in the session. This is to give readers a fake impression that the best answer is selected among many answers. While paid posters try to finish a job as quickly as possible in news review websites[19], the same behaviour does not exist here.

4.1.2 Number of Other Answers

Before the question is closed, users can post their own answers. This variable counts the number of answers other than the best one. Intuitively, if the paid posters create the sessions themselves, they may not have patience to wait for more replies. They could close the sessions and get paid as soon as possible. To test this conjecture, we show the probability distribution of this feature for campaign sessions and normal sessions in Fig.2.

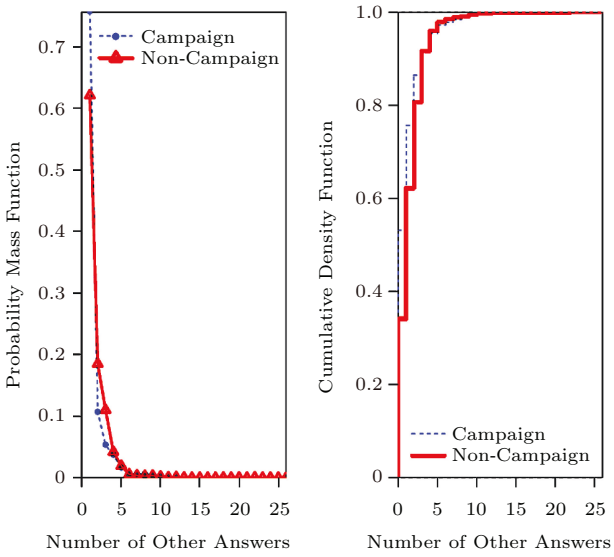

Fig.2. PMF and CDF of the number of other answers.

Similar to the interval post time, the number of other answers does not indicate much difference for the two types of Q&A sessions. This invalidates the above conjecture and we do not consider it as a good feature for the detection of paid posters in CQA portals.

4.1.3 Number of Likes

Similar to the "Like" button in Facebook, if other readers find the best answer to be helpful, they may click the "like" button. The number on the button indicates the total number of clicks. Intuitively, this feature represents users' feedback and should be helpful in identifying trustful answers. The more "likes" an answer receives, the more likely it is a good answer. However, as shown in Fig.3, this is not a reliable feature. This is because the paid posters could click the button themselves and even use different user IDs to click multiple times. This behavior is also confirmed in [9] as the "vote spam attack".

4.1.4 *Relevance Between Questions and Best Answers*

This feature is extensively used before in identifying high-quality answers[3,5-7]. The previous work is usually based on following assumptions:

1) Semantically high relevance between questions and answers indicates high quality.

2) Selected best answers should have higher quality than other answers.

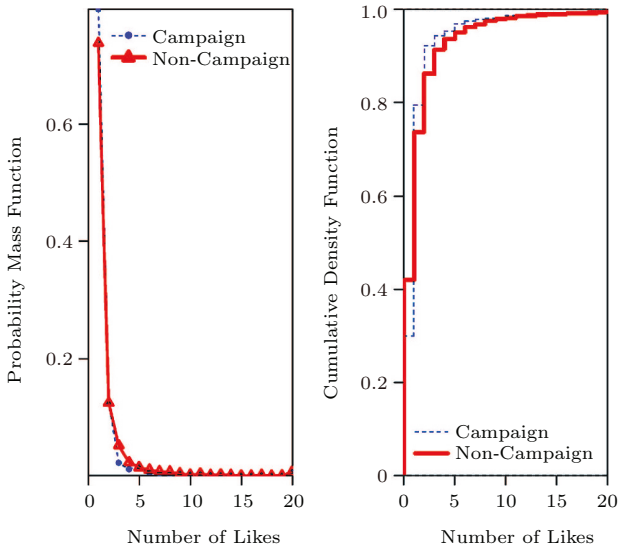

Fig.3. PMF and CDF of the number of likes.

The above assumptions are risky for the detection of potential campaigns created by paid posters. In commercial campaigns, answers with high quality are rather misleading and would beat the retrieval mechanism. Many answers are well-organized and highly related to the questions. In this sense, a "high-quality" answer does not necessarily mean trustworthiness. Thus, we do not consider the relevance measure in our work.

4.2 Special Features for CQA Portals

The limitations of existing statistical features shown above lead us to look for new features specific to users in CQA websites.

4.2.1 *Spam Grade of Questioner ID (SGqID)*

It indicates whether the questioner tends to ask campaign questions. A paid poster may use multiple IDs (for questioning and answering, respectively) to complete a Q&A campaign. Therefore, a questioner ID which appears in many malicious Q&A sessions is more likely to be associated with a paid poster. For a given questioner ID (qID), we calculate the ratio of the number of campaign sessions to the total number of sessions in which the user has participated, as shown in (1).

$$SGqID = \frac{q_1}{q_0 + q_1}, \tag{1}$$

where q_0 and q_1 are the number of non-campaign and campaign sessions where the user appears as the questioner, respectively. To avoid 0 probability, we assign 0.5 to q_1 when $q_1 = 0$. This is a technique known as Laplace correction or Laplace estimator. It has been widely adopted to avoid zero frequency problem, which arises when an entity (qID in this case) does not occur in one of the classes (campaign or non-campaign sessions). Laplace correction assigns the entity of each class a fixed pseudocount of a certain number. 0.5 or 1 is commonly used as the pseudocount in practice. In addition, if the system does not have enough information for a certain user (i.e., the denominator is less than 5), we set its *SGqID* value to 0.5. This decision follows the Maximum Entropy Principle[20], i.e., we should "make use of all the information that is given and scrupulously avoid making assumptions about information that is not available." We refer to these two techniques (Laplace correction and Maximum Entropy Principle) as "data twist" in Section 7.

4.2.2 *Spam Grade of Answerer ID (SGaID)*

It indicates whether the best answer poster tends to write campaign answers. The intuition behind is similar to that of *SGqID*. For a given answerer ID (aID), we calculate the ratio of the number of campaign sessions to the total number of sessions in which the user has participated, as shown in (2).

$$SGaID = \frac{a_1}{a_0 + a_1}, \tag{2}$$

where a_0 and a_1 are the number of non-campaign and campaign sessions where the user appears as the poster of the best answers, respectively. Similar to $SGqID$, to avoid 0 probability, we specify 0.5 to a_1 when $a_1 = 0$. If the system does not record enough information, we set its $SGaID$ value to 0.5.

4.2.3 *Spam Grade of the Text (SGtext)*

It indicates whether the collection of words in sessions associated with a user tends to be campaign specific. For a given mission of Q&A campaign, different paid posters may share the same template, which is

provided by the mission supervisor. Therefore we can expect similar words or expressions in the postings. To calculate this feature, we need to perform statistical analysis over the words. Text information of a Q&A session consists of the title, the content of question, and the content of the best answer. We remove the duplicate words so that we can get a collection of distinct words ($word_1$, $word_2$, $word_3$, ... , $word_n$) for each Q&A session. For each word, we calculate *spam grade* which characterizes the property of the word, i.e., whether it is more campaign oriented or non-campaign oriented. Words with higher benchmark are more likely to imply hidden promotion behavior, i.e., they appear in many campaigns sessions but few normal ones. To get rid of the impact of different length, we take the average value over the summation of the benchmarks of all words as the spam grade of the whole text. For each word, the definition of spam grade is defined in (3).

$$SGword_i = \log\left(\frac{N+1}{n_i+1}\right) \times \frac{s_i+1}{S+1}, \tag{3}$$

where N and S are the total number of non-campaign and campaign sessions in the databases respectively and n_i and s_i are the number of non-campaign and campaign sessions where the $word_i$ appears respectively. The intuition behind this definition is to obtain a weighting scheme showing whether a word tends to be campaign specific, based on the fraction of campaign and non-campaign Q&A sessions that contain the word. The definition of the spam grade takes both non-campaign and campaign sessions into consideration. This definition achieves desired effects: 1) the spam grade is scaled up when the word occurs fewer times in non-campaign sessions and occurs in many campaign sessions; 2) the spam grade is scaled down when the word occurs more in non-campaign sessions and occurs fewer in campaign sessions; 3) the spam grade is neutralized when the word frequently occurs (or rarely occurs) in both non-campaign and campaign sessions.

We apply "log" to avoid a large value in the equation. The term "+1" is used to normalize the result in case of zero counts. Then the calculation of the spam grade of text with L distinct words is shown in (4).

$$SGtext = \frac{SGword_1 + SGword_2 + ... + SGword_L}{L}. \tag{4}$$

5 Detection Method

In this section, we firstly select features that are useful for detection. Then we introduce a supervised learning approach, logistic regression, to calculate campaign scores (which indicate whether a Q&A session tends to be a campaign) for Q&A sessions using the selected features. Based on the scores, we can distinguish normal answers from campaign answers.

5.1 Feature Selection

We sort the 4 998 labelled samples by the timestamp of best answers and take 3 500 of them as training set (1 183 commercial campaigns and 2 317 normal Q&A sessions) and the remaining 1 498 as test set (964 commercial campaigns and 534 normal Q&A sessions). Note that the split of the dataset is arbitrary so that we can obtain a general result.

Using the training set, we extracted the most important features by calculating the information gain ratio between the class label (campaign or non-campaign) and each feature we proposed in Section 4. Information gain ratio is defined by (5).

$$Gain\ Ratio = \frac{H(Y) + H(X) - H(Y,X)}{H(X)}, \tag{5}$$

where $H(Y)$ means the entropy of X.

The gain ratios for features are shown in Table 1.

Table 1. Information Gain Ratios for Each Feature

Feature	Gain Ratio
Interval post time	0.044 287 13
Number of other answers	0.016 364 62
Number of likes	0.044 591 28
$SGqID$	0.216 314 13
$SGaID$	0.302 493 65
$SGtext$	0.171 792 17

As shown in Table 1, the spam grade features are more significant in terms of information gain ratio. Therefore, the three "SG" features will be used to build the classification model.

Fig.4 exhibits the values using the three "SG" features on the entire dataset.

Through Fig.4, we can observe a clear gap between the campaign sessions and the non-campaign sessions. We can then apply the regression based approach to calculate the campaign score, which indicates whether a Q&A session tends to be a campaign.

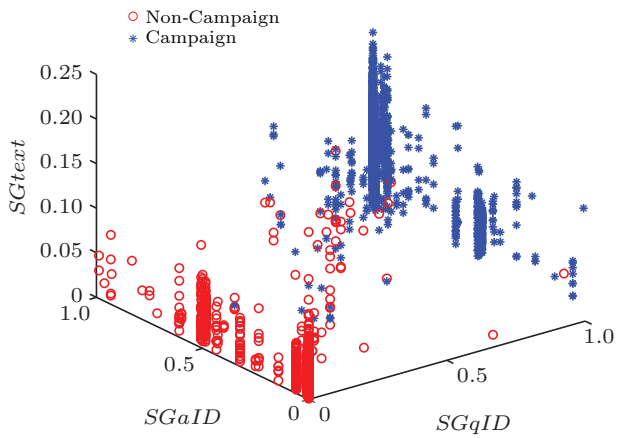

Fig.4. 4 998 samples captured by $SGqID$, $SGaID$ and $SGtext$.

5.2 The Algorithm

Fig.4 has already shown that the samples can be distinguished by the three selected features, $SGqID$, $SGaID$, and $SGtext$. In order to get a score indicating whether a Q&A session is a potential commercial campaign or not, we apply logistic regression as the learning method. We can use it to calculate values of $\Pr(Y = 1|\boldsymbol{x}, \boldsymbol{\theta})$ and $\Pr(Y = 0|\boldsymbol{x}, \boldsymbol{\theta})$. Here, Y is a indicator variable, where $Y = 1$ and $Y = 0$ represent campaign and non-campaign Q&A sessions, respectively. $\boldsymbol{x}$ is a vector of three features for each session. $\boldsymbol{\theta}$ is a vector of model parameters, each associated with a session feature and including an individually constant item (also called intercept term) which is not related to the session features.

By applying the sigmoid function, the hypothesis $h_{\boldsymbol{\theta}}(\boldsymbol{x})$ which outputs a score of $\Pr(Y = 1|\boldsymbol{x}, \boldsymbol{\theta})$ or $\Pr(Y = 0|\boldsymbol{x}, \boldsymbol{\theta})$ (termed as *campaign score*) is defined as follows:

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^{\mathrm{T}} \cdot \boldsymbol{x}}},$$

where $\boldsymbol{\theta}^{\mathrm{T}} \cdot \boldsymbol{x} = \theta_1 + \theta_2 \times SGqID + \theta_3 \times SGaID + \theta_4 \times SGtext$ and θ_1, θ_2, θ_3, θ_4 are elements of $\boldsymbol{\theta}$. To facilitate the vector calculation, we add a dummy attribute (with 1 as the value) to $\boldsymbol{x}$.

In practice, the higher the score, the higher the probability that the given session is a campaign session. The values of $\boldsymbol{\theta}$ will be learned by logistic regression. The objective then becomes a regression problem where we optimize the model so that the output campaign scores of sessions are close to their true labels (0 or 1).

The convex cost function of this optimization problem is given by

$$J(\boldsymbol{\theta}) = \frac{1}{m} \times \Sigma_{i=1}^{m} \log(1 + e^{-y^{(i)} \times (\boldsymbol{\theta}^{\mathrm{T}} \cdot \boldsymbol{x}^{(i)})}) + \frac{1}{2} \times \boldsymbol{\theta}^{\mathrm{T}} \cdot \boldsymbol{\theta},$$

where m is the number of samples in the training dataset and $\boldsymbol{x}^{(i)}$ is a vector consisting of m feature vectors of the i-th training sample. We use gradient descent method to find the minimum of the cost function and the corresponding values in $\boldsymbol{\theta}$.

5.3 Significance Test for Logistic Regression

In order to understand the relative contribution and overlap of the selected features, we perform a significance test for the proposed "SG" features in a full model (including all the three "SG" features). We also conduct multiple predictive comparisons between models which contain one or two of the "SG" features.

We use the "glm" function in R to train a full logistic regression model and examine p-values of the "SG" features. We find that the p-value of $SGqID$ is 0.364, while p-values of $SGaID$ and $SGtext$ are both below 0.05. It suggests that $SGqID$ is statistically insignificant in the full model. Furthermore, we also train different models using one or two of the "SG" features and report McFadden's R^2[21] (a pseudo R^2 measure for logistic regression) over the training set in Table 2.

Table 2. McFadden's R^2 for Different Combinations of "SG" Features

Feature	McFadden's R^2
$SGqID$	0.095 489 16
$SGaID$	0.699 391 15
$SGtext$	0.507 833 48
$SGqID + SGaID$	0.700 390 91
$SGqID + SGtext$	0.544 515 65
$SGaID + SGtext$	0.764 907 91
$SGqID + SGaID + SGtext$	0.765 102 96

In Table 2, we observe that the full model has the highest McFadden's R^2, which is only slightly better than the model with both $SGaID$ and $SGtext$. Next, we compare the predictive power on the test set of high R^2 models ($SGaID$, $SGqID + SGaID$, $SGqID + SGtext$, $SGaID + SGtext$ and the full model). Figs.5∼8 show the corresponding ROC curves and values of the area under the curve (AUC). Since the curves and AUCs of $SGaID + SGtext$ and the full model are nearly the same, we only show Fig.8 of the full model.

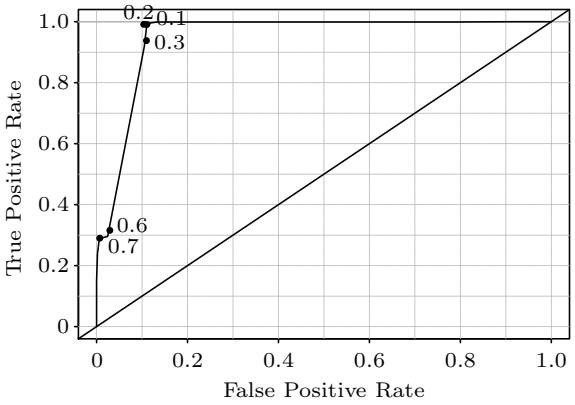

Fig.5. ROC curve of $SGaID$ on sorted data, AUC = 0.949 707 3.

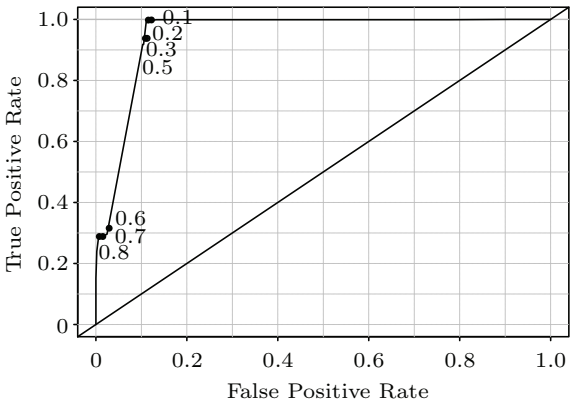

Fig.6. ROC curve of $SGqID + SGaID$ on sorted data, AUC = 0.950 282 3.

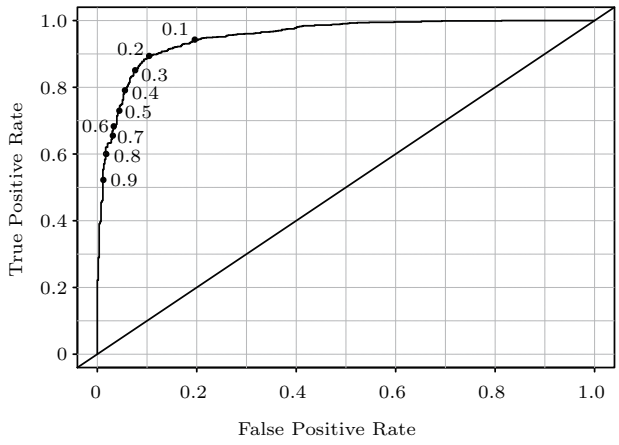

Fig.7. ROC curve of $SGqID + SGtext$ on sorted data, AUC = 0.952 979 5.

Fig.8 of the full model shows the overall best AUC (0.983 056 7). Considering that it also has the highest McFadden's R^2, we will take all "SG" features into consideration in Section 7.

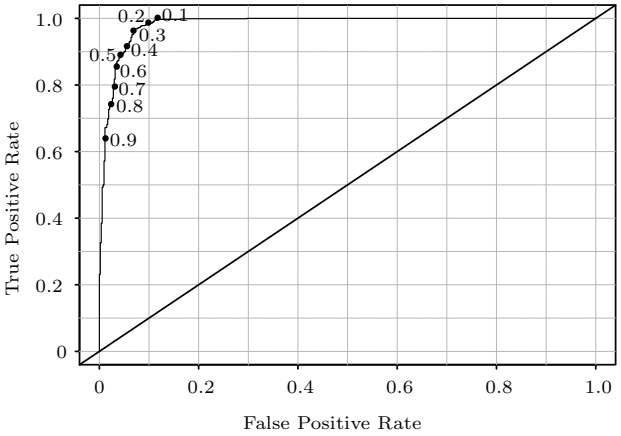

Fig.8. ROC curve of all "SG" features on sorted data, AUC = 0.983 056 7.

5.4 Classification Threshold

The value of $h_{\boldsymbol{\theta}}$ should be carefully determined. When $\boldsymbol{\theta}$ is optimized, we then calculate the campaign score of each Q&A session in the test dataset. The result is shown in Fig.8.

We observe that 0.4, 0.5 and 0.6 are closer to the top left of the figure than other values. Based on Fig.8, we set 0.5 as our threshold for $h_{\boldsymbol{\theta}}$. Note that, setting 0.5 as the classification threshold means that we would predict a positive label for a test sample when $\boldsymbol{\theta}^{\mathrm{T}} \cdot \boldsymbol{x}^{(i)} > 0$ while a negative label if $\boldsymbol{\theta}^{\mathrm{T}} \cdot \boldsymbol{x}^{(i)} < 0$.

6 Adaptive Detection System

In the previous section, we have shown that we can build a model to effectively calculate the campaign score and predict the labels of unknown sessions. In practice, newly emerging campaigns may have very different patterns of features as those used to build the model. It is necessary to develop an "adaptive" detection system that can update its database using new samples and evolve new model parameters, while maintaining stable detection performance over time. In this section, we present the design of such an adaptive detection system. We will evaluate its performance and assess whether manual labelling is necessary when adding new samples via an experiment based on a real-world dataset in Section 7.

The major components of the detection system include browser plugin and a remote server. Fig.9 shows the system architecture and the communication between the client plugin and the server.

The sequence of actions that take place when a user opens a Q/A session is:

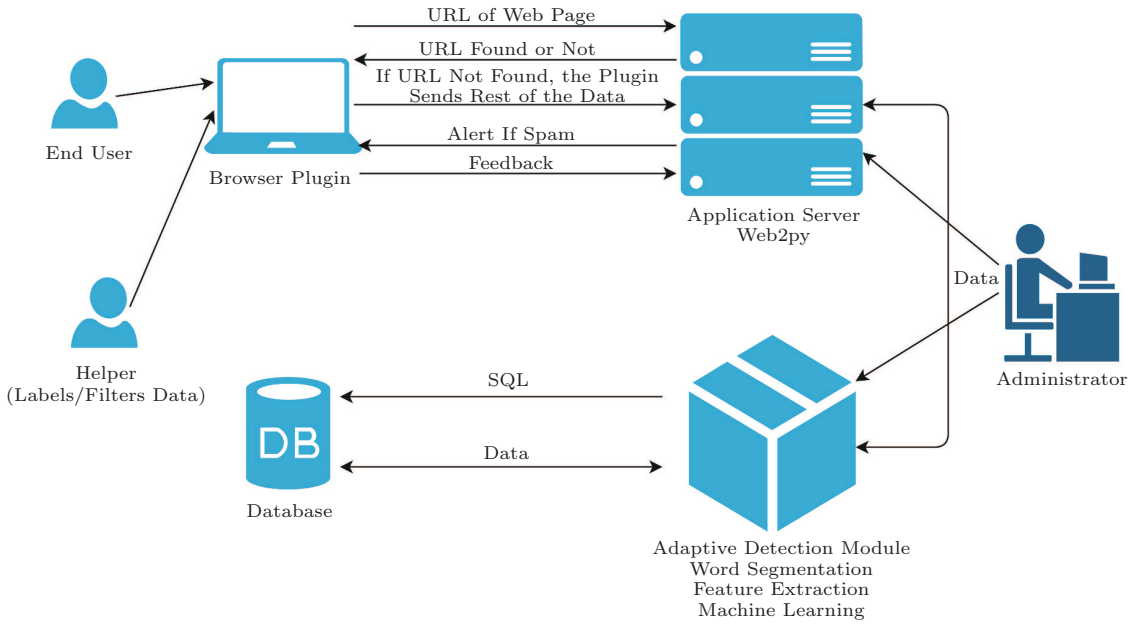

Fig.9. System architecture and communication between the client and the server.

1) The plugin first only sends the URL of the page to the server. The server searches for the URL in its database. If it is found, the server returns the score (spam rating) to the client. The client side script displays the result. This avoids unnecessarily sending complete web page to the server if it is already present in the database.

2) If the URL is not present, the server sends a response *not found* and the client after receiving the response sends the rest of the data to the server through another *XMLHTTPRequest* and waits for the server's response.

3) The server receives the data, segments the text into words, and stores it in the database. The server then extracts the statistical features necessary for the analysis from the data. Logistic regression analysis is performed to predict the class of the session (spam or no spam). If the session is classified as a spam, an alert is returned to the user.

4) The client-side script displays the result to the user.

5) (Optional) If the user is an authorized user, the user can provide feedback to the server (whether or not he/she feels the session is a campaign session). There are three types of users in the system: regular users are those who use our system and they are not granted the right to annotate sessions; helpers are those who have experience and are capable of helping label the data; the administrator is the person responsible for the management of the system. Note that helpers could be contracted out to employees of professional companies such as Rediff Shopping and eBay[16].

6) When newly labelled sessions are available, the system updates the detection model using existing and newly labelled data. Note that this step could be done regularly in a daily or even weekly basis.

7 Performance Evaluation

To evaluate the performance of online detection system, we use the collected data from Baidu Zhidao and replay the data in multiple iterations to simulate a real-world scenario. In particular, we pretend that initially we only have partial data and use the data as the training dataset to build a detection model. In each iteration, we add some new sessions and use them as the test dataset to test the performance of the detection system. At the end of an iteration, the new sessions are added into the training dataset, and the detection model is updated using the new training dataset. This step corresponds to the scenario that new data are labeled and added into the system. Then we repeat with another iteration. Note that we sort the Q&A sessions according to the timestamp when a session is closed. In this way, the performance is closer to that of a real-world scenario.

For the test, we begin with a 200-sample training set and build an initial detection model. At each iteration, the detection model will be tested on a 200-sample test

set. After evaluating the detection performance, we expand the training dataset with the 200 test samples, and update the detection model with the new training dataset. We repeat this process until we use up all 4 998 samples.

Fig.10 shows the ratio of non-campaign and campaign Q&A sessions in each iteration.

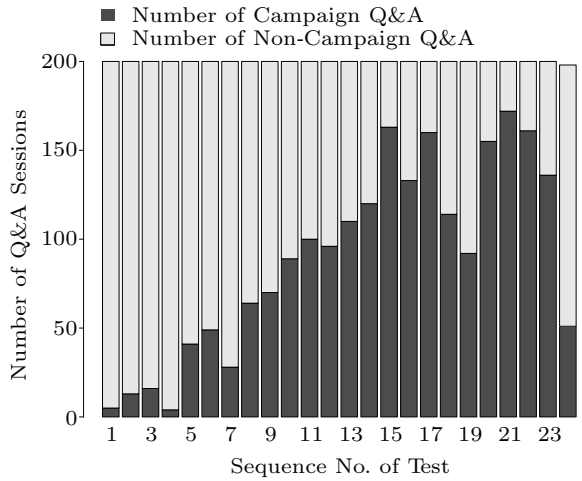

Fig.10. Ratio of non-campaign and campaign Q&A.

We evaluate the following four performance metrics:

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}},$$

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}},$$

$$F\text{-measure} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}},$$

$$\text{accuracy} = \frac{\text{true negative} + \text{true positive}}{\text{total number of users}}.$$

In the following, we will perform comprehensive experiments on the dataset, comparing different methodologies. In particular, we assess whether manual labelling is necessary (Subsections 7.1 and 7.2), and compare the adaptive model to the fixed model (Subsection 7.3) and linear classifiers as well as non-linear classifiers (Subsection 7.4). We further demonstrate the effectiveness of our model by showing that a model trained by non-twisted data and a model with text-only information do not perform well. These results are described in Subsections 7.5 and 7.6, respectively.

7.1 Adaptive Model with Manual Labelling

We first conduct experiments of an adaptive model with manual labelling, i.e., the database is updated using manual labelled (ground-truth) new samples. Fig.11 and Fig.12 show the update of model parameters and the detection performance in each iteration, respectively.

In Fig.11, "Theta 1", "Theta 2", "Theta 3" and "Theta 4" are parameters for the dummy attribute, $SGqID$, $SGaID$, and $SGtext$, respectively. We can observe that the detection model tends to converge after enough sessions have been added into the database over several iterations. For example, after 10 iterations, the precision achieves 85%∼90%.

We also notice that there is a "degraded" point at the 15th iteration in the recall, F-measure and accuracy figures. After carefully checking the log file (true/false positive and true/false negative) of this iteration, we find out that the false negative is very high, which means a large number of campaign sessions are classified as non-campaign ones. In addition, the continuously generated test set keeps changing because we sort the Q&A sessions according to the timestamp when a session is closed. Since the 200 samples of the 15th test have very different patterns, it affects the performance of the detection model significantly.

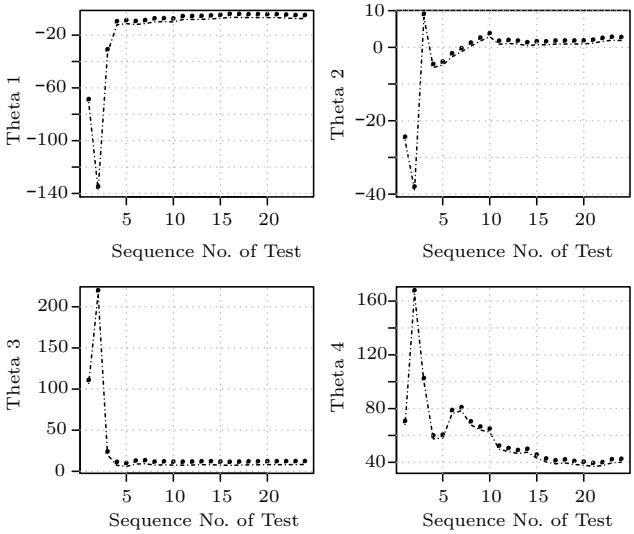

Fig.11. Adaptive changes of model parameters over time.

Nonetheless, the system is able to recover from the bad performance and works well over all measures after more Q&A data is taken into account in training the model. The four metrics are all above 80% during the last few iterations. This test scenario is similar to the practical application where we predicate the unknown sessions using current knowledge and train a new model based on the sessions after we manually label them.

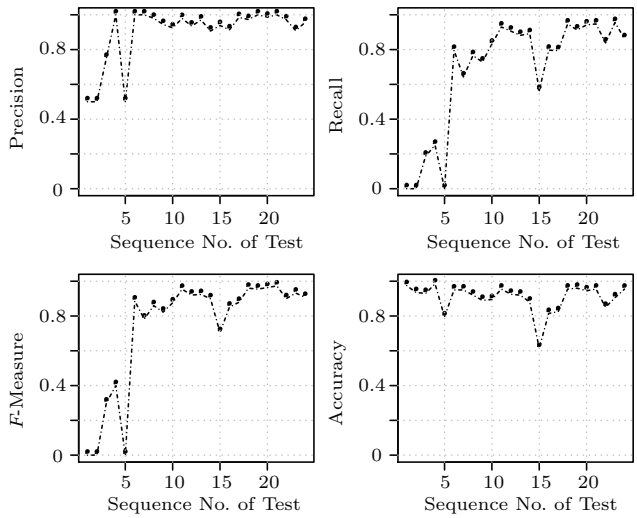

Fig.12. Performance of the adaptive detection system over time with manual labelling.

7.2 Adaptive Model Without Manual Labelling New Samples

To illustrate the advantage of manual labelling, we also perform the experiment in which we update the model using the predictions of new samples. We use 200 manually labelled samples as the initial training data and build a model. At each iteration, we test 200 new sessions using the model and insert the new samples associated with the predictions into the database. The results are shown in Fig.13.

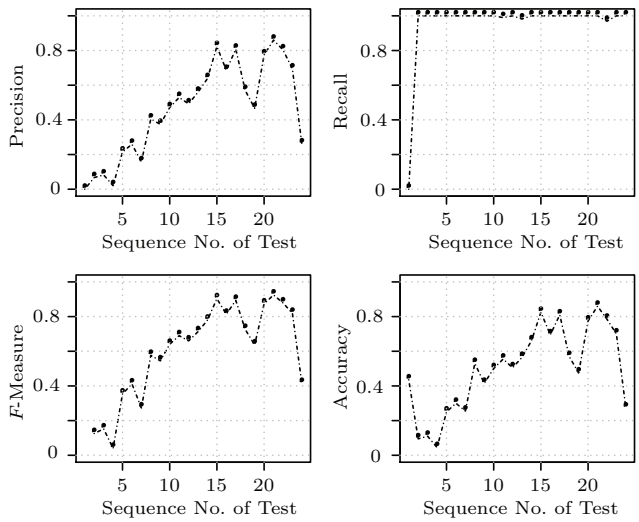

Fig.13. Performance of the adaptive detection system over time without manual labelling.

In Fig.13, we observe that the recall measure is surprisingly high; it achieves 1.0 for most iterations. It suggests that the false negative is very low. After checking its predictions of all iterations, we find that it outputs very few negative predictions, sometimes none at all. These predictions result in large performance fluctuation because the proportions of the campaign session in continual iterations are different. In addition, with more samples being used to train the model, the increasing trend in precision, F-measure and accuracy are not as stable as they are in Fig.12. Therefore, manually labelling new samples is still critical for accurate predication in practice.

7.3 Fixed Model

To illustrate the advantage of adaptivity with respect to accumulated samples, we test two types of the fixed model in which we use a fixed size training set to train the model.

7.3.1 Fixed Training Set

We use the first 200 samples as the initial training data and build a model. We fix the model parameters, and at each iteration, we test 200 new sessions using the fixed model. The results are shown in Fig.14.

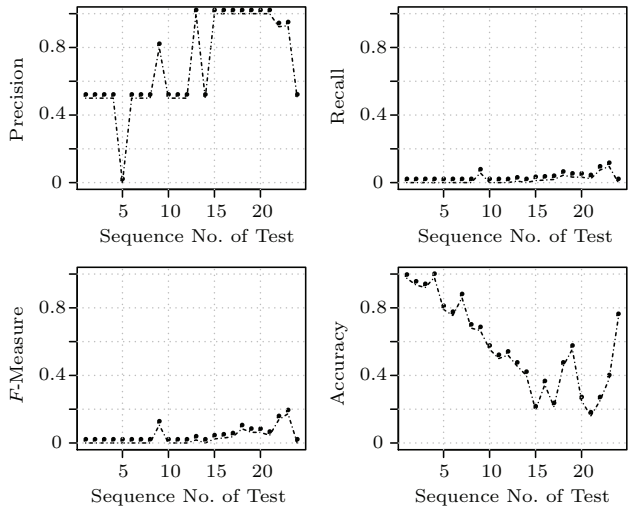

Fig.14. Performance of the fixed model.

Since the parameters of the fixed model are only trained on the first set of training samples, we do not draw the changes of the model's parameters. The precision in some tests is nearly 50% and it even becomes very high in a few tests from the 15th to the 20th iterations. However, compared with Fig.12, we note that the recall values are always very low. It means that the false negative is high. The low F-measure values confirm this

problem in the fixed model, i.e., the fixed model classifies many campaign Q&A sessions as the non-campaign sessions. Consequently, although the precision is high, other metrics indicate that the fixed model has obvious bias in classification. What is worse, this model cannot update itself by new samples because the parameters are only trained on the initial training dataset.

7.3.2 Moving Window of a Fixed Size

We use a moving window of a fixed size (200 samples) to train the model at each iteration. For example, at the first iteration, we train the model with samples $1 \sim 200$ and test it with samples $201 \sim 400$. At the second iteration, we train the model with samples $201 \sim 400$ and test it with samples $401 \sim 600$. We repeat this procedure for all iterations. The results are shown in Fig.15.

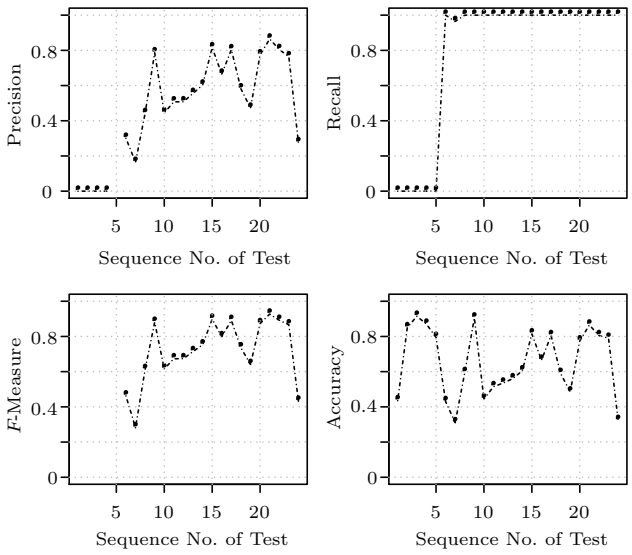

Fig.15. Performance of the fixed model with moving windows of a fixed size.

Fig.15 shows similar recall as Fig.13; it achieves 1.0 for most of iterations. Again, we check predictions of all iterations and find that the model outputs very few negative predictions, sometimes none at all. In addition, the performances of precision, F-measure and accuracy are not stable. Therefore, training by the moving window of a fixed size restricts the model's ability to adapt to new samples. The sample accumulation (quantity) is necessary to improve the performance of the detection system.

7.4 Experiments with Different Models Using Two More Advanced Classification Packages

As evaluated in Subsection 7.1, a logistic regression based method shows satisfactory overall performance when distinguishing campaign answers from non-campaign ones. To further examine the effectiveness of this method, we explore different linear classifiers as well as non-linear ones on our dataset and compare their prediction performance in Subsection 7.4. We perform experiments using two popular machine learning libraries, LIBSVM⑩[22] and LIBLINEAR[23].

LIBSVM is a general-purpose SVM solver, which supports kernel functions in order to train non-linear classifiers. On the other hand, LIBLINEAR is exclusively used for linear classification, i.e., it supports logistic regression and linear support vector machines. Without using kernels, LIBLINEAR can train a much larger set via a linear classifier. Consequently, LIBLINEAR is considered as a better choice over LIBSVM when handling large-scale datasets (e.g., document classification) for which using nonlinear mappings does not provide additional benefit.

Tables 3 and 4 list candidate models to be tested in the experiment.

Table 3. LIBSVM Kernel Types

Kernel Type	Description
t0	Linear: $\boldsymbol{u}^{\mathrm{T}} \cdot \boldsymbol{v}$
t1	Polynomial: $(\gamma \times \boldsymbol{u}^{\mathrm{T}} \cdot \boldsymbol{v} + coef0)^{degree}$
t2	Radial basis function (RBF): $\exp(-\gamma \times \lvert\boldsymbol{u} - \boldsymbol{v}\rvert^2)$
t3	Sigmoid: $\tanh(\gamma \times \boldsymbol{u}^{\mathrm{T}} \cdot \boldsymbol{v} + coef0)$

Table 4. LIBLINEAR Solver Types

Solver Type	Description
s0	L2-regularized logistic regression (primal)
s1	L2-regularized L2-loss support vector classification (dual)
s2	L2-regularized L2-loss support vector classification (primal)
s6	L1-regularized logistic regression

Fig.16 shows the running time of different models for LIBLINEAR and LIBSVM. The total time includes model training and sample prediction and it is evaluated on a sequence of 24 tests, as described in Section 7. We observe that both LIBLINEAR and LIBSVM are very fast; even the slowest model (LIBSVM

⑩http://www.csie.ntu.edu.tw/∼cjlin/libsvm, April 2015.

with t2) takes less than 2 seconds on all 24 iterations. In addition, LIBLINEAR is substantially faster than LIBSVM. The reasons for these observations are that we have a limited number of features for each sample and linear classifiers can be trained more efficiently than non-linear ones.

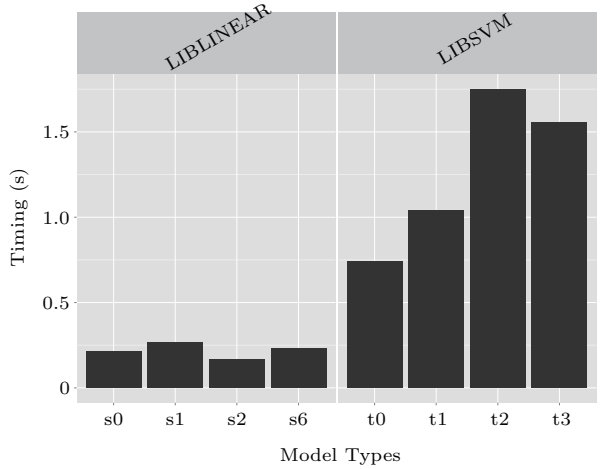

Fig.16. Timing of different model types for LIBLINEAR and LIBSVM.

We then show the performance metrics for the eight tests. As for LIBSVM, since the metrics of kernel types t0, t2 and t3 are similar, we only draw Fig.17 and Fig.18 for kernel types t1 and t2, respectively.

Since the performance curves for all models of LIBLINEAR are similar, we only draw a figure (Fig.19) for the solver type s2, which is also the fastest.

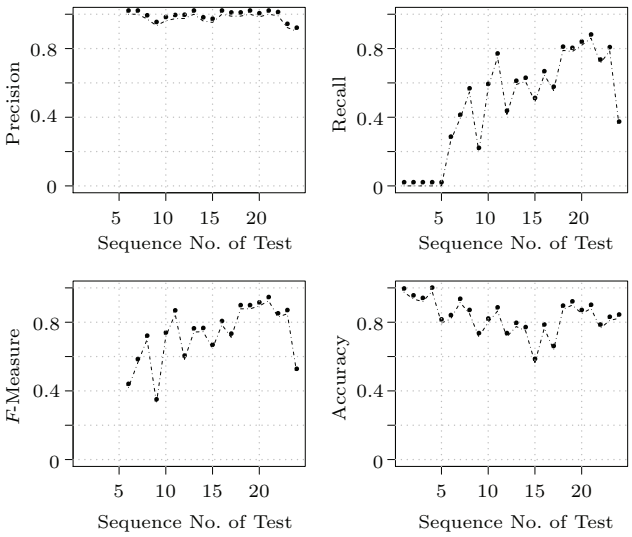

Fig.17. LIBSVM with polynomial kernel using default penalty and model parameters (t1).

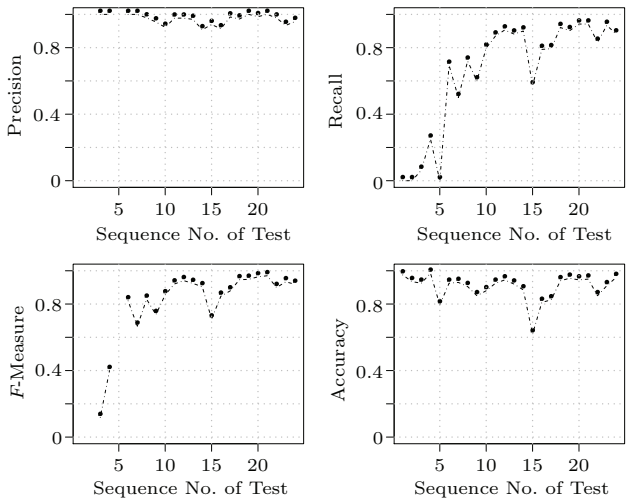

Fig.18. LIBSVM with RBF kernel using default penalty and model parameters (t2).

From Figs.17∼19, we see that the recall and the accuracy of LIBSVM with polynomial kernel are worse than those with RBF kernel. On the other hand, the precision of linear classifier trained by LIBLINEAR is not so stable as that of LIBSVM with RBF kernel. When it comes to the other three metrics, we observe similar trends for LIBLINEAR and LIBSVM with RBF kernel. Since LIBLINEAR takes much less time than LIBSVM, we can conclude that a linear classifier is more suitable for our detection problem. In addition, note that metrics of precision and F-measure in the three figures are not available for the first few tests (corresponding to the missing points in Figs.17∼19), because initially the models only return negative predictions.

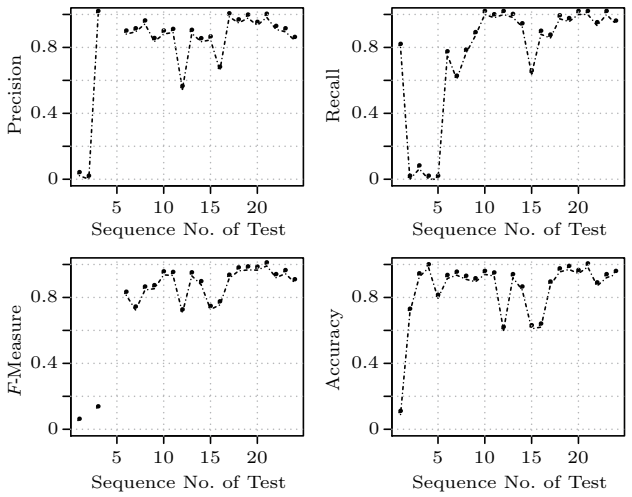

Fig.19. LIBLINEAR with L2-regularized L2-loss support vector classification (s2).

7.5 Non-Twisted Data Based Results

We have done some twists for users' features in Section 4. For example, to avoid zero probability, we specify 0.5 to $q1$ (the number of questions which have campaign answers for a specific user) when $q1 = 0$. If the system does not have enough information for a certain user (i.e., the total number of questions is less than 5), we set its $SGqID$ value to 0.5. In order to show the importance of twists, we train models based on raw data, i.e., with zero probability.

Fig.20 shows the performance metrics of this approach and it can be seen that fluctuation exists everywhere on performance curves. The overall performance is worse than that based on twisted data. It suggests that data correction should be considered before model training.

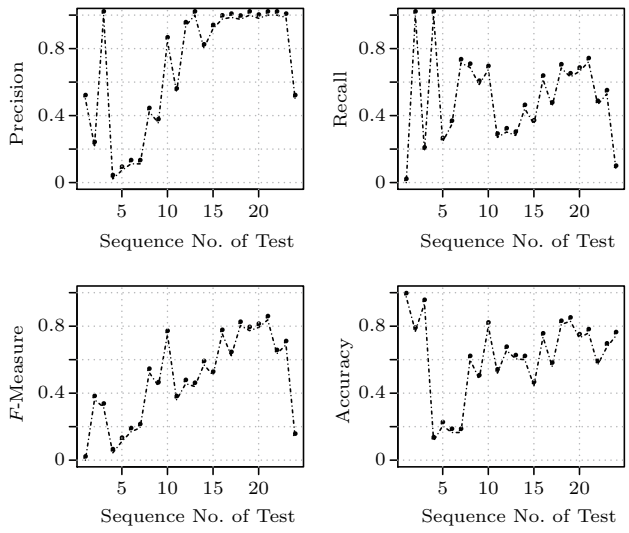

Fig.20. Performance metrics on features without data correction.

7.6 Experiments Using Only Text Information

We now only use text information (question, answers) for training the classifier. As a comparison to the previous method, we will use a typical information retrieval approach which consists of feature word selection, vectorization and classification.

7.6.1 Feature Selection

We use the Chi-square method[24-25] to retrieve a bag of feature words, a standard methodology of extracting features in documentation classification.

We define variables A, B, C and D in Table 5. For example, A is the number of campaign Q&As which have a specific word in the answers. D is the number of non-campaign Q&As which do not have the specific word.

Table 5. Chi-Square Feature Selection

Feature	Campaign	Non-Campaign	Total
With a specific word	A	B	$A+B$
Without a specific word	C	D	$C+D$
Total	$A+C$	$B+D$	N

After we collect the statistical information for every individual word, we can then compute Chi-square values. The Chi-square value of a word in the document collection is defined as

$$Chi\text{-}square(word, classification) = \frac{(A \times D - B \times C)^2}{(A+B) \times (C+D)}.$$

We compute the Chi-square value for each word in the training document collection, sort them in descending order and retrieve the first d words as the bag of the most predictive features.

7.6.2 Vectorization

After selecting feature words from the document collection, we can then vectorize each document by associating it with a vector of dimension d. To preserve consistency, we compute the weight for each dimension in the same way as how the spam grade value is calculated in Section 4. Note that for each iteration of the 24 tests, we re-extract the feature words and consequently re-compute the weight for each feature word.

7.6.3 Performance Evaluation

In order to show the impact of different dimensions, we use three settings for the following tests, i.e., $d = 100$, $d = 150$ and $d = 200$. We test LIBSVM with t2 and LIBLINEAR with s2. The results are shown in Fig.21 (LIBSVM) and Fig.22 (LIBLINEAR).

In Figs.21 and 22, different dimensions do not produce very different curves. With increased training samples, a model with higher dimension (200) is only slightly better than models with lower dimension (100 and 150). After the 15th iteration, recall and F-measure of LIBLINEAR are better than those of LIBSVM.

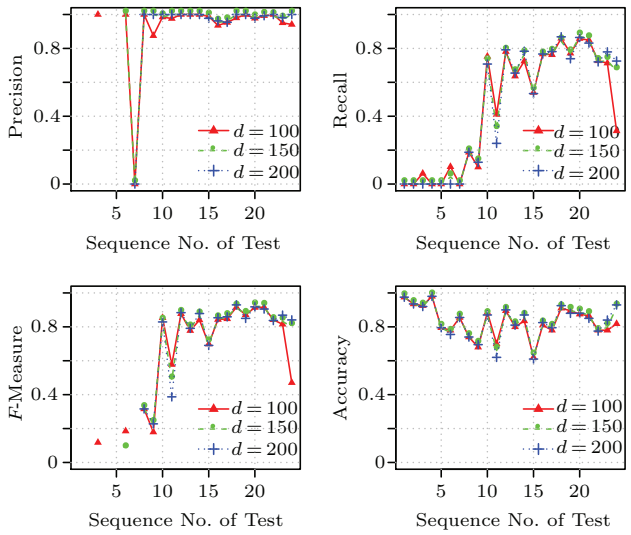

Fig.21. LIBSVM with RBF kernel (t2) using default penalty and model parameters.

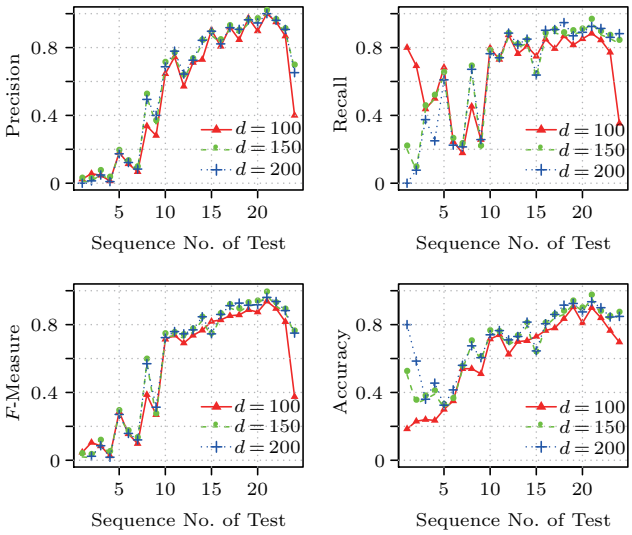

Fig.22. LIBLINEAR with L2-regularized L2-loss support vector classification (s2).

We now compare performance of text-only features (Fig.21 and Fig.22) to that of user-text features (Fig.18 and Fig.19). We list two main observations as follows.

• For LIBSVM, precision is very high using either set of features. Recall and F-measure are significantly improved using user-text features when the number of training samples is small, e.g., tests between the 5th and the 10th iteration. After the 15th iteration, recall and F-measure are also higher for user-text features ($0.9 \sim 1.0$), while text-only features vary from 0.8 to 0.9. In addition, user-text features provide more stable accuracy values than text-only features.

• For LIBLINEAR, we observe the significant improvement of the four metrics between the 5th and the 10th iteration by using user-text features. After the 15th iteration, user-text features lead to high values for all metrics that are close to 0.95. On the other hand, text-only features produce values around 0.9 which are slightly worse. Another interesting observation is that precision and F-measure of 200-dimension text-only features fall below 0.8 at the last iteration, while those with 100-dimension features drop to 0.4. In contrast, user-text features show robustness during the last iteration.

The overall performance of text-only features during the last few iterations is worse than that of user-text features. In addition, the number of features used in text-only approach (100, 150 and 200) is much more than that used in user-text method (3 as described in Section 4). This fact implies that the latter approach runs much faster while preserving high performance. To summarize, our proposed approach exhibits both effectiveness and efficiency, which are important factors in practice.

8 Conclusions

Detection of hidden campaigns can improve the user's experience when using current social websites. In this paper, we disclosed the behavior of a specific group of online paid posters who create commercial campaigns on the community Q&A websites. We collected real-world datasets and identified effective features to distinguish normal sessions and the campaigns. The performance of our classifier, with integrated statistic and semantic analysis, is promising in the real-world case study. Based on a learning technique, we also implemented a prototype of adaptive detection system which can retrieve the result in real time. The campaign scores and/or predicated labels can help users make better decisions when searching for answers on CQA portals and help the questioners select better answers as well.

References

[1] Jeon J, Croft W B, Lee J H, Park S. A framework to predict the quality of answers with non-textual features. In *Proc. the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, August 2006, pp.228-235.

[2] Jurczyk P, Agichtein E. Discovering authorities in question answer communities by using link analysis. In *Proc. the 16th*

ACM Conference on Information and Knowledge Management, November 2007, pp.919-922.

[3] Agichtein E, Castillo C, Donato D, Gionis A, Mishne G. Finding high-quality content in social media. In *Proc. the International Conference on Web Search and Web Data Mining*, February 2008, pp.183-194.

[4] Wang G, Wilson C, Zhao X, Zhu Y, Mohanlal M, Zheng H, Zhao B Y. Serf and turf: Crowdturfing for fun and profit. In *Proc. the 21st International Conference on World Wide Web*, April 2012, pp.679-688.

[5] Liu Y, Li S, Cao Y, Lin C Y, Han D, Yu Y. Understanding and summarizing answers in community-based question answering services. In *Proc. the 22nd International Conference on Computational Linguistics*, Volume 1, August 2008, pp.497-504.

[6] Bian J, Liu Y, Agichtein E, Zha H. Finding the right facts in the crowd: Factoid question answering over social media. In *Proc. the 17th International Conference on World Wide Web*, April 2008, pp.467-476.

[7] Bian J, Liu Y, Zhou D, Agichtein E, Zha H. Learning to recognize reliable users and content in social media with coupled mutual reinforcement. In *Proc. the 18th International Conference on World Wide Web*, April 2009, pp.51-60.

[8] Kleinberg J M. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 1999, 46(5): 604-632.

[9] Bian J, Liu Y, Agichtein E, Zha H. A few bad votes too many? Towards robust ranking in social media. In *Proc. the 4th International Workshop on Adversarial Information Retrieval on the Web*, April 2008, pp.53-60.

[10] Page L, Brin S, Motwani R, Winograd T. The pagerank citation ranking: Bringing order to the Web. Technical Report SIDL-WP-1999-0120, Stanford Digital Library Technologies Project, 1998.

[11] Pera M S, Ng Y. A community question-answering refinement system. In *Proc. the 22nd ACM Conference on Hypertext and Hypermedia*, June 2011, pp.251-260.

[12] Fichman P. A comparative assessment of answer quality on four question answering sites. *Journal of Information Science*, 2011, 37(5): 476-486.

[13] Sakai T, Ishikawa D, Kando N, Seki Y, Kuriyama K, Lin C. Using graded-relevance metrics for evaluating community QA answer selection. In *Proc. the 4th International Conference on Web Search and Web Data Mining*, February 2011, pp.187-196.

[14] Jindal N, Liu B. Opinion spam and analysis. In *Proc. the International Conference on Web Search and Web Data Mining*, February 2008, pp.219-230.

[15] Ott M, Choi Y, Cardie C, Hancock J T. Finding deceptive opinion spam by any stretch of the imagination. In *Proc. the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Volume 1, June 2011, pp.309-319.

[16] Mukherjee A, Liu B, Glance N S. Spotting fake reviewer groups in consumer reviews. In *Proc. the 21st International Conference on World Wide Web*, April 2012, pp.191-200.

[17] Huang M, Yang Y, Zhu X. Quality-biased ranking of short texts in microblogging services. In *Proc. the 5th International Joint Conference on Natural Language Processing*, November 2011, pp.373-382.

[18] Huang C, Jiang Q, Zhang Y. Detecting comment spam through content analysis. In *Proc. the 2010 International Conference on Web-Age Information Management*, July 2010, pp.222-233.

[19] Chen C, Wu K, Srinivasan V, Zhang X. Battling the Internet water army: Detection of hidden paid posters. In *Proc. the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, August 2013, pp.116-120.

[20] Kapur J, Kesavan H. Entropy Optimization Principles with Applications. Academic Press Inc., 1992.

[21] McFadden D. Conditional logit analysis of qualitative choice behavior. In *Frontiers in Econometrics*, Zarembka P(ed.), New York: Academic Press, 1974, pp.105-142.

[22] Chang C, Lin C. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2011, 2(3): 27:1-27:27.

[23] Fan R, Chang K, Hsieh C, Wang X, Lin C. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 2008, 9: 1871-1874.

[24] Zheng Z, Wu X, Srihari R. Feature selection for text categorization on imbalanced data. *ACM SIGKDD Explorations Newsletter*, 2004, 6(1): 80-89.

[25] Forman G. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research*, March 2003, 3: 1289-1305.

Cheng Chen received his B.S. degree in computer science from Beijing University of Posts and Telecommunications, Beijing, in 2010. He received his M.S. degree in computer science from the University of Victoria, Canada, in 2012, and he is currently a Ph.D. candidate there. His research interests are online social networks, recommender systems, and distributed algorithms for graph mining.

Kui Wu received his Ph.D. degree in computing science from the University of Alberta, Canada, in 2002. He joined the Department of Computer Science at the University of Victoria, Canada, in 2002, and is currently a professor there. His current research interests include big data, cloud computing, and network performance evaluation. He is a senior member of IEEE and a member of ACM.

Venkatesh Srinivasan is an associate professor in the Department of Computer Science at the University of Victoria. Prior to that, he was a postdoctoral fellow at the Max-Planck Institute for Informatics in Saarbrücken, Germany, Center for Discrete Mathematics and Theoretical Computer Science at Rutgers University in New Brunswick, US, and the Institute for Advanced Study in Princeton, US. He received his Ph.D. degree in computer science from the Tata Institute of Fundamental Research, Bombay. His research interests are in algorithms and complexity of computing.

Kesav Bharadwaj R is currently working as a software developer for the D. E. Shaw Group in Hyderabad, India. His work is focused on developing models used in valuation of various financial instruments. He completed his undergraduate program in computer science and M.S. degree in physics from the Birla Institute of Technology and Science — Pilani, India. He has done a research internship at the University of Victoria during which he worked on an online system for detecting commercial campaigns on CQA forums.