Qian Xu, Evan Wei Xiang, Qiang Yang, Jiachun Du and Jieping Zhnog. SMS Spam Detection using Non-content Features. To appear in IEEE Intelligent Systems, 2012, DOI: 10.1109/MIS.2012.3. Date of Publication Online: 17 January 2012.

SMS Spam Detection Using Non-Content Features

Qian Xu, Evan Wei Xiang and Qiang Yang
Department of Computer Science and Engineering
Hong Kong University of Science and Technology, Clearwater Bay, Kowloon, Hong Kong
{fleurxq,wxiang,qyang}@cse.ust.hk

Jiachun Du and Jieping Zhong
Huawei Technology Inc.
Shenzhen, Guangdong Province, China
{dujiachun,zhongjp}@huawei.com

Abstract

Short Message Service (SMS) text messages are indispensable in our lives today, but along with the convenience of using SMS messages in our daily lives, we also face a serious problem caused by SMS spamming. In this article, we explore a service-side solution to use graph data mining in finding out spammers from non-spammers. An important issue is whether we can detect spammers without checking into the contents of the messages; these are the content-less features. Using a real-world data set from a large telecommunications operator in China, we examine the effectiveness of various content-less features that range from network and to time-oriented categories. We find that some intuitively appealing features are in fact not very effective, whereas a combination of temporal and network features can be very useful in training high-performance classifiers for spammer detection.

Keywords: Spam Detection, SMS Spam, Social Media Spam, Data Mining.

1. Introduction

Short text messages (SMS) are an important means of communication today between millions of people around the world. SMS services, which are a must-have service nowadays for telecom operators, transmit their messages using standardized communications protocols [1]. At the same time, SMS messaging has become a perfect target for abuse via what are known as spamming - the misuse behavior of SMS to achieve some harmful purposes. Spamming is a serious problem for SMS today, as it is for emails and social networking services, because it disrupts people's daily life and harms well-being of telecom operators. As reported in [2], up to 30% of messages are recognized as spam in Asia, mainly due to the low cost of sending short messages.

Such a huge number of SMS spam seriously harms the confidence of telecommunications service users in their service providers. Thus, spam-filtering strategies have been tabled and tested around the world. In China, three major Chinese telecom operators, including China Mobile, China Telecom and China Unicom, have tried to impose limits on text messaging so that a given phone number can send no more than 200 messages per hour and no more than 1,000 messages per day on weekdays [3] [4]. In response, SMS spammers have also been adapting their spamming strategies in increasingly innovative ways. As a result, more effective approaches are in dire need to detect and filter SMS spam automatically and accurately.

In the past, various computational approaches, in particular data mining methods have been developed to detect email spam, and some have achieved a certain degree of success. Content-based approaches [5] are among the first to be applied. In email spam filtering, for example, content-based filtering methods consider the content-based features that can be used for classification. A spam email often contains some indicative keywords, such as "free", "awards", etc., or unusual distribution of punctuation marks and capital letters, such as "BUY!!", "MONEY" [6], such that these keywords become important features for a machine learning-based classification algorithm to use.

Due to the similarity between text documents in spam emails and spam SMS, content-based approaches in email spam detection research have been widely employed to detect SMS spam and spammers. In [7], Cormack et al., considered the problem of content-based spam filtering for short text messages that arise in three contexts: mobile SMS communication, blog comments, and email summary information. Huang et al. [8] used auxiliary information to boost the content-based approaches, who included mobile-station information of short messages based on the assumption that spam senders diffuse spam SMS at the same location. Zhang et al. [9] added additional meta-information to the characteristics of spam messages to the message contents, such as high sending frequency etc. A drawback of content-based spam-filtering methods is that they all require the contents of SMS messages, which are expensive or infeasible to obtain and can easily sacrifice the privacy of users. Moreover, SMS spammers often adapt the way they compose their contents through keywords, as we often see in email spam, where they may insert some special characters to escape the watch of spam filtering. These limitations of content-based spam detection techniques are a major bottleneck for making them more applicable in practice. Thus, we intend to find content-less methods for spam detection.

Based on the discussion above, we investigate ways to detect spam message senders based on non-content features that include temporal and graph-topology information but exclude contents, thus addressing user-privacy issues. We focus on the problem of identifying professional spammers based on the overall message-sending patterns. We consider professional spammers as those who have purchased a mobile communication ID, and whose sole purpose is to send large numbers of spam messages for commercial gain. Furthermore, we only concentrate on finding SMS spam on the server side, as the client-side detection is mostly content and ID based solutions, which is outside of our focus.

An earlier related work is attributed to Huang et al. [10], who proposed a complex-network based SMS filtering algorithm that compares an SMS network with a phone-calling communication network. Although such comparison can provide additional features, obtaining well-aligned phone-calling networks and SMS networks that can be aligned perfectly is difficult in practice. In this paper, we present an effective SMS anti-spam algorithm that only considers the SMS communication network. We first analyze characteristics of the SMS network, and then examine the properties of different sets of meta-features including static features, temporal features and network features. We incorporate these features into an SVM classification algorithm and evaluate its performance on a real SMS dataset and a video social network benchmark dataset. We also compare the SVM algorithm to a KNN based algorithm to reveal the advantages of the former. Our experimental results demonstrate that SVM based on network features can get 7%-8% AUC (Area Under the ROC Curve) improvement as compared to some other commonly used features.

2 Feature Extraction

The main dataset we consider is a realistic data from a Telco (telecommunications company) in China, which is also one of the largest telecommunications operators in the world. In this data, we have 4,900,468 SMS senders, The SMS dataset collected in seven days (25/03/2010 - 31/03/2010) from a province in China. In all, we have 3,589,661 legitimate senders and 1,310,592 unknown-type senders. Domain experts manually identified 215 spammers that serve as positive examples of spam messages. While the number of spammers is small, these spammers are very tedious and time-consuming to find out by humans. It also reflects the reality in industrial practices.

We first extracted features that characterize the messages and message senders from different perspectives (see Table 1). These features allow us to explore further the static, temporal and network features of SMS senders in detail. To build the full training set, we also randomly selected collections of 215 non-spammers that serve as ‘negative examples to pair with the spammers for our analysis.

2.1 Static Features

The number of messages

In the category of static features, we examine the number of messages within a time period as a property for describing a sender. The spammers usually send a large number of short messages simultaneously to make up for the cost, while normal users do not have such a pattern except for some special holidays such as Chinese New Years. Therefore, we attempt to explore whether the number of messages of a sender can be used as an important attribute to distinguish spammers from legitimate users. We plot the distribution of the number of messages in each day during seven days of a week in the test period for each category of users (spammers vs. non-spammers) in Figure 1(a). The X-axis indicates the percentage of senders and Y-axis indicates the total number of messages in each day during these seven days; the plot is in log-scale. As Figure 1(a) shows, spammers tend to send much more short messages than normal senders.

Message size

The message size of an SMS (including text, graphics, etc.) is a static feature that we can use. Figure 1(b) shows the distribution of the size of all messages during a seven-day period for each category of users (spammers vs. non-spammers). The X-axis indicates the percentage of senders considered and the Y-axis indicates the size of total messages during the seven days (in log-scale). In our analysis, we discover that the size of legitimate messages tends to be less than that of spam messages, perhaps because spammers often include plenty of information in a message to maximize the impact. However, with the knowledge of this feature, a spammer may start to randomize the size of the messages to avoid detection.

2.2 Temporal Features

Digital Object Indentifier 10.1109/MIS.2012.3 0885-9000/$26.00 2011 IEEE

Table 1: Feature Description

Feature set	Specific feature
Static features	Total number of messages for seven days Total size of messages for seven days Response ratio
Temporal features	Number of messages during a day, on each day of week from 1 to 7 Average number of messages in seven days Standard deviation of the number of messages for seven days Size of messages during a day, on each day of week from 1 to 7 Average size of messages for seven days Standard deviation of the size of messages for seven days Number of recipients during a day, on each day of week from 1 to 7 Average number of recipients for seven days Standard deviation of the number of recipients for seven days Average sending time for seven days Standard deviation of the sending time for seven days Average sending time gap for seven days Standard deviation of the sending time gap for seven days
Network features	Number of recipients Cluster coefficient

Number of messages during a day, on each day of week

In the category of temporal features, we consider the time-dependent information. For each day of the week, the number of messages sent within that day is calculated. Figure 1(c) plots the distribution of the number of messages in each day for each category of senders. In the figure, the X-axis shows days of a week, and the Y-axis shows the number of messages for the categories of messages (spam or non-spam). The spam messages are clustered in 25 to 300 messages range, whereas the number of messages for legitimate senders is less than 25.

Size of messages during a day, on each day of week

In the above section, Figure 1(b) shows the distribution of the size of total messages during seven days for each category of users. In this analysis, we try to examine the distribution of the size of messages in each day for each category of users. Figure 1(d) confirms our expectation that messages sent by legitimate senders can be biased towards smaller sizes, which leads to similar conclusions as we mentioned above.

Time-of-day

The time of day when the message was sent is another important feature in the category of temporal features. The intuition behind this feature is that the pattern of legitimate messages may be more evenly distributed than that of spam messages, in particularly in the daytime. The Figure 1(e) illustrates that in the daytime, spammers tend to send messages at several time slots in the period of 8am to noon and in the period of 6pm to 8pm, whereas legitimate senders tend to send messages at any time during the day-time. At night, spammers stop sending messages while a few of the legitimate users are also active.

2.3 Network Features

Number of recipients

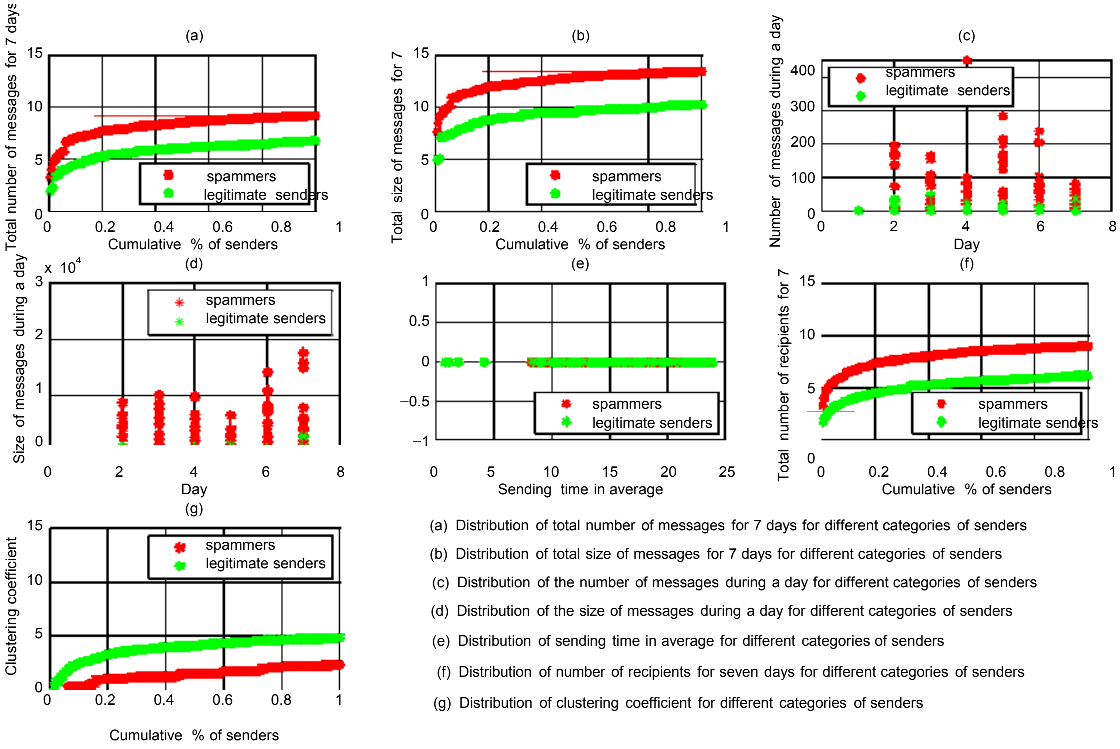
An SMS user functions in a network of all users. The number of recipients of an individual sender is an important feature in the category of network features, which can be reflected by the out-degree of a node in the SMS communication network. Our intuition is that spammers tend to send an invalid message to a large number of receivers simultaneously, where the receivers themselves do not know each other well, while normal users usually have a limited set of familiar persons. Figure 1(f) shows the distribution of the number of recipients for each category of senders, where the number of senders increases along the X-axis and the number of recipients increases along the Y-axis. Note that the Y-axis is on a log-scale to focus the plot on small values. Figure 1(f) confirms our intuition that the number of recipients of a spammer is clearly larger than that of legitimate users, since spammers aim at spreading their news or advertisements as widely as possible.

Clustering Coefficient

Among network features, we also consider clustering coefficient. Clustering coefficient is a property that measures the connectivity among the neighborhood of a node. If the neighborhood of a node is fully connected, the clustering coefficient is one. A value close to zero means that there are hardly any connections in the neighborhood. In an undirected network, the clustering co-efficient C_n of a node n is defined as $C_n = 2e_n/(k_n(k_n - 1))$, where k_n is the number of neighbors of n and e_n is the number of connected pairs between all neighbors of n. In the SMS

Digital Object Indentifier 10.1109/MIS.2012.3 0885-9000/$26.00 2011 IEEE

Figure 1: Data analysis and non-content-based features of a telecommunications dataset.



(a) Distribution of total number of messages for 7 days for different categories of senders

(b) Distribution of total size of messages for 7 days for different categories of senders

(c) Distribution of the number of messages during a day for different categories of senders

(d) Distribution of the size of messages during a day for different categories of senders

(e) Distribution of sending time in average for different categories of senders

(f) Distribution of number of recipients for seven days for different categories of senders

(g) Distribution of clustering coefficient for different categories of senders

communication network, we adopt clustering co- efficient as a network property measure. The intuitive idea is that the recipients of a legitimate sender may be friends as well with a high probability; however, spammers send messages randomly, thus their receivers do not contact with each other. We illustrate the distribution of clustering coefficient for each category of senders in Figure 1(g). The Y-axis indicates the values of clustering coefficient (log-scale) and the number of senders increases along the X-axis.

3 Classification Algorithms

Having discussed the various types of features associated with the SMS communication network, we now consider how to build classifier systems that can distinguish between spammers and non-spammers. We consider two very different algorithms: SVM and K-nearest neighbor (KNN) algorithms, as they represent different ways to exploit non-content features: while SVM pays attention to the margins and cases near the separating hyperplanes, KNN focuses on the 'typical' cases that represent positive and negative cases.

Support vector machines (SVMs) are popular classification algorithms. Thus, here we give a general description of SVMs only; interested readers can find technical details in [11]. A support vector machine (SVM) implements the following ideas: It maps the input vectors $\tilde{x} \in \mathrm{R}^{d}$ into a high dimensional feature space $\Phi(\tilde{x}) \in \mathrm{H}$ and constructs an optimal separating hyperplane, which maximizes the margin, i.e. the distance between the hyperplane and the nearest data points of each class in the space H. Different mappings give rise to different SVMs. A mapping $\Phi(\cdot)$ can be realized by a kernel function $\mathrm{K}(\tilde{x}_{\mathbf{i}}, \tilde{x}_{\mathbf{j}})$ which defines an inner product in the space H. The decision function implemented by SVM is:

$$f(\tilde{\mathbf{x}}) = \operatorname{sgn}\Big(\sum_{i=1}^{N} \mathbf{y}_{\mathrm{i}} \alpha_{\mathrm{i}} \cdot K(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}_{\mathrm{i}}) + \mathbf{b}\Big), \qquad (1)$$

where the coefficients $\boldsymbol{\alpha}_{\mathbf{i}}$ are obtained by solving the following convex Quadratic Programming (QP) problem:

$$\text{Maximize} \quad \sum_{i=1}^{N} \alpha_{\mathrm{i}} - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_{\mathrm{i}} \alpha_{\mathrm{j}} \cdot y_{\mathrm{i}} y_{\mathrm{j}} \cdot K(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}_{\mathrm{i}})$$

$$\text{subject to} \quad 0 \le \alpha_{\mathrm{i}} \le C \qquad (2)$$

$$\sum_{i=1}^{N} \alpha_{\mathrm{i}} y_{\mathrm{i}} = 0, \mathrm{i} = 1, 2, \ldots, N.$$

SVMs can reach global optimal when solving quadratic programming (QP) problems. They have been extended to handle large feature spaces and effectively avoid

Digital Object Indentifier 10.1109/MIS.2012.3 0885-9000/$26.00 2011 IEEE

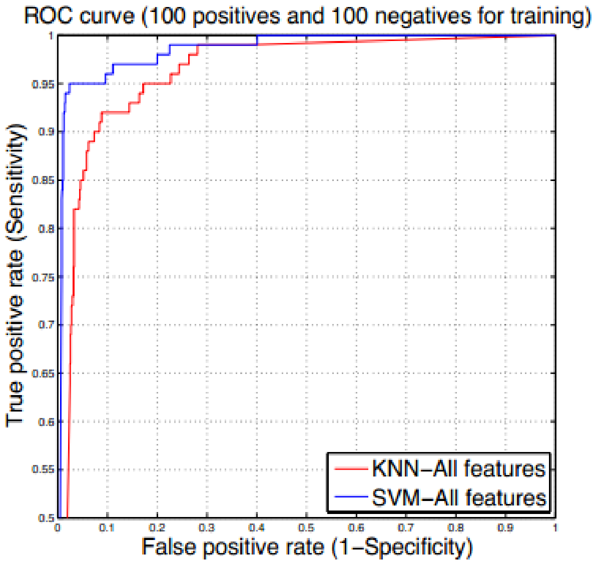
overfitting by controlling the classifier margins. In Equation (2), C is a regularization parameter that controls the tradeoff between margins and misclassification errors.

The KNN algorithms are built on the concept of distance between instances. For each test data instance, KNN first finds out the top K nearest neighbors according to the distance measure. It then finds a weighted majority class among the possible class labels. Weights may be introduced to reflect distances to the test instance. When used with the non-content features, this algorithm is easy to implement and can naturally incorporate network and temporal features. KNN is one of the most commonly used non-content feature algorithms in previous literature, and can thus be used as a baseline algorithm for comparison.

4 Experimental Results

In this section, we set out to evaluate the effectiveness along two dimensions. Along the feature dimension, we ask: which categories of features would give us the best performance in spammer detection? Along the algorithm dimension, we ask: which algorithms (SVM or KNN) should we use in the detection? We run tests on the telecommunications dataset as mentioned above, as well as on a benchmark dataset for spammer detection.

Figure 2: Comparing SVM and KNN on the Telco dataset.



4.1 Performance Measurement

For performance measurement, we use the area under the receiver operating characteristic (ROC) curve, which is denoted as AUC. Ranking based evaluation metrics are used increasingly in machine learning and data mining community when dealing with imbalanced data [12, 13]. When the data are imbalanced, cost-sensitive methods must be considered as well [14, 15]. The ROC measure plots the true positive rates (TPR or sensitivity) against the false positive rates (FPR= 1 − Specificity), where

$$TPR = \frac{True \quad positives}{True \quad positives + False \quad negatives}$$

$$FPR = \frac{False \quad positives}{False \quad positives + True \quad negatives}$$

In these equations, “positives” and “negatives” refer to the predicted class labels, while “True” and “False” denote the correctness of the predictions. TPR and FPR depend on the classifier function h and the threshold θ used to convert h(x) to a binary prediction. Varying the threshold θ from 0 to 1 changes the paired values of TPR and FPR, which gives the ROC curve. The area under the curve (AUC) indicates the performance of this classifier: the larger the area, the better is the Algorithm [16].

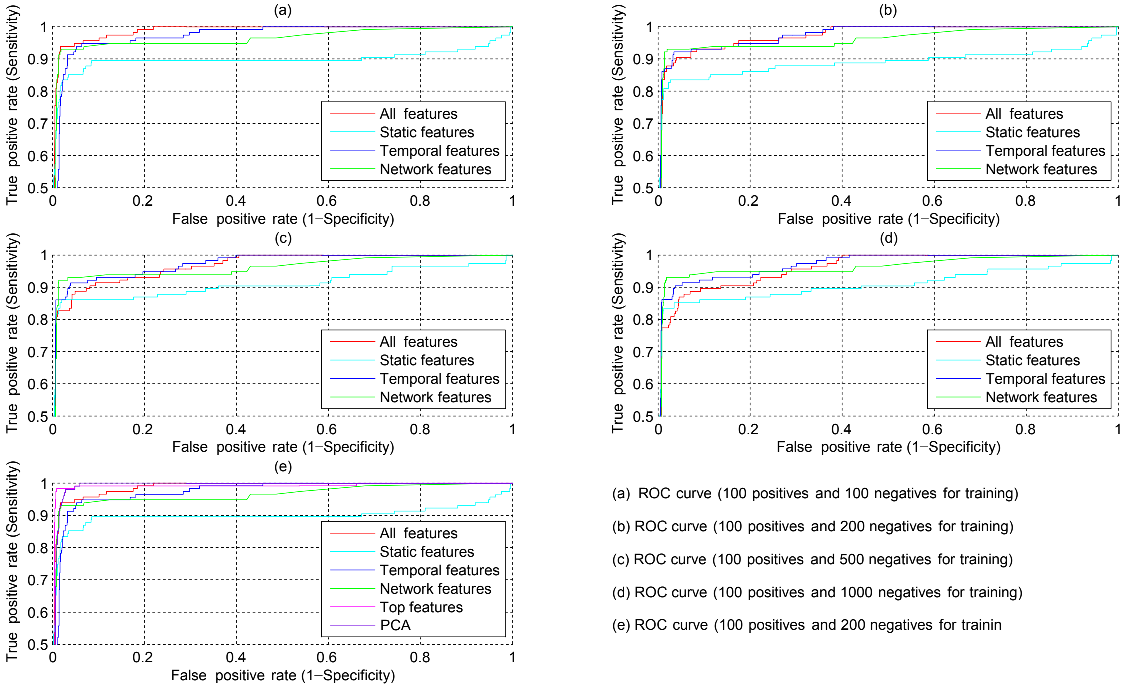
4.2 Comparison with a Baseline Method

In order to verify the effectiveness of our proposed method, in the first experiment, we compared SVM classifiers with K-nearest neighbor). In order to obtain the optimal parameter settings for SVM and KNN, we first tune the parameters C and K to achieve the optimal accuracy via ten-fold cross validation. For the SVM classifier, we use LIBLINEARSVM [17]. We randomly sampled 100 spammers as positive examples, and 100 legitimate senders as negative examples for training. For the testing dataset, we used 100 spammers as positive examples, and 1,000 legitimate senders as negative examples. Note that the testing dataset has no overlap with the training dataset. The results are shown in Figure 2, illustrating that SVM classifiers can achieve better performance in a comparison with KNN based on the same feature sets.

4.3 Comparison on Different Feature Sets

We randomly sampled 100 spammers as positive examples, and sampled ten, five, two, and one times of the spammers as negative examples for training, respectively. For the testing dataset, we also used 100 spammers as positive examples, and 1,000 legitimate senders as negative examples. As a result, we can get four results for various numbers of sampling data. The experimental results are shown in Figures 3(a), 3(b), 3(c) and 3(d), and explained in detail below.

Figure 3: Comparison of different feature categories.



(a) ROC curve (100 positives and 100 negatives for training)

(b) ROC curve (100 positives and 200 negatives for training)

(c) ROC curve (100 positives and 500 negatives for training)

(d) ROC curve (100 positives and 1000 negatives for training)

(e) ROC curve (100 positives and 200 negatives for trainin

We considered four sets of feature representations: 1) only static features; 2) only temporal features; 3) only network features; 4) combining static features, temporal features and network features together to get a set of "all features". The four experiments are designed to examine the contribution and importance of different sets of features. The experimental results show that if we only use the set of static features to train the classifier, we can reach a performance of AUC at 88.3%. However, if temporal features and network features are used individually, the AUC can get additional 7% and 8% improvements, respectively. These results indicate that, compared with static features, network properties and temporal information in an SMS communication network can indeed help achieve a better performance.

Furthermore, we find that the set of "all features", which combine static features, temporal features and network features together, results in almost the same improvement as using temporal features and network features alone. This is perhaps because some features are redundant and noisy, and can sometimes cause performance degradation. Therefore we employ principle component analysis (PCA) to extract informative features instead of using all features. It is also clear to us that a linear SVM can be used successfully to tell the relative importance of the features. Due to the space limitation, we list the top 13 of the extracted features in Table 2 according to the corresponding feature weights given by the SVM classifier.

Based on these observations, the clustering coefficient and SMS sending time are not informative in distinguishing spammers from legitimate senders. This is perhaps due to fact that the period of data collection covers seven days only. In such a short period, the neighborhood of a sender may not contact with each other. While the size of messages is an essential feature to classify spammers and non-spammers, in order to further examine the importance of various features, we distill the top 13 ranked features in Table 2, including static feature (1), temporal features (2-12) and network feature (13) as an SVM input and compare the results with PCA and other feature combinations in Figure 3(e), where features extracted by PCA and "top features" achieve similar results, and several most important features can lead to the best performance in comparison with other feature combinations.

Table 2: Ranking based on importance of features.

Rank	**Feature Description**
1	Total size of messages for seven days
2	Size of messages during 7th day
3	Standard deviation of messages sizes in seven days
4	Size of messages during 5th day
5	Average size of messages for seven days
6	Standard deviation of the sending time for seven days
7	Average sending time gap for seven days
8	Size of messages during 4th day
9	Size of messages during 6th day
10	Size of messages during 2th day
11	Size of messages during 3th day
12	Standard deviation of the sending time gap in seven days
13	Number of recipient

4.4 Additional Experiments on a Video Social Network Data

In order to further examine the feasibility and robustness, we run experiments on another real-world dataset that has been used to detect spammers in online video social networks [18]. This dataset has 8 static features, 8 network features and 2 temporal features. We randomly sampled 79 spammers as positive examples and 79 legitimate users as negative examples for training, and then sampled 78 spammers as positive examples and the remaining 562 normal users as negative examples for testing. Similarly, two experiments were conducted. The first experiment compares the AUC performance of KNN and SVM based on the same feature set, and the second one examines effects of different feature sets on AUC performance. The experimental results are shown in Figure 4.

We can see from Figure 4(a) that the SVM classifier has stronger ability to detect spammers in online video social networks compared to the KNN classifier. Moreover, in Figure 4(b), we confirm that temporal and network features can be incorporated into conventional static features to achieve better performance when detecting spammers. However, we find that temporal information in this data cannot lead to satisfactory performance, mainly because this dataset provides two temporal features only.

4.5 Summary

To sum up, in the first experiment, we compared SVM classifier with KNN classifier trained on the same feature set. The result is shown that SVM classifier outperforms when predicting spammers. In the subsequent experiment, SVM classifiers were trained using static, network, temporal features and their combinations. The experimental results illustrate that compared with static features, network properties and temporal information in an SMS communication network can indeed help achieve a better performance. Furthermore, if temporal features, network features and conventional static features are integrated, the performance improvement is similar as using temporal features and network features alone, perhaps due to redundant features. Therefore, some advanced feature selection methods can help improve the AUC performance further.

As we mentioned in the Introduction Section, existing spam filtering methods require the contents of SMS messages, which may sacrifice the privacy of users, or need to employ auxiliary information such as a calling network [10], which are expensive or infeasible to obtain; therefore we cannot compare with existing methods for SMS spammer detection.

5 Conclusion

In this paper, we have examined mobile-phone SMS message features from static, network and temporal views, and proposed an effective way to identify important features that can be used to construct an anti-spam algorithm. We exploited a temporal analysis to design features that can detect SMS spammers with both high performance, and incorporated these features into an SVM classification algorithm. Our evaluation on a real SMS dataset showed that the temporal features and network features can be effectively incorporated to build an SVM classifier, with a gain of around 8% in improvement on AUC, as compared with those that are only based on conventional static features.

In the future, we wish to extend out work in several directions. First, we will consider network evolutionary features such as how the network associated with a node changes with time in a certain time period. Second, we will consider meta-level features such as weekdays and weekends, originating sender locations of various SMS messages and so on. Finally, we will consider a wider range of tests by including more positive examples that are highly representative of the spammers as they evolve.

Acknowledgement

We thank the support of Hong Kong ITF project ITS/579/09. We also thank Lili Zhao for her help in our experiments.

References

[1] http://en.wikipedia.org/wiki/SMS

[2] http://en.wikipedia.org/wiki/Mobile_phone_spam.

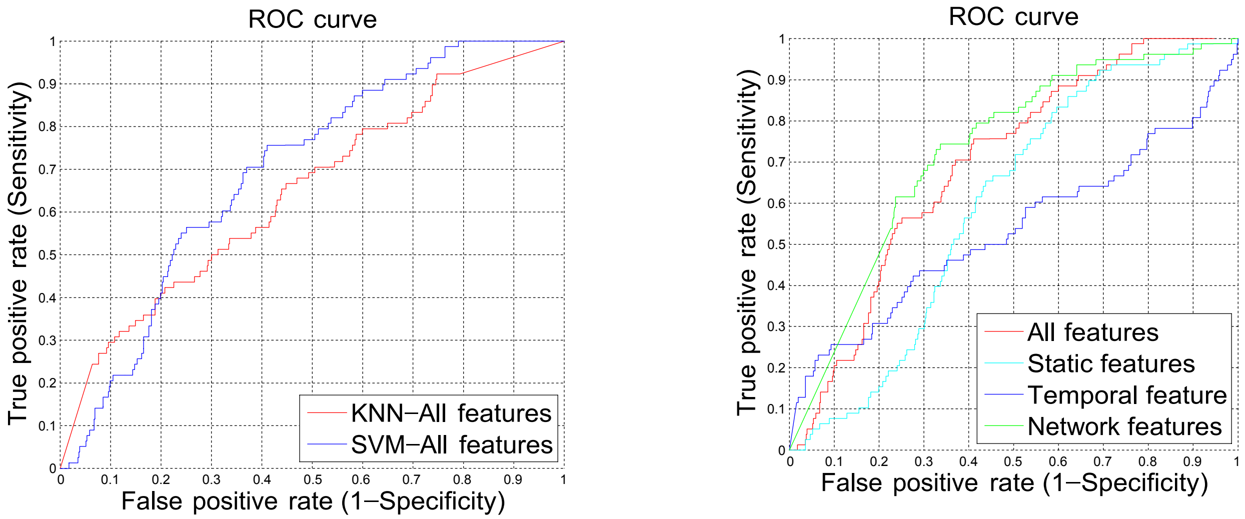
[3] Le Yu. China take steps to deal with SMS spam messages. Reuters 2009. http://www.reuters.com/article/2009/06/12/us-china-sms-idUSTRE55B1RU20090612.

[4] Wang Xing. You may get fewer spams on cell phones. China Daily 2009. Http://www.chinadaily.com.cn/china/2009-06/13/content_8280507.htm.

[5] Fabrizio Sebastiani. Machine learning in automated text categorization. ACM Computing Surveys, 34(1):1–47, 2002.

[6] José Bringas, Enrique Puertas Sánz, and Francisco Carrero García. Content based SMS spam filtering. In Proceedings of the 2006 ACM symposium on Document engineering, Pages 17-114. The Netherlands, October 2006.

Digital Object Indentifier 10.1109/MIS.2012.3 0885-9000/$26.00 2011 IEEE



(b) Effectiveness of different features for video dataset.

Figure 4: Results on the video dataset. (a) Comparing SVM and KNN on the video dataset.

[7] Gordon V. Cormack, José María Gómez Hidalgo, and Enrique Puertas Sánz. Spam filtering for short messages. In Proceedings of ACM CIKM 2007: 313-320, Portugal, November 2007.

[8] Huang Wenliang, Zhang Ni, and Dong Yutao. Spam SMS detection based on mobile terminal location and content. Mobile Communications (In Chinese), 13, 2008.

[9] Zhang Wei and Wang Zi-Xuan. GSM spam SMS filtering solution. Telecom Express: Networking and Communications (In Chinese), 3, 2009.

[10] Huang Wen-Liang, Liu Yong, Zhong Zhi-Qiang, and Shen Zhong-Ming. Complex network based SMS filtering algorithm. China Academic Journal Electronic Publishing House (In Chinese), 13, 2008.

[11] Christopher J. C. Burges. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery Journal, Springer, 2(2):121–167, 1998.

[12] Tom Fawcett. An introduction to ROC analysis. Pattern Recognition Letters, 27(8):861–874, 2006.

[13] Nathan Nan Liu and Qiang Yang: EigenRank: a ranking-oriented approach to collaborative filtering. In Proceedings of the ACM SIGIR 2008: 83-90, Singapore, 2008.

[14] Xiaoyong Chai, Lin Deng, Qiang Yang, and Charles X. Ling: Test-Cost Sensitive Naive Bayes Classification. In Proceedings of the IEEE ICDM 2004: 51-58, Brighton, UK, November 2004.

[15] Ke Wang, Senqiang Zhou, Qiang Yang and Jack Man Shun Yeung: Mining Customer Value: From Association Rules to Direct Marketing. In Data Mining and Knowledge Discovery Journal, Springer, 11(1): 57-79, 2005.

[16] Matthew G. Culver. Active learning to maximize area under the ROC curve. In Proceedings of the ICDM 2006: 149–158, Hong Kong, China, December 2006.

[17] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang- Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. In: Journal of Machine Learning Research, 9:1871–1874, 2008.

[18] Fabrício Benevenuto, Tiago Rodrigues, Virgílio Almeida, Jussara Almeida and Marcos Gonçalves. Detecting spammers and content promoters in online video social networks. In Proceedings of the ACM SIGIR 2009: 620-627, Boston, MA, USA, July 2009.

Digital Object Indentifier 10.1109/MIS.2012.3 0885-9000/$26.00 2011 IEEE