http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 7 (5): 790-795, 2008
ISSN 1812-5638

The Feature Updating Algorithm for Short Message Content Filtering

Qindong Sun, Hongli Qiao and Zuomin Luo
School of Computer Science and Engineering, Xi'an University of Technology, Xi'an, China

Abstract: This study presents a dynamic updating algorithm of adverse SMS feature library which is used to support the identification and filtration of adverse mobile short message content. It accesses new short message features by continuous computation and combines complete feature updating with partial feature updating to renew the adverse short message feature library. New library is used for filtering adverse short messages. Experimental results show that the adverse short message content filtering system based on the renewable adverse feature library has a stable performance and its evaluation criteria of F1 gets an average value of over 0.9.

Key words: Short message, content filtering, feature updating

INTRODUCTION

Gartner's report predicted that the count of global mobile Short Messages Service (SMS) messages sent in 2007 was about 1.9 trillion and nearly 30% of them were in China. There was a survey in China showing that 33.4% of mobile users have the experience of receiving pornographic SMS messages unexpectedly (Zhang, 2006). Mobile short messages with adverse information like pornography are usually sent to mobile terminals without mobile users' awareness and may annoy most of them. These messages are especially harmful to teenagers who are innocent of ill information. So it is important to deal with those adverse messages. Solutions of this problem require supports of both statutes and information filtering technologies. Many countries have issued laws and rules to judge and punish SMS spam transmission (Zheng, 2006), researchers have also paid more attention to research on SMS contents' identification and filtering technologies with the purpose to prevent, or to reduce, the spreading of adverse SMS messages.

Content based SMS message filtering technologies are hotspots of relating researches. Hidalgo *et al.* (2006) did some content filtering experiments based on machine learning algorithm with English and Spanish SMS spam corpuses to prove that Bayesian filtering technologies are efficient to SMS spam. Deng and Peng (2006) designed a distributed SMS content filtering system based on Bayesian techniques considering other SMS features, such as the fact that length of SMS spam is usually larger than most ordinary short messages, to improve the performance of SMS filter. Besides, many other studies on content filtration of texts (Lynam *et al.*, 2006; Pang *et al.*, 2007), data streams (Sculley and Wachman, 2007; Liu *et al.*, 2008), web pages (Chau and Chen, 2008; Lin *et al.*, 2008) and email spam shown in some studies (Cormack and Lynam, 2007; Dong *et al.*, 2006; Guo *et al.*, 2006; Hsiao and Chang, 2007) are also helpful to research on SMS spam filtering technologies.

Information content processing technologies mostly require the support of certain content feature library. The performance of adverse SMS content filter especially depends on the quality of adverse SMS content feature library. Cormack *et al.* (2007) did experiments using top performing email spam filters on mobile spam messages and proved that feature engineering is more critical for mobile spam filtering than for email filtering. Because the content of SMS message streams are changing with time, its feature library should also be updated accordingly to insure the stable efficiency of adverse SMS message filter. Existing studies specifically aimed at SMS content feature library updating are still far from enough now. Lanquillon and Renz (1999) studied on user interest based information filtering technology and about tracking of content change without feedback, but it didn't involve the detection on SMS content alteration. Study on SMS spam filtering (Deng and Peng, 2006) has spoken of self study and knowledge updating abilities of this filtering system, but it is lack of detailed algorithm and corresponding experimental data.

Considering the fact that SMS messages are short in length and their content features are sparse, this study makes some modifications on TF-IDF term weighting formula which is commonly used for texts classification, to distinguish importance of different adverse SMS content keywords. After the basic job of feature selection, this study provides a detailed updating algorithm on adverse SMS message feature library so that the adverse SMS filtering system supported by the library runs smoothly.

Corresponding Author: Qindong Sun, School of Computer Science and Engineering, Xi'an University of Technology, Xi'an, China

THE FRAMEWORK OF ADVERSE SMS FILTER

SMS content filtering system deploys between Short Message Service Center (SMSC) and Internet Short Message Gateway (ISMG). With the support of adverse SMS feature library and certain filtering algorithm, it figures out adverse SMS messages in the SMS stream and leaches them if required. Distinguishing adverse SMS messages from harmless SMS messages demands high accuracy to reduce cases of misjudgment. On the other hand, that recall rate must be high is also important, because it means few adverse SMS messages survived while the filtering system is running. The content of SMS messages' stream is changing with time so the content of adverse SMS messages is not fixed either. Content filtration based on settled adverse SMS feature library can't meet the need of practical application and thus adverse SMS feature library needs renewal.

Feature updating is a process of periodic machine learning and progressive circulation. The successive adverse SMS feature updating constitutes a time related feature library-update-chain. New features come from learning and computation to new adverse SMS message corpus composed of adverse short messages. New adverse SMS message corpus comes from the continuous running of SMS filtering system that distinguishes adverse SMS messages from others and the SMS filtering system runs smoothly by the support of a perfect renewable SMS message feature library.

As is shown in Fig. 1, the initial adverse SMS message corpus and ordinary SMS message corpus without adverse short messages are set up through manual selection. The initial adverse SMS feature library is then established by feature selection and feature weight computation on these corpuses. Subsequently, the initial feature library works as Running Feature Library to help SMS Content Filtering Module to filter adverse SMS messages. When the time trigger is excited, the Updating Feature Library which has been renovated will turn to Running Feature Library to support the SMS filtering system while the former Running Feature Library will be new Updating Feature Library waiting to be refreshed next time.

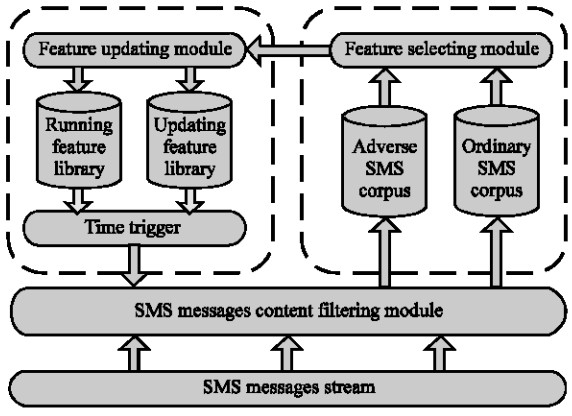

Fig. 1: The framework of adverse SMS filtering system

EXTRACTION OF SMS FEATURES

Formalization of SMS content: Statistical machine learning requires much computation, but SMS message texts are composed of characters and thus can't be computed directly. So, the first step is to convert the character space of SMS message text content into numerical space. The most commonly used numerical expression of text content is VSM. We can view each sentence as a collection of single words. Different words have different importance in the SMS message content, so they usually need to be weighted. Suppose feature space is $F = \{f_1, f_2, \ldots \ldots, f_n\}$ where $f_i = <T_i, W_i>$. T_i is the keyword whose serial number is i and W_i is its weight. Then each SMS message SM can be formalized as follows:

$$SM = \{f_1, f_2, \ldots \ldots, f_n\} \quad (1)$$

As Chinese text is made up of characters but not separate words like English, each SMS message will be cut into independent words at first. Some words like conjunctions, auxiliary words and interjections are meaningless, so they are canceled. Keywords come from content words left through certain feature selection algorithm and each of them should be weighted. One keyword and its weight compose a feature. The set of all features constitutes the vector space and each SMS message can be mapped to this space as a single vector. This is the representation of SMS content with numeric vector.

Selection of adverse SMS keywords: This study applies supervised machine learning to access adverse SMS content keywords. There are two types of learning corpus, one is negative SMS corpus C_n which is full of adverse SMS messages and the other is positive SMS corpus C_p which is made up of ordinary SMS messages without adverse content. There are many calculation methods to select SMS message keywords such as Document Frequency, Term Frequency, Information Gain, Mutual Information and χ^2 value. χ^2 is one of the commonly used methods. This value, when used to

extract adverse SMS features, can express the dependence of negative SMS content with a certain member word.

$$\chi^2(t,C_n)=\frac{\left[P(t,C_n)P(\bar{t},\overline{C_n})-P(t,\overline{C_n})P(\bar{t},C_n)\right]^2}{P(t)P(\bar{t})P(C_n)P(\overline{C_n})} \quad (2)$$

where, $P(t, C_n)$ is the probability that SMS messages with word t appear in corpus C_n while $P(\bar{t}, C_n)$ is the probability that SMS messages without word t appear in corpus C_n. $P(\bar{t},\overline{C_n})$ is the probability of SMS messages without word t existing in corpus C_p while $P(t,\overline{C_n})$ is the probability of SMS messages with word t being in corpus C_p. P(t) is the proportion of SMS messages with word t in both corpuses and $P(C_n)$ is the percentage that SMS messages in C_n takes in both corpuses. So $\chi^2(t, C_n)$ expresses the relationship between word t and class C_n. The closer their relationship is, the more prominent t's effect is when it worked as a keyword to express adverse SMS contents.

When χ^2 value of each single word from SMS messages is computed, they are arranged in descendent order. Those whose values take the frontal S stations in the array $T_1, T_2, \ldots, T_s, \ldots$ are selected as keywords.

Weight computation of SMS messages' keywords: Different adverse SMS keywords have different adverse degree, so each of them must be weighted. There are many ways to compute word weight in Text Data Mining technology. Boolean weight, term frequency and document frequency are rarely used because they are too rough. TF-IDF is more widely used and it performs better. TFC and LTC are converted from TF-IDF for improvement. Unlike conventional TF-IDF method, LTC formula computes term frequency with logarithm and thus impairs the negative effect from word frequency difference. It adapts to the computation of SMS message keywords' weight in this study

$$LTC(T_i)=\frac{\log(TF(i,C)+1.0)\log(N/n_i)}{\sqrt{\sum_{j=1}^{s}\left[\log(TF(j,C)+1.0)\log(N/n_j)\right]^2}} \quad (3)$$

Ordinary LTC formula applies in text classification to computes keywords' weight in a single text which is to be labeled as a member of certain class. But here short messages are too short in length, so that weight of keyword in a single short message is meaningless. So we prefer to weight keywords in the cluster of short messages of a single corpus. Here, T(i, C) is the frequency of SMS messages' keyword T_i in the corpus C, N is the total of SMS messages in both learning corpuses, n_i is the number of SMS messages with keyword T_i and S is the quantity of SMS message's keywords. In this study we want to compute the weight of adverse SMS keywords, so corpus C is C_n here. This formula implies that if a keyword has higher frequency in C_n and shows centralized distribution (This means that the keyword appears in a stable part of the corpus but not all), it should be given more weight.

In formula (3) the factor log (N/n_i) emphasizes the centralized distribution of SMS message keywords, but this is not specific. Weight of adverse words should insist that this keyword is densely distributed in negative SMS corpus and rare appears in the positive one. So this study adds a weight adjusting factor:

$$\alpha(T_i)=\sqrt{\frac{M_i}{N_i}} \quad (4)$$

where, M_i is the number of SMS messages containing adverse keyword T_i in corpus C_n and N_i is the number of SMS messages containing adverse keyword T_i in both corpuses. $N_i \geq M_i$. Then the weight formula of adverse SMS messages' keyword T_i becomes

$$W(T_i)=LTC(T_i)\alpha(T_i) \quad (5)$$

Each adverse SMS content feature is composed of a single adverse SMS messages' keyword and its weight. The collection of all adverse SMS content features is the adverse SMS feature library which can be used to support the SMS filtering system.

UPDATING ALGORITHM OF ADVERSE SMS FEATURE LIBRARY

The content filtering module and the feature updating module should run in parallel to collaborate in the SMS filtering system, so time span of updating should be defined.

Definition 1

Feature updating cycle: New adverse SMS features are computed from SMS corpuses caught by SMS filtering system in a fixed time interval. This time interval is defined as the feature updating cycle. It is labeled as T_{renew}.

The updating feature library is labeled as L_{old} and the collection of new adverse SMS features learned in new negative corpus through computation with Eq. 2 is represented with S_{new}, $S_{new} = \{<T_i, W_i> | 1 \leq i \leq n\}$. We mark the feature library after feature updating which will work as the running feature library as L_{new}. There are two different tactics to update the SMS feature library form L_{old} to L_{new} with S_{new}. The first way is to renew it completely. S_{new} replaces the current Running Feature Library (L_{old})

and becomes L_{new} while the L_{old} is eliminated: $L_{new} = S_{new}$. This strategy is favorable to new adverse SMS features and may better recognize new adverse SMS contents. But it completely ignores the function of old features. The changing of SMS content is gradual but not abrupt; some old features are still useful. So we adopt the second tactic of partly feature library updating in this paper. In this method, L_{new} will be made up of two parts after updating: $L_{new} = A \cup B; A \subset L_{old}; B \subset S_{new}$.

Suppose that N_{max} is the fixed size of adverse SMS feature library and N_{new} is the size of S_{new}. To update L_{old} partly, it must be true that $N_{new} < N_{max}$.

Definition 2

Completely new feature: In the newly accessed feature set S_{new}, features whose keyword part can't be found in L_{old} are called completely new features.

Definition 3

Reserved feature: In the newly accessed feature set S_{new}, features whose keywords are found in L_{old} are called reserved features.

The adverse SMS feature library updating algorithm, which adopts the idea of eliminating low weight features and updating weight of retained features, is as follows:

Step 1: Get out all pairs of $<T_i, W_i>$ from current feature library L_{old} to constitute the collection of features to be updated and label it as S_{old}.

Step 2: Arrange features of S_{old} in weight-descending order to get the list of adverse SMS features list-old. The size of this list is N_{max} too. New features of S_{new} are saved to a new array named List-new which has the size N_{new}.

Step 3: Set a counter N_x ($N_x = N_{new}$). If $N_x \neq 0$, fetch the head element of list-new, $V_{head} = <T_{head}, W_{head}>$ and search it from List-old; if $N_x = 0$, that is when list-new is empty, go to Step 5.

Step 4: If an element V_x ($T_x = T_{head}$) were found in list-old, we would get a reserved feature. Then we update the weight of this feature in list-old: $W_x = W_{head}$, move V_x from its original position to the head of list-old to avoid being replaced by completely new features later, eliminate V_{head} from list-new and reduce N_x by 1 then go back to Step 3. If no element meet the condition of V_x ($T_x = T_{head}$) were found, we reduce N_x by 1 and go back to Step 3 directly.

Step 5: Count the quantity of new SMS features left in list-new and mark it as N_{left} ($N_{left} \leq N_{new}$). These features are completely new features. Use those features to replace the last N_{left} features in list-old completely and we will get the new list-old.

Step 6: Transmit all features from the new list-old into running feature library and that is the renewed adverse SMS feature library L_{new} we get at last.

This algorithm classifies features to be updated into two different types, some of them have identical keywords with those in the running feature library. These words are still useful to represent adverse SMS messages. We retain those words and update their weight to form new features. Others are completely new words learned from recent corpuses, which should be added into the running feature library entirely and thus some of old features in the running feature library should be washed out. We prefer those with high weights and thus some low weight features are eliminated.

EXPERIMENTAL RESULTS AND ANALYSIS

This study makes use of machine learning method to extract adverse SMS keywords that constitute adverse SMS feature library supporting SMS filtering system and updates the adverse feature library to keep a stable performance of the adverse SMS filtering system. The corresponding experiments simulate adverse SMS feature library updating process of the SMS filtering system. For the sake of experimental simplicity, our filtering target of adverse SMS messages is special in adult content. We use precision, recall and F1 measure of adult SMS messages' recognition by SMS content filtering system as evaluation parameters. Experiments were done in two ways: experiment on the open test corpus and that on the closed test corpus. Through these experiments we can observe the course of adult SMS feature library updating and changes of parameters above. It can help us to evaluate the performance of SMS feature library updating algorithm and find its problems.

Experiments were carried out on corpuses group by group. Each group was made up of a negative corpus and a positive corpus. The negative corpus contains 300 adult SMS messages while the positive one contains 1500 ordinary SMS messages without adult contents. We have prepared 8 groups of corpuses like that. Because the initial process of machine learning and filtering test cost two groups of corpuses, there are 6 times of updating experiments. The fixed size of feature library N_{max} is 1000 and the number of new features is fixed on 250: $N_{new} = 0.25\, N_{max}$. The composing of new feature set is shown in Table 1.

According to the statistical result of new feature sets, each time there are 250 new features in which complete new features take a part of about 15 to 25%. So we can see that features with new keywords have accounted for a considerable proportion and need special attention. But complete new features are not in the dominating position in new feature sets, so the method of updating SMS

Table 1: Changing of new adult SMS feature set in updating process

Experimental procedure	No. of complete new features	No. of retained features	Ratio of complete new features
The first updating process	36	214	0.144
The second updating process	40	210	0.160
The third updating process	58	192	0.232
The forth updating process	52	198	0.208
The fifth updating process	50	200	0.200
The sixth updating process	61	189	0.244

Table 2: Performance of system during feature updating process

Experimental procedure	Experimental type	Precision	Recall	F1 measure
The initial learning process	Closed test	0.9867	0.99	0.9883
	Open test	0.9734	0.9767	0.9750
The first updating process	Closed test	0.9966	0.9867	0.9917
	Open test	0.9722	0.9333	0.9524
The second updating process	Closed test	0.9898	0.9733	0.9815
	Open test	0.9716	0.9133	0.9416
The third updating process	Closed test	0.9932	0.98	0.9866
	Open test	0.9822	0.92	0.95
The forth updating process	Closed test	0.9833	0.9867	0.9850
	Open test	0.9811	0.8667	0.9203
The fifth updating process	Closed test	0.9967	0.9967	0.9967
	Open test	0.9751	0.7833	0.8687
The sixth updating process	Closed test	0.9966	0.99	0.9933
	Open test	0.9382	0.81	0.8694

feature library completely is not required. This result has also validated that this study takes the idea of updating SMS feature library with new features combined with old features is reasonable.

We adopt the precision, recall and F1 measure as parameters to estimate the stability of the SMS filtering system's performance during the adverse SMS feature library updating process. The statistical result is listed in Table 2 as follows,

We can see from the experimental data that during the feature library updating process and in open tests, the experimental adverse SMS message filtering system gets the good performance of over 0.9 in average F1 measure and precision values are always higher then 90% too. It shows that the feature updating algorithm performs well in the application of SMS content identification and filtration. But in the last two groups, the F1 measures are lower than that of other groups, it is obviously caused by the decline of recall parameter. In negative SMS corpuses of the last two groups there are more vague adverse short messages which are difficult to recognize, it is the main reason of recall drop. So algorithm in this paper is not good enough to extract or update features of obscure adverse SMS messages. Further improvement is required.

CONCLUSIONS

Adverse SMS messages and SMS spam are making more and more trouble for mobile users, so content based SMS filtering technology has obtained widespread concern. This study has engaged in the problem of SMS message content feature renovation for the use of SMS message filtering application. It brings forward a detailed SMS content feature library updating algorithm. This algorithm embodies the thought of integration and collaboration of old features and new features. Updating process adopts the method of retaining high weight features and eliminating low weight features. We have validated this algorithm through experiments on closed test corpuses and open test corpuses. Experimental result displayed a stable performance of SMS filtering system using this algorithm and the average F1 measure is above 0.9. Theory of this algorithm is also meaningful in Internet information filtering applications. This algorithm didn't perform well in vague adverse SMS filtering, so the next problem to be solved is the feature expression, extraction and refreshment of ambiguous adverse SMS content.

ACKNOWLEDGMENT

The research presented in this paper is supported by the Science Foundation of China Postdoctor (Grant No. 20070410379); the Natural Science Foundation of Shaanxi Province (Grant No. 2007F13).

REFERENCES

Chau, M. and H. Chen, 2008. A machine learning approach to web page filtering using content and structure analysis. J. Decision Support Syst., 44: 482-494.

Cormack, G.V. and T.R. Lynam, 2007. Online supervised spam filter evaluation. J. ACM Trans. Inform. Syst. (TOIS), 25: 1-11.

Cormack, G.V., J.M.G. Hidalgo and E.P. Sanz, 2007. Feature engineering for mobile (SMS) spam filtering. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'07), 23-27 July, ACM, New York, USA., pp: 871-872.

Deng, W.W. and H. Peng, 2006. Research on a naive bayesian based short message filtering system. 2006 International Conference on Machine Learning and Cybernetics. 12-16 August, Dalian, China, pp: 1233-1237.

Dong, J.S., H.X. Cao, P.P. Liu and L. Ren, 2006. Bayesian Chinese spam filter based on crossed N-gram. Proceedings of the 6th International Conference on Intelligent Systems Design and Applications (ISDA'06), 16-18 October, USA., pp: 103-108.

Guo, Y.H., Y.L. Zhang, J.Y. Liu and C. Wang, 2006. Research on the comprehensive anti-spam filter. 2006 IEEE International Conference on Industrial Informatics, August 2006, USA., pp: 1069-1074.

Hidalgo, J.M.G., G.C. Bringas, E.P. Sánz, P.G. Enrique and C. Francisco, 2006. Content based SMS spam filtering. 2006 ACM Symposium on Document Engineering (DocEng 2006), 10-13 October, Amsterdam, Netherlands, pp: 107-114.

Hsiao, W.F. and T.M. Chang, 2007. An incremental cluster-based approach to spam filtering. J. Expert Syst. Appl., 34: 1599-1608.

Lanquillon, C. and I. Renz, 1999. Adaptive information filtering: Detecting changes in text streams. Conference on Information and Knowledge Management (CIKM-1999), 2-6 November, USA., pp: 538-544.

Lin, P.C., M.D. Liu, Y.D. Lin and Y.C. Lai, 2008. Accelerating web content filtering by the early decision algorithm. J. IEICE Trans. Inform. Syst., E91-D: 251-257.

Liu, Y.B., J.R. Cai, J. Yin and A.W.C. Fu, 2008. Clustering text data streams. J. Comput. Sci. Technol., 23: 112-128.

Lynam, T.R., G.V. Cormack and D.R. Cheriton, 2006. On-line spam filter fusion. Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 6-11 August USA., pp: 123-130.

Pang, X.L., Y.Q. Feng and W. Jiang, 2007. A spam filter approach with the improved machine learning technology. Proceedings of the 3rd International Conference on Natural Computation (ICNC 2007), 24-27 August, USA., pp: 484-488.

Sculley, D. and G.M. Wachman, 2007. Relaxed Online SVMs in the TREC Spam Filtering Track. http://www.eecs.tufts.edu/~dsculley/papers/trec.2007.spam.pdf.

Zhang, X.P., 2006. SMS spam worries four hundred million mobile users. J. CWEEK, 35: 70-70.

Zheng, S.R., 2006. Punish SMS spam: foreign experiences. J. Digital Commun. World, 5: 24-26.