International Journal of Security and Its Applications
Vol. 10, No. 9 (2016), pp.45-54
http://dx.doi.org/10.14257/ijsia.2016.10.9.06

Implementation of Bad Information Filtering System Based on SVM Algorithm

Xiao-Lan Xie[1,2] and Zhen Long[3*]

[1]*College of Information Science and Engineering, Guilin University Of Technology, Guilin ,Guangxi Zhuang Autonomous Region, China, 541000*
[2]*Guilin University of Technology, Guangxi Key Laboratory of Spatial Information and Geomatics, China, 541000*
[3]*College of Mechanical and Control Engineering, Guilin University Of Technology, Guilin, Guangxi Zhuang Autonomous Region, China, 541000*
729020742@qq.com

Abstract

This paper puts forward text filtering system based on SVM algorithm regarding to the problem that bad and sensitive information of user tag that may exist in the collaborative geographic information plotting system. Then further improvement has been made towards the TF-IDF for feature extraction and thus achieved the function that automatically blocks the bad and sensitive information marked by users.

Keywords*: SVM; text filtering; feature extraction; TF-IDF*

1. Introduction

Along with the constant development and progress of ages, the internet has brought the whole world with unprecedented new opportunities. After the agricultural and industrial society, we are now marching towards the information society with all the efforts. However, the network also becomes the carrier of bad information, which seriously polluted the cyberspace. Among them, content such as pornography, reactionary, bad words are particularly serious with most of the bad information being in disguise in the following ways such as using special symbols as "☆" , "♯", "&" and "*"etc. or dividing Chinese character components into two parts or even replacing with spelling and so on. Therefore, how to effectively filter bad information has become a sort of question that deserves our attention.

Based on a subproject of "863 Program", this paper makes the filtration of towards the bad information marked by the user in collaborative plotting using SVM (Support Vector Machine) algorithm and realizes the masking function of bad and sensitive information.

2. Text Classification

2.1. Word Segmentation

Among the language components, vocabulary is the only one that can do independent activities and also has meaning. It is known to all that English words use spaces between the words while Chinese expressed with characters and thus with no distinction mark between vocabularies. Therefore, Chinese word segmentation processing becomes the key issue during the information filtering process.

It produces a lot of word after particles and number of dimensions is also high in the text which can even reached at thousands of dimensions or even hundreds of thousands of

* Corresponding Author

ISSN: 1738-9976 IJSIA

dimensions. If there are no dimension cuts, then dimension disaster would occur and thus stop words after Chinese word segmentation should be cut such as “的”, “了”, “我的” and “可以” *etc*. After traversing directories and files of the data set, text segmentation could be made and finally forms a dictionary model. This design makes the pretreatment after word segmentation process and automatically removes spaces between words and other various characters so as to get rid of the camouflage function of the bad and sensitive information, which can be shown in Figure 1.

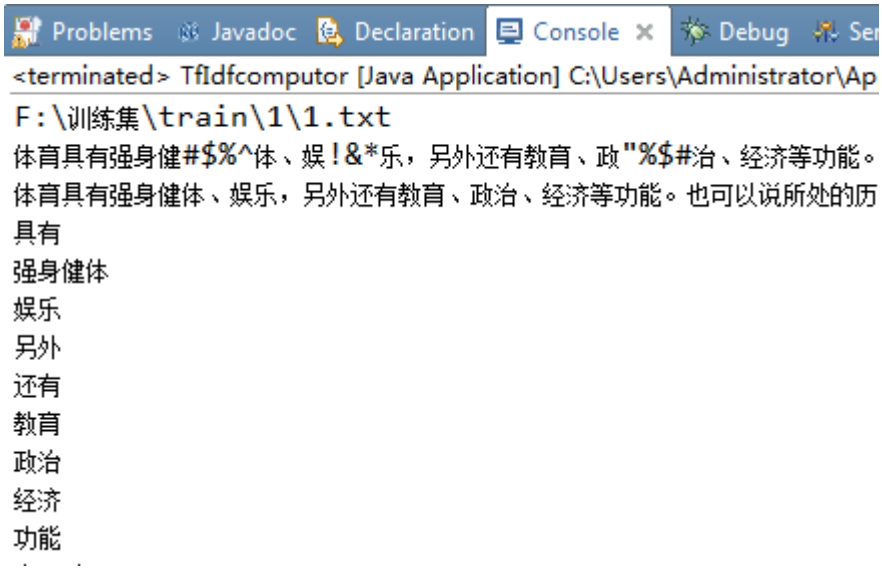

Figure 1. Removing Stop Words then Segmentating Text

2.2. χ^2 Statistics

χ^2 statistics mainly used for the feature extraction after word segmentation and thus to reduce the dimensions[1-3]. It can reflect the degree of relationship between the vocabulary entries and text category and if a term has a higher statistics towards a certain category, the relationship degree between the vocabulary entry and the category is higher and thus owns more category information. Therefore, this design uses CHI (CHi- square statistic) algorithm for feature selection after word segmentation. Its principle is as follows:

Suppose there are H texts, among them, h texts are about military which studies the correlation between “plane” and military. Assume that A is the number of the texts that both contains “plane” and belongs to “military” category and B is the number of the texts that contains “plane” but not belongs to “military” category while C refers to the number of the texts that not contains “plane” but belongs to “military” category and D refers to the number of the texts that neither contains “plane” nor belongs to “military” category, which is shown as Table 1.

Table 1. Chi Algorithm Feature Selection Principle

	Belongs to “military”	Not belongs to “military”	Total
Contains “plane”	A	B	A+B
Not contains “plane”	C	D	C+D
Total	A+C	B+D	H

Among them, A+B+C+D=H, A+C=h and B+D=H-h.

The probability that H texts contain "plane" is (A+B)/H, and the number of the texts that belongs to "military" is A+C which means E11 texts contains the word "plane". Suppose that theoretical value is E, actual value is xi and extent of the deviation is D_n.

$$E_{11} = \frac{A+B}{H}(A+C) \qquad E_{12} = \frac{A+B}{H}(B+D)$$

$$E_{21} = \frac{C+D}{H}(A+C) \qquad E_{22} = \frac{C+D}{H}(B+D)$$

$$D_n = \sum_{i=1}^{n} \frac{(x_i - E)^2}{E} \tag{1}$$

Then, $D_{11} = \frac{(A - E_{11})^2}{E_{11}}$ and value of D_{11}, D_{12}, D_{21}, D_{22} can be obtained through the same method.

$$\chi^2(Plane, Military) = D_{11} + D_{12} + D_{21} + D_{22} \tag{2}$$

Putting the value of D_{11}, D_{12}, D_{21}, D_{22} into the formula (2), the formula (3) can be obtained.

$$\chi^2(\text{Plane, Military}) = \frac{H(AD - BC)^2}{(A+C)(A+B)(B+D)(C+D)} \tag{3}$$

Therefore, the vocabulary entry and category can be expressed in the following formula:

$$\chi^2(t, C_i) = \frac{H(AD - BC)^2}{(A+C)(A+B)(B+D)(C+D)} \tag{4}$$

Due to the H, h and H-h have the same words towards the same category of text, we only have to care about the square root size order that a bunch of words towards one specific class instead of the specific value. And thus (4) can be simplified as (5):

$$\chi^2(t, C_i) = \frac{(AD - BC)^2}{(A+B)(B+D)} \tag{5}$$

Through calculating, every word in each category will obtain its CHI value and then rank all the words in each category based on the CHI and achieve the former n words. At last, extract first n words of each category and forms a new characteristic vectors and thus a new set of attributes would be obtained.

Though χ^2 statistics can make feature extraction, it has the following disadvantages: (1) Large amount of calculation towards the feature selection plus a low efficiency. (2) The dimension is still high after feature selection and unable to provide with attribute weight. Therefore, this paper would extract feature vector through another modified TF-IDF algorithm to determine the final attributes.

2.3. Modified TF-IDF Algorithm

Seeing entry ω as random variable, ω takes value from in all kinds of categories equals to the word frequency in each class. Variance $D(\omega)$ refers to ω's degree of dispersion in each category. The smaller $D(\omega)$ the is, the larger the degree of dispersion of ω in each category is and the less effect on classification would become. If ω presents an approximate uniform distribution of in each category, and then the $D(\omega)$ would be close

to zero. Due to the probability of ω's distribution in each category is difficult to calculate, this paper choose mean variation $\bar{d}$ to substitute $D(\omega)$.

Supposing that there are N categories in total, $TF_i(\omega)$ shows word frequency of entry ω in C_i category and $\overline{TF(\omega)}$ shows the average word frequency of entry ω in each category which is shown as formula (6).

$$\overline{TF(w)} = \frac{1}{N}\sum_{i=1}^{N} TF_i(w) \tag{6}$$

Suppose that the mean variation of ω in each category is $\bar{d}(w)$ and thus the mean square deviation is shown as formula (7).

$$\overline{d^2}(w) = \frac{1}{N}\sum_{i=1}^{N}\left[TF_i(w) \quad \overline{TF(w)}\right]^2 \tag{7}$$

Obviously, if ω presents a uniform distribution in each category with $\overline{d^2}(w) = 0$, it means ω has no effect on the classification. Assume there are M texts in the C_i category, $\overline{TF_i(w)}$ refers to the average word frequency of entry ω in C_i category with the calculation method as (8).

$$\overline{TF_i(w)} = \frac{1}{M}\sum_{j=1}^{M} TF_{ij}(w) \tag{8}$$

Using $\overline{d^2{}_1}(w)$ to indicate the mean square deviation of ω in Ci category, formula (9) can thus be achieved:

$$\overline{d_1{}^2}(w) = \frac{1}{M}\sum_{j=1}^{M}\left[TF_{ij}(w) \quad \overline{TF_i(w)}\right]^2 \tag{9}$$

Now transform the formula (9) into (10) and make it smaller than 1.

$$\overline{d_2{}^2}(w) = \frac{\frac{1}{M}\sum_{j=1}^{M}\left[TF_{ij}(w) \quad \overline{TF_i(w)}\right]^2}{\frac{1}{M}\sum_{J=1}^{M}(TF_{ij}(w))^2} \tag{10}$$

And thus after modification, TF-IDF can be expressed as:

$$Q(w) = TF(w) \times IDF(w) \times \overline{d^2(w)} \times \overline{d_2{}^2(w)} \tag{11}$$

Delete the smaller value of feature vector based on TF-IDF value, and then use the zoom tool built in the LIBSVM[5] to normalize all the TF-IDF value and then form the vector table after zoom to [-1, 1] which is shown as in Figure 2. Among them, the first line is vector label and this paper divided the text into two kinds with bad information text marked as -1 and others marked as 1. The other columns represent all feature vectors and rows are the number of text of training. Table 2 shows that in the SVM model, the accuracy rate of different feature selection methods in various dimensions.

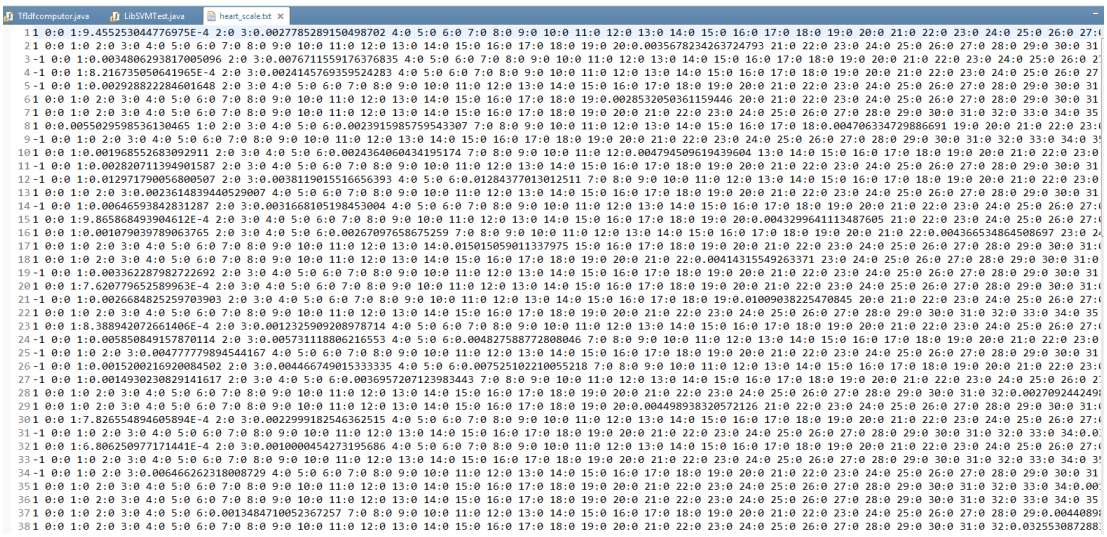

Figure 2. The Result of Normalizing all the TF-IDF Value

Table 2. Different Feature Selection Methods and SVM 's Accuracy Rate in Various Dimensions

Methods \ Dimensions	1000	3000	5000
Only CHI	52%	62%	60%
Only TF-IDF	57%	64%	61%
Combination of CHI and TF-IDF	75%	87%	86%

It can be seen from the Table 2 that the combination of CHI and TF-IDF in extracting characteristic values can effectively improve the accuracy of SVM classification; plus, it can be obtained that the increasing dimension may reduce classification accuracy and thus feature selection after the word segmentation is rather necessary.

3. Support Vector Machine Theory

Support Vector Machine theory is put forward in 1995 and shows many unique advantages [4] in solving the pattern recognition of small sample, nonlinear and high dimension ones. The text vector shows that the dimension is rather high if there is no dimension reduction process after word segmentation, and it is hard to handle this with other algorithm but SVM can do this. Towards the text classification, it would be seen as nonlinear as linear is one special case of nonlinear and then the paper would make simple introduction of nonlinear support vector machine (SVM).

3.1. Nonlinear SVM (Support Vector Machine)

Sensors are unable to solve the xor problem and thus one way is to use multilayer forward network and the other would be mapping the input vector to a high-dimensional feature vector space and then constructed the most optimized classification plane in this space, which can be called as SVM. When the low dimension inverts into a high dimensional feature space, it is hard to find the most optimized classification plane in the feature space; however, this can be artfully solved through introducing kernel function. RBF (Radial Basis Function) kernel function adopted in this design is shown as formula (12), and theory of support vector machines after introducing kernel function is shown as Figure 3.

$$K(x,y)=\exp\left[-\gamma\left\|x-y\right\|^2\right]^q \tag{12}$$

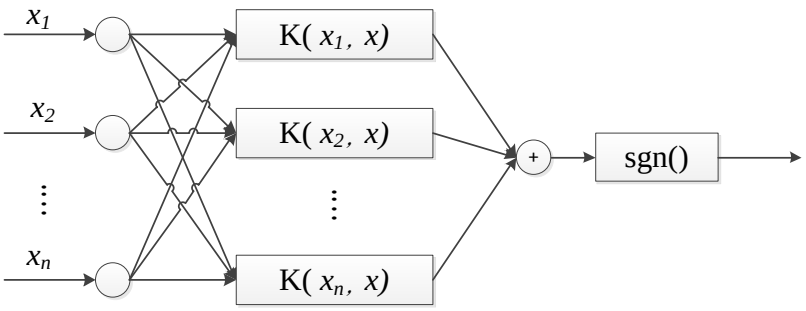

Figure 3. Kernel Function Support Vector Machine Theory

Then support vector machine can be described as:

(1) Suppose the known training set $Z=\{(x_1,y_1),(x_2,y_2),\cdots(x_n,y_n)\}\in(X,Y)^N$, $x_i\in X\subset R^n$, $y\in(-1,1)\quad i=1,\cdots,n$.

(2) Choose kernel function *K(x, y)* and the appropriate penalty parameters C >0, and then construct and solve the optimal problem:

$$\begin{cases}\min\limits_{\alpha} & \frac{1}{2}\sum\limits_{i=1}^{l}\sum\limits_{j=1}^{l}y_iy_j\alpha_i\alpha_jK(x_i,x_j)-\sum\limits_{j=1}^{l}\alpha_j \\ & s.t.\quad \sum\limits_{i=1}^{l}y_i\alpha_i=0 \\ & 0\le\alpha_i\le C\quad i=1,\cdots,l\end{cases} \tag{13}$$

Then get the optimal solution: $\alpha^*=(\alpha_1^*,\cdots,\alpha_n^*)^T$.

(3) Calculate $\omega^*=\sum\limits_{i=1}^{l}y_i\alpha_i^*x_i$ choose one positive component α of α^* which is smaller than C.

$$b^*=y_j-\sum_{i=1}^{l}y_i\alpha_i^*(x_i,x_j) \tag{14}$$

(4) Construct the hyperplane $(\omega^*\cdot x)+b^*=0$ and obtain the decision function:

$$f(x)=\mathrm{sgn}(\sum_{i=1}^{l}y_i\alpha_i^*K(x_i\cdot x)+b^*) \tag{15}$$

4. Experiment and Results

Choice of operation platform, development platform and database platform in this scheme are shown as Table 3.

Table 3. Application Platform and Software Environment

Name Requested	Detail Requirement
Server Operating Systems	Windows Server 2008 Version
Database runtime environment	MYSQL 5.1 Version
Software development environment	MyEclipse10.7 Version and SVN server
Application platform	JDK 1.7 Version
WEB Server Platform	Tomcat 7.0 Version
Server hardware requirements	Above Multi-Core CPU 2.0GHz, hard disk more than 1T, memory more than 2G
Client software environment	Support browser of high-speed mode
Client hardware environment	Above CPU 1.0GHz, hard disk more than 320GB, memory more than 1G

Based on the above theory and its principle, the flow chart is as shown in Figure 4. Among them, preprocessing includes word segmentation and remove of stop words etc. The SVM model program adopted in this experiment uses LIBSVM [5] toolkit package for reference which refers to the SVM software package explored by Professor Zhi-Ren Lin in Taiwan. It can solve some problems like classification, regression, distribution of track etc. and also provides many language interfaces as JAVA and MATLAB etc. and thus can be used in Windows or UNIX platform.

In this experiment, the training text both serves as the training text and also as a test text so as to make the precision of the model be the highest. The model precision is not too high after getting the SVM classifier model in the first place and with the initial accuracy of this design is Accuracy=69.230769%. Then cross validation should be adopted to choose the optimum parameter -c and -g in order to adjust the parameters after improving the model accuracy. The accuracy could be achieved 86.379125% after adjusting precision. After adjusting parameters, make the training towards the whole training set to obtain the support vector machine (SVM) model and then use it to make text prediction and measurement test towards the text under test. Some important parameters in this experiment are shown in Table 4.

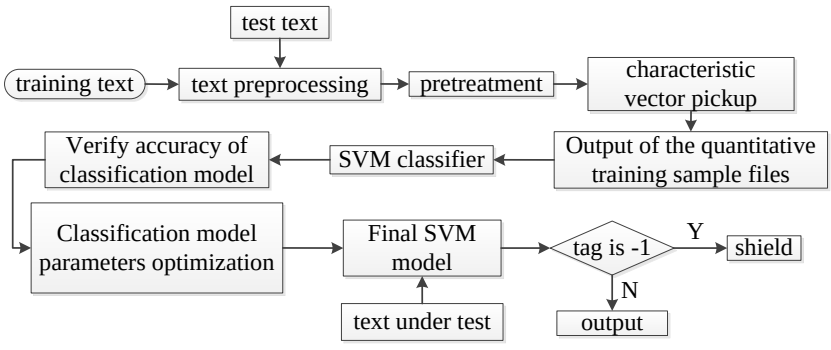

Figure 4. Flow Chart of SVM Bad Information Classification

Table 4. LIBSVM Parameter Setting

Parameter	Parameter specification	Parameter values	Parameter value meaning
-s	SVM type	0	C-SVC
-t	Kernel function type	2	RBF function
-c	Set up parameters of C-SVC, e –SVR and v-SVR	0.3536	—
-g	Gamma function setting in kernel function	0.7017	—

The training text would be divided into two categories which including "-1" and "1" after obtaining the final model and then filtrate the "-1" class text. When input the text under test, the model would make automatic classification of the articles so as to realize the function of bad text filtering which is shown as Figure 5.

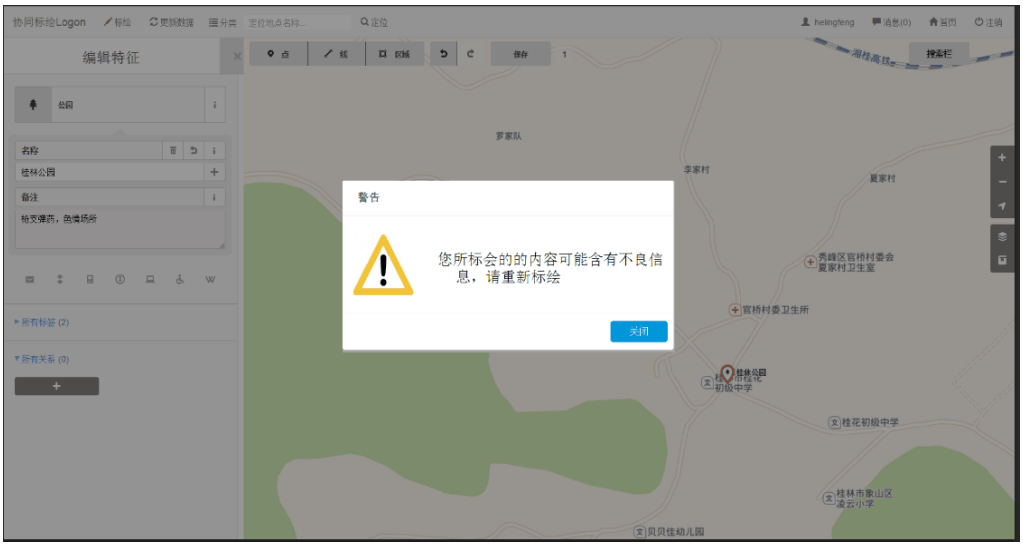

Figure 4. Filtering of Bad Information in the Comments

5. Conclusion

Through procedures of word segmentation, feature extraction and SVM model *etc.*, this design has successfully realized the automatically screening of the bad and sensitive information marked by users in the collaborative plotting system. Besides, the author uses two kinds of algorithm for feature selection in order to achieve the purpose of dimension reduction and improve the accuracy of SVM model. Plus, this design improves model accuracy through changing the choice of parameters and kernel function in SVM part. The final results show that the model accuracy is much higher and could effectively filter the bad and sensitive information.

Acknowledgements

This research work was supported by the 'Ba Gui Scholars' program of the provincial government of Guangxi, the National Natural Science Foundation of China (Grant No.61540054), National High Technology Research and Development Program 863 under Grant No. 2013AA12A402, Natural Science Foundation of Guangxi Provincial under Grant No. 2013GXNSFAA019349.

References

[1] H. Xuan-Jing, W. Li-De, S. Qi yang zhi and X. Guo-Wei, "Journal of Chinese Information Processing", vol. 6, no. 14, **(2000)**.
[2] L. Hui, S. Zhong-Zhi and X. Zhuo-Qun, "Journal of Chinese Information processing", vol. 2, no. 16, **(2002)**.
[3] M. Hui-Min, "Research and Implementation of the Automatic Chinese Text Categorization", North China Electric Power University, Beijing, **(2004)**.
[4] L. Xia and L. Wei, Computer Education, vol. 20, no. 2, **(2007)**.
[5] Chang C. C. and Lin C. J., LIBSVM: A library for support vector machines [EB/OL]. http://www.csie.ntu.edu.tw/~cjlin/libsvm, **(2006)**.
[6] M. Hammami and Y. Chahir, "Web Guard: Web based adult content detection and filtering system", Proceedings of the IEEE/WIC International Conference on Web Intelligence, Chicago, United states, **(2003)** September 213-218.
[7] O. Chapelle, V. Vapnik, O. Bousquet, Mukherjee and Sayan, "Machine Learning", vol. 46, no. 1-3, **(2002)**.
[8] C. Cortes and V. Vapnik, "Machine Learning", vol. 11, no. 20, **(1995)**.
[9] Amari S. and Wu S., "Neural Networks", vol. 12, no. 6, **(1999)**.
[10] Y. H .Li and A. K. Jain, The Computer journal, vol. 8, no. 41, **(1998)**.

Authors

Xiao-lan Xie, She got her PhD in Xidian University, Shan Xi, China. She is a Professor in School of information science and engineering, Guilin University of Technology. Areas of research include Cloud computing, Grid computing and Intelligent Decision System. She is a committee member and deputy secretary general of Cloud computing expert committee of China communication society. She is also a member of China computer society CCF and IEEE CS.

Zhen Long, He pursues his master degree in Guilin University of Technology. His research interests include embedded system, machine learning, data mining, *etc*.