A DBN-Based Classifying Approach to Discover the Internet Water Army

Weiqiang Sun, Weizhong Zhao, Wenjia Niu, Liang Chang

A DBN-based Classifying Approach to Discover the Internet Water Army

Weiqiang Sun[1,*], Weizhong Zhao[1], Wenjia Niu[2], Liang Chang[3]

[1]College of Information Engineering,
Xiangtan University,
Xiangtan City, Hunan, 411105, China,
{swq.sdh.llh,zhaoweizhong}@gmail.com
[2]Institute of Information Engineering,
Chinese Academy of Sciences,
Minzhuang Road 89, Beijing, 100093, China,
niuwenjia@iie.ac.cn
[3]Guangxi Key Laboratory of Trusted Software,
Guilin University of Electronic Technology,
Guilin, 541004, China

ABSTRACT. The Internet water army (IWA) usually refers to hidden paid posters and collusive spammers, which has already generated big threats for cyber security. Many researchers begin to study how to effectively identify the IWA. Currently, most efforts to distinguish non-IWA and IWA in data mining context focus on utilizing classification-based algorithms, including Bayesian Network, SVM, KNN and etc... However, Bayesian Network need strong conditional independence assumption, KNN has big computation costs, above approach may affect the effectiveness to some extent in real industrial applications. Hence, Neural Networks-like deep approach for IWA identification gradually becomes an emerging but possible direction and attempt. Unfortunately, there also exists one main problem, which is how to balance the deep learning and computation costs in hierarchical architecture. More specially, combine leaning-level heuristic training design and computing-level concurrent computation is a challenging issue. In this paper, we propose a collaborative hierarchical approach based on the deep belief network (DBN) for IWA identification. Firstly, a DBN-based collaborative model with hierarchical classifying mechanism is built. Then towards Hadoop platform, the Downpour Stochastic gradient descent (Downpour SGD) is exploited for DBN pre-training. Finally, the dynamical workflow will be designed for managing the whole learning-based classifying process. The experimental evaluation shows that the valid of our approach.

KEYWORDS: Internet Water Army, DBN, Downpour SGD, Collaboration, Hadoop

1 Introduction

With the rapid development of the Internet, cyber security issues become more and more prominent. Recently, the internet water army (IWA) draws more and more attentions. The IWA refers to post online comments or articles with particular content on social media for some hidden purpose in order to affect public opinion. They intentionally or unintentionally spread rumors and attack others, causing extreme emotions and national antagonism, which belong to an emerging threat for cyber security. Therefore, effective regulatory on IWA is desperately needed.

To defend the IWA, the basic challenge is to identify IWA firstly. Actually, IWA identification is essentially a classification, which should analyze the users' historical behaviors and compute the inner-class and inter-class differences between IWA and normal user. Most common classification algorithms have been exploited in IWA identification, such as Bayesian Network, SVM, KNN, and Neural Networks and so on. However, for real industrial application of big data context, these approaches have some disadvantages to some extent. For instance, the Bayesian network is based on probability and statistics, which predict samples classes using Bayesian principles. However, Bayesian principles need a strong conditional independence assumption which is always invalid for IWA in real scenario. SVM requires prior calculation of the samples' space vector and the weight of each dimension, while the weight setting always relies on experience and problem analysis and directly affects the accuracy of the result. KNN is a lazy learning method, which stores samples until they are needed to learning. It may lead to a large computational overhead if the sample set is complex.

With the development of deep learning, the work based on neural network algorithm draws special attention again. As a commonly used classification algorithm, it determines its model parameters by training and objectively reflects the impact of various factors on the final result. However the basic neural network model is so complicated. When the training date set scales up, the training process will take too much time to complete, and furthermore, improper initializations of weight easily result in the local minimum problem. These disadvantages can bring poor convergence, low accuracy, time-consuming problems when using neural network-type algorithm to detect IWA.

Hence, a collaborative hierarchical approach based on DBN will be proposed in this work. Our approach will build a parallel-improved DBN model and defines the mechanism of DBN collaboration between the various parts of the model. The model mainly consists of three modules: user data pre-processing module, DBN model training module and the collaboration module. Especially for the collaboration module, we define the feedback process of the convergence and accuracy of the DBN in the form of workflow. Beside, parallel computing model are used in user data pre-processing module and DBN model training module. We hope that this approach can not only improve the convergence and accuracy of IWA detection, but also solves the problem of costing too much time in the model training process with the mass of sample data.

The rest of the paper is organized as follows. Section 2 discusses related work, followed by the presentation of a DBN-based classifying approach and its details in Sec-

tion 3. Section 4 presents the experimental results that illustrate the benefits of the proposed scheme compared to other approaches. Finally, Section 5 provides the concluding remarks and future work.

2 Related work

The related work of IWA identification can be grouped into two aspects: behavior-based recognition and content-based recognition. In addition, spam detection work in Web 2.0 also has close and significant inspiration for IWA identification.

Behavior-based work can be further divided into two categories: direct calculation type [1-2] and training-leaning type [3]. In the direct calculation type, Chen Kai et al [1] aimed at the account list which has forwarded hot topic. They determine the initial IWA sample collection S and fans of S by artificial judgments and choose part of the fans of S to join in S through a sample filter in next epochs. This method uses the social relations of the IWA to identify more IWA. However, it has some limitations that it ignores the few relations between the IWA. Zhang Guoqing et al [2] extend the feature vector to 16 dimensions and identify IWA by comparing the pre-set threshold to the result of the calculation of the weighted dimensions. However, in this method, the weight of dimensions is set relatively subjective. In the leaning-training type, Han Zhongming et al [3] transform the behavior and features of the micro-blog users to characteristic vector, constructing a probabilistic graph of features and behavior: $P(D|w)P(w)=(\Pi P(z^{(i)},y^{(i)}|x_1^{(i)},\ldots x_n^{(i)},w))P(w)$, where D is the number of samples, $x_n^{(i)}$ is the feature characteristic of dimension n of the i th user, $yn^{(i)}$ is the behavioural characteristics of dimension n of the i the user, w is the weight of each feature weight, z is the probability for user is IWA. Finally, use the Maximum Likelihood Estimation to get parameter w and so on in probabilistic graph, which can calculate the probability of that a given user is IWA. The above methods are largely dependent on the selected characteristics which will result in selection too artificial and subjective.

Content-based work mainly focus on posting content itself, through distinguishing IWA through statistics of normal and abnormal posting depending on language characteristic. There are three mainly ideas in this approach: the first one4 is to use emotional tendency for a single posting. In this idea, they consider IWA has a strong emotional tendency to beautify or demonize something; the second [4-7] is based on the tendency to deceive of a single posting. In this idea, posting of IWA is deceptive opinion spam with some statistical law. The work [8] is based on similarity of multiple posts. In this idea, they consider IWA tend not to write multiple posts in order to pursue post speed. Instead, IWA prefers to make slight changes to the same post and forward largely. However, these methods based on the content analysis have a common defect that a simple analysis of contents would not be conclusive evidence for IWA identification. In addition, in the evolution of fighting with IWA, count of the machine post decreased and manual post increased, which makes identification based on content analysis extremely difficult.

Spam is the term widely used to describe the phenomenon that spam messages spread everywhere. Spam, broadly defined, refers the behavior of sending information

actively with no specific target through the electronic information system, involving micro-blog and forums. Non-target greedy spread for behavior and spam for content, there are similarities between Spam and IWA. Anti-Spam research has a history of ten years, mainly including text-based, behavior-based and image-based feature extraction. There has been a lot of typical work: a) text-based mainly uses bag-of-words (BoW) [9], space binary polynomial hashing (SBPH) [10], orthogonal space bigrams (OSB) [11], and biological immune system (BIS) [12]; b) behavior-based spam detection is to filter spam by extracting the behavioral characteristics of the email. These commonly used methods are based on system log and message header information [13], attachment [14], and the network [15]; c) image-based feature extraction mainly focuses on extracting the key characteristics of the picture. There are a lot of the typical approaches to extract e-mail feature based on machine learning in intelligent spam detection, involving Bayesian [16,17], K-nearest neighbor[18,19], Boosting Trees [20], SVM [21,22], Rcchio [23] and Artificial Neural Networks [24], and so on.

It is worth noting that the Neural Network has been a very special research in social media Spam, due to the surprising improved recognition effect. However, it is difficult to be widely used because of the non-ignorable locally optimal, slow convergence [25] and other problem. In the works of J.Clark et al [24], the email classification based on the artificial neural network received 99.13% recognition accuracy on Ling-Spam corporace10 dataset. Guangchen Ruan et al [26] use BP Neural Network to identify Spam. First, they extract feature from email using KLGA algorithm and reduce the dimension of the vector space model of message. Then they classify the message by three layers BP Neural network. This approach has a 97%-99% recognition rate on PU and Ling-Spam corpora10 dataset. But its main shortcomings include the need for more time to compute and the great impact on network convergence the initial settings have.

With the breakthrough that Hinton et al did on Neural Network in 2006, deep learning technology, especially DBN [27], gained success in image recognition, voice recognition, and natural language content comprehension. So this has inspired us to take advantage of the new generation of neural networks to extract behavior and feature, exchanging time for the calculation of neurons for accuracy and adaptive performance in IWA recognition.

3 Overview of the classifying approach

The classifying approach is designed to consist of three modules: user data pre-processing module, DBN model training modules and collaboration module. The user data pre-processing module convert the original user descriptive information to vector, and divide the user data set into two parts: DBN model training data and test data; DBN model training module trains DBN model with training data set which is get in user data pre-processing module, includes two processes: model pre-training process and model fine tuning process; Collaboration module defines the feedback process of the convergence and accuracy of the first two modules in the form of workflow. Schematic overview of the approach is shown in Figure 1.

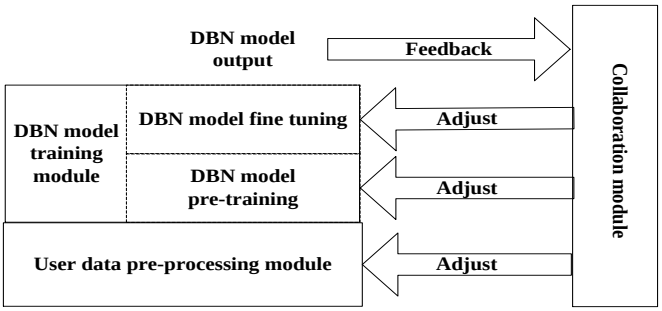

Fig. 1. Schematic Approach Overview

3.1 Data pre-processing

User information needs to be expressed in the form of a mathematical transformation before the classification. Objectively, the user relevant information contains user name, user registration time, previous login time, login IP, browsing history records, post history records, replies history records, friends' records, followers' records, followings' records. We select the representative information as the reference to classify users and accordingly propose a multi-attribute description user information framework.

Using the multi-attribute descriptive user information framework, user descriptive information can be transformed into a vector which is mathematical representations of the user information. In addition, in order to facilitate setting initial weights in DBN model training, value of each dimension in user information vector need to be in [-1,1]. We present a normalization to deal with each dimension of the vector. That is to say we extract the value range of each dimension of the vector, find and normalized the dimensions which value range is beyond [-1, 1].

As shown in Figure.2, the vector generation process and normalization process can use parallel computing model. In user descriptive vector generation phase, all user information is randomly divided into m group to parallel process. Each group will be responsible for transforming user descriptive information to descriptive vector and each vector will be assigned an ID number then. Finally, we obtain a collection which contains pairs of user ID and descriptive vector. In the user descriptive vector normalization process, we first use MapReduce parallel frame to find the dimensions which value range is beyond [-1, 1] of the vector as well as the biggest absolute values of these dimensions. Then we use these values to normalize these dimensions. Normalization process can also divide user descriptive vector into m group to normalize in parallel.

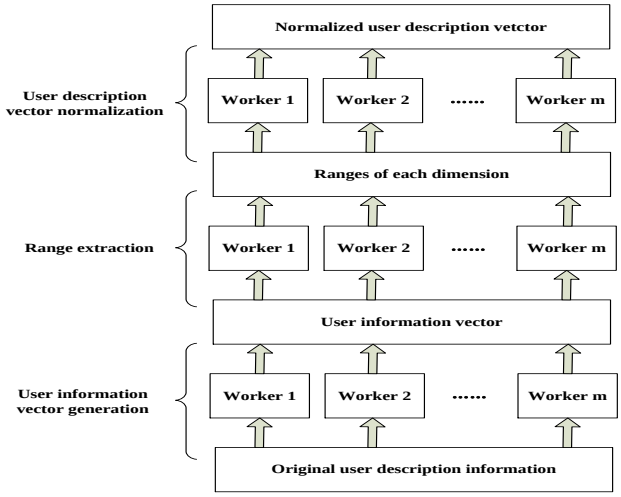

Fig. 2. Figure.2 Parallel Pre-processing Diagram

Through the above process, the normalized descriptive user vector collection can be obtained. A part of the collection data is classified, that is, this part of users has been marked as IWA or normal. This type of date is called "classified data sets". The "classified data sets" is divided into two parts for the subsequent DBN model training. One part called "training data set" is for the training while the other part called "test data set" is for accuracy tests for the DBN model obtained after training. As the DBN model needs enough samples to learn so that hiding law in these samples can be simulate, the "training data set" generally is given more samples. However, too many samples will increase the amount of computation and bring an over calculate disadvantage.

3.2 DBN Training

DBN (Deep Belief Network) model is a kind of deep neural network model and is a probability generation model that consists of multilayer random variables. Basic DBN model consists of two layers of RBM (Restricted Boltzmann Machines) and a layer of BP neural network (Back Propagation Neural Network). The training process of the DBN model is divided into two phases: pre-training phase and fine-tuning phase.

Pre-training phase use layer by layer unsupervised greedy learning methods to train the two RBM layers in the model: First, use the input data and the first layer if hidden layers as a RBM to train to get the parameters, then fix the parameter of the RBM and take the first hidden layer as the visible layer and the second as the hidden layer to train the new RBM. Finally, we obtain the parameters of the second RBM and determine the parameters of the two RBM at the moment and complete the pre-training process of the DBN model. In this process, the training process of each RBM is independent, which greatly simplifies the process of training the model.

After the pre-training, the entire network is equivalent to BP neural network. This BP neural network contains two layers of hidden nodes, network parameters between the input layer and the first layer of hidden nodes, as well as the parameters between the two layers of hidden node, has completed initialization. You only need to randomly initialize the network parameters of the second hidden layer nodes and output nodes, you can take error back propagation trainings according to the normal BP neural network training methods until the model reaches convergence or termination conditions, this process is called fine-tuning.

In the DBN model pre-training phase, we use layer by layer greedy unsupervised learning methods to train two RBM. Compared to the traditional multi-feedback training model, this approach simplifies the training process model, accelerating the training speed of the model to a certain extent. However, In the face of massive training data set, single RBM layer training still takes a long time. So, parallel processing is applied to accelerate the single RBM layer training and speed up the pre-training of DBN model, shortening the DBN model pre-training stage time.

For parallel processing algorithms used in RBM, we examined the following three kinds of schemes: MapReduce model, the traditional SGD mode and Downpour SGD model. MapReduce model is good at data parallel processing in the usual sense, but not suitable for depth iterative calculation in deep network training; Traditional SGD (Stochastic gradient descent) model is the most commonly used optimized method for training the Deep Neural Networks. However, this model is essentially serial, which means data movement between machines would be very time-consuming; Downpour SGD is a parallel optimization to the traditional model and is an asynchronous stochastic gradient descent variant which use single DistBelief model and has many distributed copies. It is a good choice for large-scale data parallel computing. Accordingly, we choose DownPour SGD for parallel processing to train RBM training process.

As shown in Figure 4, parallel RBM training based on Downpour SGD has a basic idea: the training data is divided into several subsets and distributed on multiple Worker servers; each Worker server runs a copy of RBM model and just simply does communication with parameter server. The parameter server stores the current state of model parameters. Parameters of the model can be updated by updating the parameters storing in the parameter server. At training phase, each Worker server obtains the parameters of current state of the model from the parameter server and performs min-batch according to these parameters, calculate the gradient and push the results back to parameter server. In a simple implementation of Downpour SGD, you can set the Work server to obtain the updated parameters from parameter server after every "n-fetch" times of mini-batch and push the new calculated gradient back to parameter server to update.

The gradient update process is performed asynchronously in DownPour SGD, in this way, even a Worker server is down, it will not affect the work of other Worker servers. Although asynchronous update process will lead to the parameters of each Worker server slight difference, the algorithm has a good stability in current implementation.

After the training of the two RBM, the pre-training process of the DBN model is complete. The model is equivalent to a four-layer BP neural network now. The parameters between the three bottom layers have been initialized. Initialize the parameters between the top two layers and train the BP neural network with the training data set, which is fine-tuning process of the DBN model.

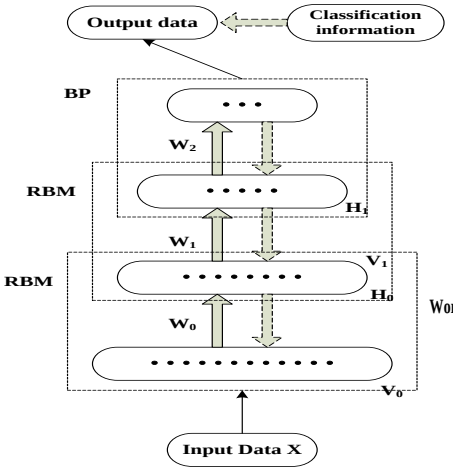

Fig. 3. Basic DBN Model

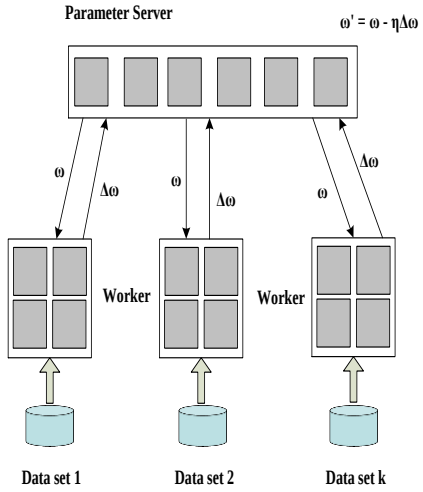

Fig. 4. Downpour SGD

3.3 Workflow-based Collaborative Learning

In the model training process, different settings of parameters in each module may affect the output of subsequent module so that it will affect the accuracy of the final DBN model. For example, if the proportion of the training data set is too low, it will prevent the IWA feature extraction, resulting in a low accuracy; if the maximum number of epochs is too low, it will make the network immature. In the collaborative module, we define the feedback process of the convergence and accuracy of the first two modules in the form of workflow. According to the convergence and accuracy of DBN model, we determine whether to make a reverse adjustment to the parameters of the user data pre-processing module and DBN training module or not, finally, improving the performance of the DBN model.

Workflow is a class of business processes that can be completely or partially performed automatically. Documents, information, and tasks can be passed and executed between different actors based on a series of procedural rules. The workflow we defined is shown as Figure 5.

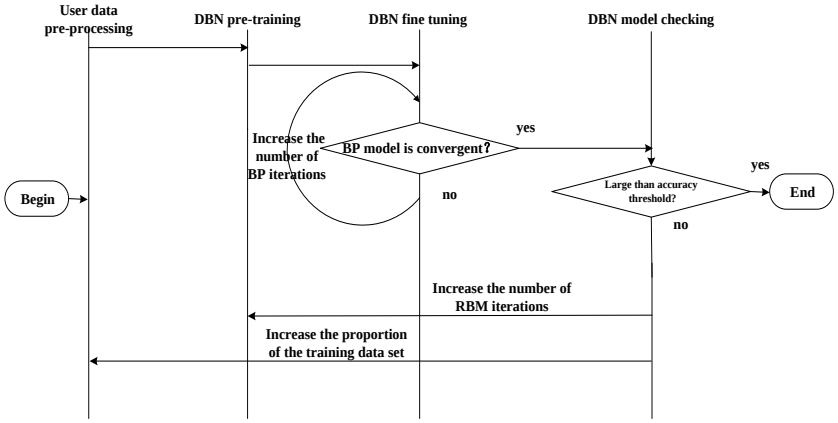

Fig. 5. Collaborative Learning Workflow

In this workflow, the data pre-processing module uses the original user data to generate a set of normalized user descriptive vector. After the completion of the DBN training process and the network weights have been initialized, the flow enters DBN fine tuning stage. If the BP network is not convergent in fine tuning stage, increase the number of epochs of BP until the convergence is achieved, then you get the trained DBN model. After checking DBN model with the test data set, and if the accuracy of DBN model does not achieve the desired threshold, increase the proportion of the training data set in classified data set in the user data pre-processing stage and increase the number of epochs of RBM in DBN pre-training stage. Retrain DBN model until the DBN model has a desired accuracy.

4 Experiment

4.1 Setting Up

We get 3019 user information from Sina WeiBo by using web crawlers. The information contains 26 behaviour related dimensions which including duration of registration, total number of micro-blog, the number of forwards, the number of collections, member level, the number of Tab, location information, the number of self-describing words, total number of links contained in all blogs, the number of followers, the number of followings, the ratio of mutual followers, rate of interactions, the time of most blogs, and so on. Then we try to mark these users manually which includes 588 IWA and 2431 normal users in three months, this is very challenging. All DBNs, which containing 4 hidden layers each with 2048 units, were trained using SGD with a mini-batch size of 128 training cases in pre-training. For simplicity, we use a default learning rate of 0.005 in Gaussian-binary RBMs while using a learning rate of 0.08 in binary-binary RBMs. For fine-tuning, we used SGD with the same mini-batch size in pre-training. The learning rate started at 0.1. If the accuracy fells at

the end of each epoch, the learning rate is halved. This process continues until the leaning rate fell below 0.001.

To evaluate the performance, we designed two experiments. We hope to check: 1) whether our improved DBN has acceptable training cost;2)whether our improved DBN has an averagely better identification accuracy than normal DBN.

4.2 Training Cost

We used 60% of the 3019 user information as the training data while the rest 40% as test data both in our improved DBN and normal DBN. In this experiment, we ran 100 epochs for Gaussian-binary RBMs and 50 epochs for binary-binary RBMs in pre-training. We determined that our improved DBN should take an accuracy of 85% to 90% as a threshold in DBN model checking part of the workflow. Figure 6 shows the time cost of DBN models in pre-training and fine-tuning when our improved DBN complete training and achieve the accuracy goal. We can observe that the time under the accuracy of 90% is just double of the time under 85%. It is acceptable to get more five percent accuracy to archive an accuracy of 90%.

4.3 Identification Accuracy

The second experiment was designed to show identification Accuracy with different number of training data and test data. The proportion of the training data in this experiment increased from 10% to 90% and the rest were used as test data. The figure 7 shows that our improved DBN always has an averagely better accuracy than normal DBN, especially when the proportion of training data is small than 40%. We can see that the accuracy will begin falling when the proportion is larger than 75%. When the proportion is 60%, both our improved DBN and normal DBN nearly have the same accuracy.

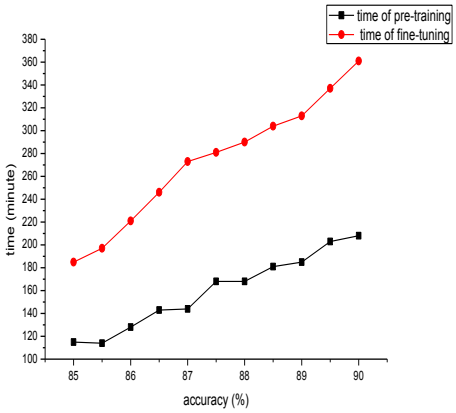

Fig. 6. Training Cost

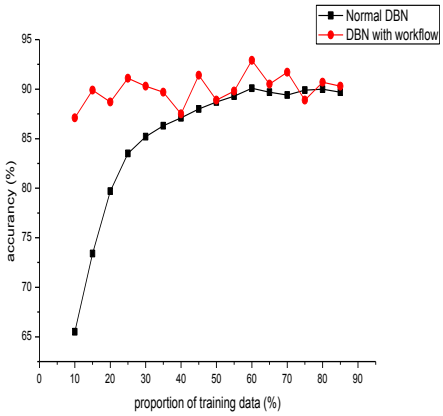

Fig. 7. Identification Accuracy

5 Conclusions

This paper proposes a collaborative hierarchical approach based on the DBN for IWA identification. We found that parallel computing can accelerate the user data pre-process and the training process when the simple set is complex and large. We believe that this research still runs its beginning stage, and in the future， we will put more study and discussion on how to control the balance point between computing cost and identification accuracy.

6 Acknowledgements

This research is supported by the National Natural Science Foundation of China (No. 61105052, 61103158, 61202398, 61272295), the Strategic Priority Research Program of the Chinese Academy of Sciences Grant (XDA06030200), Guangxi Key Laboratory of Trusted Software.

7 References

1. Kai Chen, Qu Zhou, Yi Zhou, Chengfeng Lin. Method for capturing water armies on microblog platforms, UNIV SHANGHAI JIAOTONG, CN103095499A (2013), Chinese Patent.
2. Guoqing Zhang, Jiangong Bian, Chuan Fu, Yanjun Li. Microblog ghostwriter identifying method and device, INST COMPUTING TECH CN ACADEMY, CN103198161A (2013), Chinese Patent.
3. Zhongming Han, Yueliang Wan, Fengmin Xu. Microblog water army identifying method based on probabilistic graphical model, UNIV BEIJING TECH & BUSINESS, CN103077240A (2013), Chinese Patent.
4. Wei Zhang, Zhonghua Zheng, WEI GAO, Zhihu Shuai, Yinxing Zhou. Detection and determination method of network navy, ANHUI BORYOU INFORMATION TECHNOLOGY CO LTD, CN102629904A (2012), Chinese Patent.
5. Xu Q, Zhao H. Using Deep Linguistic Features for Finding Deceptive Opinion Spam[C]// International Conference on Computational Linguistics, 2012: 1341-1350.
6. Lau R Y K, Liao S Y, Kwok R C W, et al. Text mining and probabilistic language modeling for online review spam detecting[J]. ACM Transactions on Management Information Systems, 2011, 2(4): 1-30.
7. Harris C. Detecting deceptive opinion spam using human computation[C]//Workshops at AAAI on Artificial Intelligence, 2012.
8. Bhattarai A, Rus V, Dasgupta D. Characterizing comment spam in the blogosphere through content analysis[C]//Computational Intelligence in Cyber Security, 2009. CICS'09. IEEE, Symposium on. IEEE, 2009: 37-44.
9. Gansterer W, Ilger M, Lechner P, et al. Anti-spam methods-state of the art[J]. Institute of Distributed and Multimedia Systems, University of Vienna, 2005.
10. Guzella T S, Caminhas W M. A review of machine learning approaches to spam filtering[J]. Expert Systems with Applications, 2009, 36(7): 10206-10222.

11. Siefkes C, Assis F, Chhabra S, et al. Combining winnow and orthogonal sparse bigrams for incremental spam filtering[M]//Knowledge Discovery in Databases: PKDD 2004. Springer Berlin Heidelberg, 2004: 410-421.
12. Ruan G, Tan Y. A three-layer back-propagation neural network for spam detection using artificial immune concentration[J]. Soft Computing, 2010, 14(2): 139-150.
13. Yeh C Y, Wu C H, Doong S H. Effective spam classification based on meta-heuristics[C]//Systems, Man and Cybernetics, 2005 IEEE International Conference on. IEEE, 2005: 3872-3877.
14. Hershkop S. Behavior-based email analysis with application to spam detection[D]. Columbia University, 2006.
15. Ramachandran A, Feamster N. Understanding the network-level behavior of spammers[C]//ACM SIGCOMM Computer Communication Review. ACM, 2006: 291-302.
16. Ming L, Yunchun L, Wei L. Spam filtering by stages[C]//Convergence Information Technology, 2007. International Conference on. IEEE, 2007: 2209-2213.
17. Meizhen Wang, Zhitang Li, Hantao Wu. An improved Bayes algorithm for filtering spam email[J]. Journal of Huazhong University of Science and Technology(Nature Science Editi on), 2009 (8): 27-30.
18. Xing Li, Ying Tian, HaiXin Duan. Implementation and evaluation of Chinese spam filtering system [J]. JOURNAL OF DALIAN UNIVERSITY OF TECHNOLOGY, 2008 (z1): 189-195.
19. Sakkis G, Androutsopoulos I, Paliouras G, et al. A memory-based approach to anti-spam filtering for mailing lists[J]. Information Retrieval, 2003, 6(1): 49-73.
20. Schapire R E, Singer Y. BoosTexter. A boosting-based system for text categorization[J]. Machine learning, 2000, 39(2-3): 135-168.
21. Drueker H, Donghui W W, Vapnik V N. Support vector machines for spam categorization[J]. Neural Networks, IEEE Transactions, 1999, 10(5):1048-1054.
22. TANG Zhao-hui, FU Jian-ming, DU Nan-shan. Design and Analysis of Spam-Filtering System Based on Words Segmentation [J]. Journal of Wuhan University(Natural Science Edition), 2005, 51(S2)191-194.
23. Schapire R E, Singer Y, Singhal A. Boosting and Rocchio applied to text filtering[C]//Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1998: 215-223.
24. Clark J, Koprinska I, Poon J. A Neural Network Based Approach to Automated E-Mail Classification[C]//Web Intelligence. 2003: 702-705.
25. Xu Z B, Zhang R, Jing W F. When does online BP training converge?[J]. Neural Networks, IEEE Transactions, 2009, 20(10): 1529-1539.
26. Guangchen R, Ying T. A three-layer back-propagation neural network for spam detection using artificial immune concentration[J]. Soft Computing, 2010, 14(2):139-150
27. HINTON G, OSINDERO S, THE A. A fast learning algorithm for deep belief nets[J]. Neural Computation, 2006, 18(7): 1527-1554.