International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 04 | Apr 2020 www.irjet.net p-ISSN: 2395-0072

Twitter Spam Detection using Cobweb

Vidhi Tiwari[1], Akarsh Srivastava[2], Mrs. P. Akilandeswari[3]

[1]SRMIST, Department of Computer Science and Engineering, Kattankulathur Campus
[2]SRMIST, Department of Computer Science and Engineering, Kattankulathur Campus
[3]Assistant Professor, Department of Computer Science and Engineering, Kattankulathur Campus, SRM Institute of Science and Technology (formerly known as SRM University).

Abstract - *With the advent of online social networks, or OSN, there has been a significant rise of users daily, with twitter being one of the largest social media networking sites in the world, for the world to have ever used. The network has been an ever-growing web of users who may or may not have malicious intent towards society. With the increase in the number of people using the OSN, there are more chances of exploitation. One of the major sources of exploitation is spam on the twitter feed. It is known by survey, that 1 in every 21 tweets is spam. There are multiple ways a tweet could be used in harmful ways as well as multiple ways a tweet can be made into a spam and hence, it is necessary to differentiate a tweet that is normal or spam. This project aims to classify spam using a data stream clustering technique called cobweb and a result enhancer namely gradient booster. Cobweb is an enhanced clustering algorithm that classifies data using trees. We aim to combine this to gradient boosting, which is also a tree-based classifier to create a more enhanced spam detection system than there already are.*

Key Words*: *Twitter; spam; cobweb; gradient boosting; URL; hashtag; phishing; mention; logarithmic function

1. INTRODUCTION

Ever since the dawn of the time of online social networks, OSNs (for example Twitter, Facebook, Instagram), there have been new and upcoming users everyday. This growing network provides opportunities to an array of people using the network since multiple accounts from a user are possible. While many users have been using this commendable technology for greater use such as reaching out to help to employment opportunities, organizing fundraisers or reaching out to help causes, it is often used for malicious motives which, unfortunately has a lot of channels. Twitter has been one of the most leading social media networks since OSNs with a growth rate of the number of users being exponential. A survey conducted in 2018 showed that there are a total of 231 million users on Twitter. Out of these about 200 million log onto their accounts everyday. This gives a lot of ammunition to spammers. A statistical study showed that 1in every 21 tweets is spam. With a number this high, we could be duped potentially every time we are logged in. Spammers use multiple techniques to create spam tweets. The most popular method is the shortened URLs. Since the URLs' destination isn't known, spammers use it to deceive users. Clicking on the link would lead you to get malware. Another method used is the use of hashtags. Any word with the symbol of '' in front of it can be used as a traffic generator. Tapping on it could lead to potential spam. 'Mentions' work the same way as hashtags. They both generate traffic on and off the platform.

2. CONCEPTS IMPLEMENTED

2.1 Cobweb Clustering

Cobweb is a data stream clustering algorithm that was developed by Douglas H.Fisher. It is a system that works on the idea of conceptual clustering of the input data. The algorithm observes the data and creates a classification tree. Each node symbolizes a class and all the daughter nodes are expressed in the form of a probabilistic concept (PC) which is a probable summarization of all attributes of the nodes which is used as a label for the parent node (class). There are a few operations that cobweb executes to create the classification trees:

1) Merging of The Nodes: Merging the nodes requires replacement by a singular node whose children are the additive of the children of the primary node set. The child node should summarize the attribute-value distributions of all objects classified under them in the tree.

2) Splitting a given node: The node in reference is divided into two and replaced with its children nodes.

3) Inserting a new node: A node is created which represents the object that will be inserted into the tree.

4) Passing an object down the hierarchy of the tree: This step calls the Cobweb algorithm on the object and the subtree rooted in the node.

The basic algorithm for cobweb works by taking a node as an input for a record. If the base has no children, the node at hand is set as the child node and the category of the child's features is added to the tree. However, if the case has children, the Category Utility is calculated for the given node and the best CU, that is, the smallest value leading class inherits the given node as a child. This process is repeated for all entries of the dataset. The main reason why a classification tree is used is to predict any missing attribute or recognize a new class of a new object and add it to the tree.

2.2 Gradient Boosting

Gradient boosting is one of the machine learning techniques for any set of classification and regression problems. It gives, as result, a prediction model in the form of an assemblage of weaker prediction models, which is a method that uses multiple weak algorithms to train into a better working model, typically decision trees. It works on building the model in a stage-like fashion much like most of the boosting algorithms do, and it creates a more accurate system by allowing the optimization of an arbitrary differentiable LS to give a generalized result. In many supervised algorithms used as learning algorithms, the programmer has an output variable p and a vector of input variables namely q described using a joint probability distribution P(p,q). Using a training set (p,q). . . (pn,qn) of already known values of p and corresponding values of q, the goal is to find an approximate value to a function that gives a minimized value of the expected value of some specified LF. The gradient boosting algorithm assumes a real-valued q and looks for an approximation for the value of p, from the given set of values, in the form of a weighted sum of functions from some class, called base learners, also known as weak learners.

3. STATE OF ART

Spam detection has had tremendous breakthroughs since the onset of everyday use of social media platforms, especially on twitter.

In 2019, a novel stream clustering technique was designed which used incremental naive bayes classifier that was trained to work efficiently on micro-clusters. The INBs capture the boundary and mean values of the microclusters, whereas the Euclidean distance just utilizes the mean value of the clusters. This mostly gives skewed results for asymmetric big micro-clusters. In this paper, DenStream was promoted to, by the proposed framework, to be called INB-DenStream. To show the efficacy of INB-DenStream, common and known methods much like DenStream, CluStream and StreamKM++ were applied to Twitter datasets and their performances were noted for values such as purity, overall precision, overall recall, F1 measure, parametric sensitivity, and computational complexity. The results of the comparison showed the superiority of their method to the rivals in almost all the datasets.

In 2017, Mahdi Washha,, Aziz Qaroush, Manel Mezghani and Florence Sedes formulated the Hidden Markov Model (HMM) to be a time dependent model for real-time filtering of spam tweets solely based on the topic of the tweets. This method is only functional on the easily available and openly accessible meta-data in the tweet object to detect spam tweets exiting from a stream of tweets related to a topic (e.g., #Trump), while considering the state of previously handled tweets associated with the same topic. Compared to the robust and well known classical time-independent classification method, the Random Forest algorithm, the experimental evaluation shows that the efficiency increases on the increase the quality of the topics in terms of overall precision, recalls, and F measure performance metrics.

In 2018, Rutuja Katpatal, Aparna Junnarkar studied the various classifiers and performed and presented a comparative study of them all for research purposes.

Anti Trust Rank algorithm is a known link-based spam detection algorithm which works on the principle that spam pages are more likely to be referenced by other spam pages. It may function on any possible platform. Since a real world connected web graph involves multiple billions of nodes, it is crucial to develop work efficient spam detection algorithms for making the process easier. Thus, in 2018, Yeon Seong Jeong, Joyce Jiyoung Whang, Inderjit S. Dhillon, Seung Goo Kang and Jungmin Lee developed the asynchronous Anti Trust Rank algorithm which allows us to reduce the number of arithmetic operations required to detect spam as compared to the traditional synchronous Anti-TrustRank algorithm by a large margin, without lowering the performance in detecting web spam.

Spam detecting systems mainly build their classification architecture which includes the binary classification of their data and then it can be solved by a chosen machine learning algorithm that suits best to give the best possible results. The ML algorithms, for example, Naïve Bayes classifier or support vector machine classifier then report the behavior of these designed models. Hence, in 2016, Miss Shukla, Twinkle Kailas and Prof.D.B. Kshirsagar showed the effect of the data related factors, namely the spam to non-spam ratio, the training data size, and data sampling, to the detection performance of the system. The feature of the shown system was time varying spam tweet detection. The System showed that spam detection was a huge challenge and it bridges the space between the performance evaluation. It mainly focused on the data

that was used for training and testing the system, the features and the model to identify the genuine, legitimate user and report the spam user by providing the answer in a binary value, that is, 1 or 0.

4. INFERENCE FROM SURVEY

Inference from the survey was done for over multiple journal papers as well as conference papers for a clear understanding of the subject. The basic idea of spam on twitter was gathered by papers written by Aparna Junnarkar, Rutuja Katpatal, Aparna Junnarkar [3]; H. Tsukayama [11]; D. Song, K. Thomas,V. Paxson and C. Grier [13]; G. Stringhini,G. Vigna and C. Kruegel [14]; Chao Chen [23]. The tested methods were understood and the results were analyzed. This gave an average measure of values as results that we use to compare our results with. This provides a standard to our work, given this is a system that is being built to provide an efficient alternate method to detect spam on twitter.

5. DATA SET ANALYSIS AND STUDY

With the use of machine learning algorithms: cobweb and gradient boosting, there is a need to train the algorithm before we test the system for detecting spam. Hence, there are two sets of data sets that are used.

5.1 Training Data set

This system uses a training data set which is used to help increase the accuracy of the algorithm used. The accuracy of the performance of the algorithm depends solely on the amount of training data used and the complexity of the data set, that is, a good mix of data in the data set. Programmers divide the data set that is acquired into two uneven parts. The greater, three-fourth is used for training while the smaller division is used for the testing. That is done when not a lot of data can be gathered. This project allowed us to gather enough tweets to have multiple data sets.

Table -1: Training data set attributes

	Attribute	Data Type
1	Tweet ID	Integer
2	Tweet	Text
3	Followers	Integer
4	Following	Integer
5	Is_retweet	Binary
6	Location	Text
7	Spam_or_notspam	Text (Spam/Not Spam)

5.2 Testing Data Set

The testing data set is similar to the training data set but smaller in data entries as it is only used for testing the accuracy of the system. In this case, it is missing an attribute for spam or not spam as that is the test, but at times the testing and training data set is the same.

Table -2: Testing data set attributes

	Attribute	Data Type
1	Tweet ID	Integer
2	Tweet	Text
3	Followers	Integer

4	Following	Integer
5	Is_retweet	Binary
6	Location	Text

6. Methodology Used

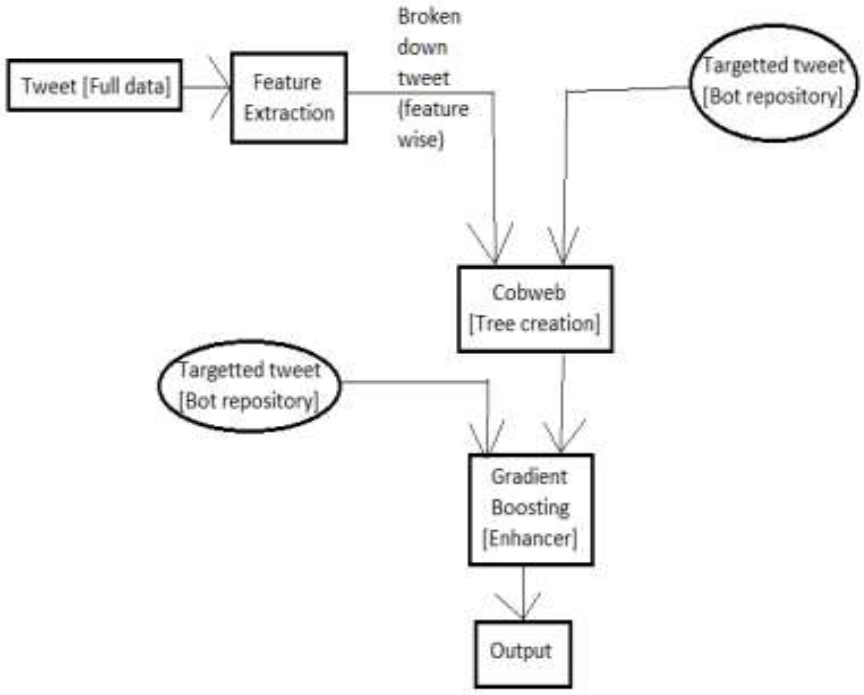
With this project, we propose to attempt to improve the method with which spam on the social media platform, Twitter, is detected.

We aim to implement cobweb, which is an enhanced conceptual clustering algorithm, that classifies data streams using trees in combination with gradient boosting, which would be used to enhance the produced clustered structure fed into the boosting technique to enhance the output accuracy.

This hybrid model works like a typical flow chart model where the cobweb technique would first create the tree by creating classes and adding nodes to a class according to it's Category Utility(CU). This is repeated for the length of the data set and this tree is then fed into the gradient booster for enhancement of the classification. Gradient boosting creates trees in a stage-wise manner using the weaker associations and trains the algorithm repeatedly to improve the classification model.

7. Architecture Diagram

Fig -1: Architecture diagram representing the flow of the algorithm



8. MODULES

Module 1: Feature Extraction

Feature extraction is the method of reducing dimensions of the data at hand. This module will single out every part of the tweet and label them separately. For example, all the characters after @ would be stored under a single username and labeled. Some other features of the tweet include hashtags, links in the tweet, mentions, etc.

Module 2: Matrix Creation

Matrix creation is the most crucial part of a big data working solution. This module creates a matrix out of the labeled data from the previous module and feeds it into the main for prediction after the training of the algorithm is complete.

Module 3 : Training

The data from the complete dataset is divided into parts for training and testing. The bigger chunk of data is often used for training the algorithm. Hence, this module of the project is concerned with training the algorithm or better functionality.

Module 4 : Testing

After the training of the algorithm, it is necessary to test the data on the remaining portion of the dataset that was saved for testing. Hence, this module is designed for testing the trained algorithm to check if it gives the desired results.

Module 5 : Prediction

This module is concerned about prediction or final outcome of the entire project. It creates a classification report and lists out the spam tweets and account IDs.

9. OUTPUTS AND RESULTS

The parameters used can vary greatly in the range of values that the algorithm can use. Hence it is important for them to be reduced to a common range of values called scaling to be implemented throughout. The default range for Cobweb clustering in 0.5. However it does not yield the correct results in all cases and needs to be changed accordingly. Experimentally, different values of scaling were used against different numbers of inputs and the results were compiled into a table, as given below in Fig-2.

Fig -2: Table with scaling values

	1	10	100	1000
1	(1.0, 0.00032401 084899902344)	(0.2, 0.00804615 0207519531)	(0.58, 0.311749 4583129883)	(0.689, 8.19210 6008529663)
3	(1.0, 0.00022315 97900390625)	(0.2, 0.00763249 397277832)	(0.78, 0.346787 9295349121)	(0.797, 8.16788 625717163)
5	(1.0, 0.00022244 45343017578)	(0.2, 0.00748491 2872314453)	(0.78, 0.379726 8867492676)	(0.826, 8.93060 8034133911)
7	(1.0, 0.00022554 397583007812)	(0.2, 0.00780773 1628417969)	(0.77, 0.385082 2448730469)	(0.823, 9.65178 3227920532)
9	(1.0, 0.00024557 11364746094)	(0.2, 0.00755023 9562988281)	(0.77, 0.395266 2944793701)	(0.827, 11.4313 10653686523)
11	(1.0, 0.00023174 285888671875)	(0.2, 0.00808215 1412963867)	(0.77, 0.390966 65382385254)	(0.831, 11.6969 8429107666)
13	(1.0, 0.00023722 64862060547)	(0.2, 0.00780701 6372680664)	(0.76, 0.392870 9030151367)	(0.83, 13.66006 5174102783)
15	(1.0, 0.00022864 341735839844)	(0.2, 0.00757288 932800293)	(0.76, 0.393646 0018157959)	(0.838, 14.6246 80519104004)
17	(1.0, 0.00023818 016052246094)	(0.2, 0.00808715 8203125)	(0.76, 0.378375 768661499)	(0.836, 12.5460 02388000488)
19	(1.0, 0.00022602 081298828125)	(0.2, 0.00787520 4086303711)	(0.76, 0.393804 31175231934)	(0.834, 13.9925 37498474121)

The column heads contain the number of inputs during every iteration during the testing. First row heads contain the scaling values used for every iteration.

The cells contain 2 values, the accuracy and time taken, in that order, for the corresponding values of scaling and number of inputs.

For a particular number of inputs, the accuracy increases as the value of scaling increases. It stops increasing after scaling settles down on a particular value. Similarly, time taken increases as the scaling value increases. Evidently, it is a logarithmic function, the relation of scaling(s) and number of inputs(n). It is, therefore, optimal to use the value which gives the maximum accuracy but takes the minimum time.

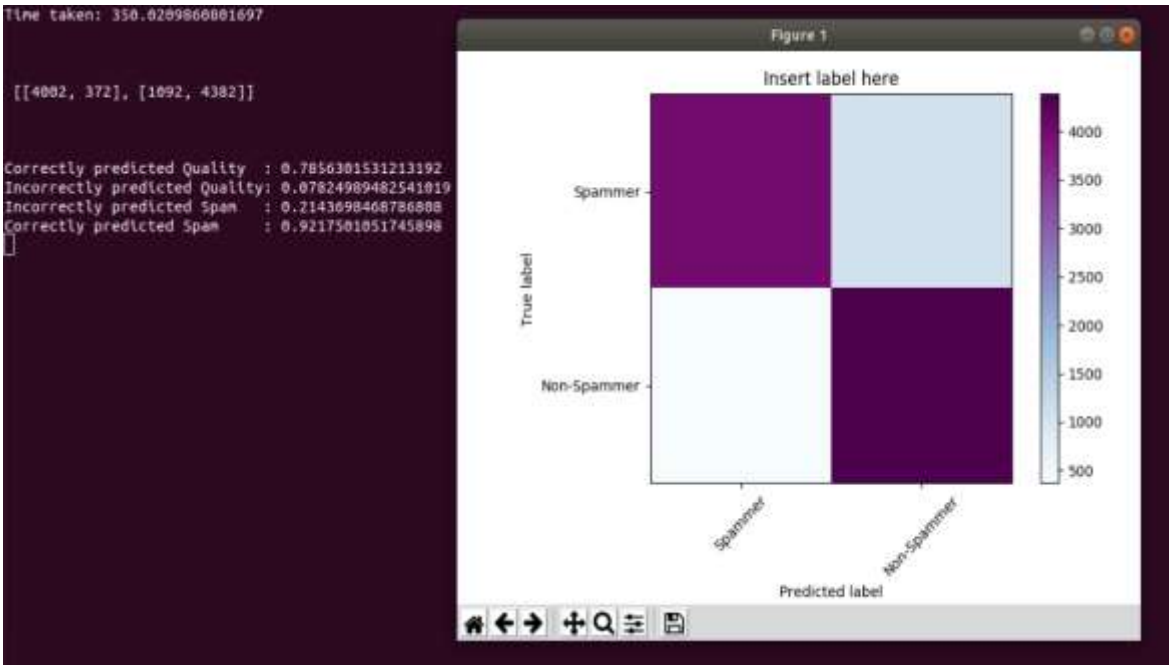
It is found to be:

$$s = 2*(log\ n) - 1 \quad (1)$$

In addition, the other parts of the results obtained can be divided into two sections namely graphic and declarative.

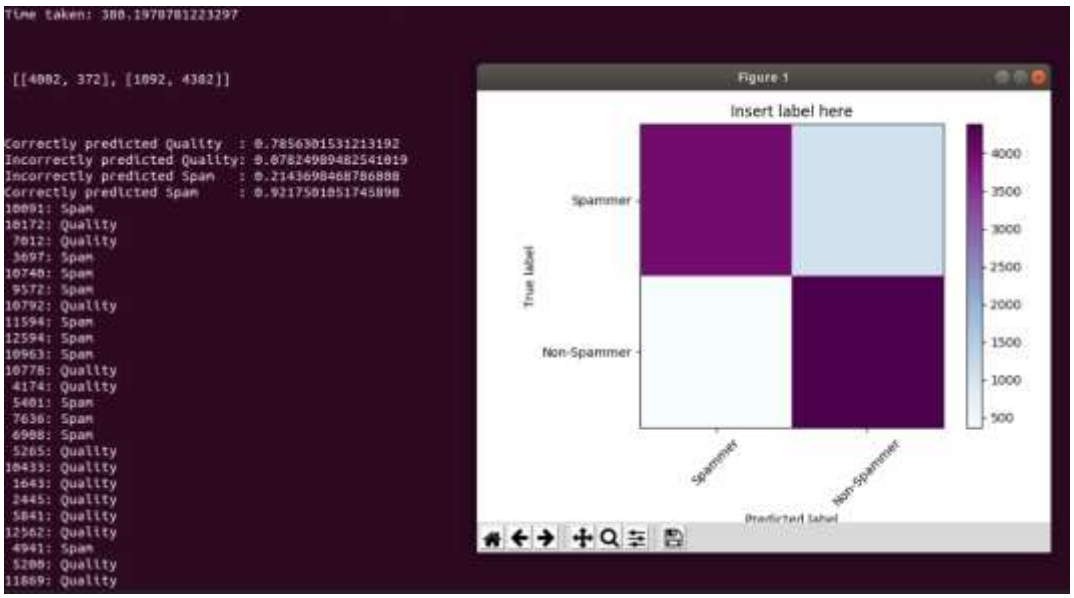
The graphic part of the output depicts the spam to not spam ratio, mostly depending on the amount of training given to the data. This also brings statistical data to the output which can be used for comparison as shown in Fig-3.

Fig -3: Graphical Output



The plain declarative part of the output depicts the tweet id and in text, whether the tweet is spam or not, as depicted in Figure 4. This is a rather tedious output to read due to the high amount of tweets in the data set.

Fig -4: Declarative Output



10. CONCLUSION

Upon running the hybrid model for spam detection on the testing data, the desired output of tweet id and the declaration of spam or not spam is achieved. The acquired value of f measure, which is dependent on the spam to non-spam ratio, is found to be at par with the existing models. Hence, it is safe to conclude, this is an alternate yet successful hybrid model to detect spam on twitter.

11. REFERENCES

[1] A Novel Stream Clustering Framework for Spam Detection in Twitter; Hadi Tajalizadeh and Reza Boostani; IEEE TRANSACTIONS ONCOMPUTATIONAL SOCIAL SYSTEMS, VOL. 6, NO. 3, JUNE 2019

[2] A Topic-Based Hidden Markov Model for Real-Time Spam Tweets Filtering; Mahdi Washha, Aziz Qaroush, Manel Mezghani, Florence Sedes; 21st International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2017, 6-8 September 2017, Marseille, France

[3] Spam Detection Techniques for Twitter; Rutuja Katpatal, Aparna Junnarkar; International Research Journal of Engineering and Technology (IRJET), Volume 05, May 2018.

[4] Design of Machine Learning Approach For Spam Tweet Detection; Miss. Shukla Twinkle Kailas, Prof.D.B.Kshirsagar; IJARIIE-ISSN (O), Volume 3, May 2018.

[5] Fast Asynchronous Anti-TrustRank forWeb Spam Detection; Joyce Jiyoung Whang, Yeon Seong Jeong,Inderjit S. Dhillon, Seonggoo Kang, Jungmin Lee; MIS2, Marina Del Rey, CA, USA, Volume 02, Issue 5, April 2016.

[6] O. Kurasova, V. Marcinkevicius, V. Medvedev, A. Rapecka, and P. Stefanovic, “Strategies for big data clustering,” in Proc. IEEE 26th Int. Conf. Tools Artif. Intell., Nov. 2014, pp. 740–747.

[7] Twitter spam detection; Shradha Hirve,Swarupa Kamble ; IJESC,49-57, April 2017.

[8] Constraine d NMF-base d semi-supervise d learning for social me dia spammer detection ;Dingguo Yu, Nan Chen, Frank Jiang,Bin Fu, Aihong Qin; Elsevier-2018.

[9] Early filtering of ephemeral malicious accounts on Twitter; Sangho Lee, Jong Kim; Elsevier, Issue 48-57, 2015.

[10] Twitter Spam detection; Shradha Hirve,Swarupa Kamle; IJESC-2016.

[11] Twitter turns 7 : User sends over 400 million tweets everyday; H. Tsukayama; Washington Post(2013)

[12] Twitter spammer detection using data stream clustering ; Z. Miller, B. Dickinson, W. Deitrick, W. Hu, and A. H. Wang, April-2017.

[13] Suspended accounts in retrospect: An analysis of Twitter spam;K. Thomas, C. Grier, D. Song, and V. Paxson(2013).

[14] Detecting Spammers on social media; G. Stringhini, C. Kruegel, and G. Vigna(2015)

[15] An in-depth analysis of abuse on Twitter; J. Oliver, P. Pajares, C. Ke, C. Chen, and Y. Xiang,

[16] Semi-supervised spam detection in Twitter stream; S. Sedhai and A. Sun; IEEE Trans. Comput. Social Syst., vol. 5, no. 1, 2018

[17] A performance evaluation of machine learning-based streaming spam tweets detection; C Chen el at.; IEEE Trans. Comput. Social Syst., vol. 2, no. 3, 2015

[18] Twitter spam detection based on deep learning; T. Wu, S. Liu, J. Zhang, and Y. Xiang, Elsevier-2017

[19] Statistical Features-Based Real-Time Detection of Drifted Twitter Spam; Chao Chen, Yu Wang, Jun Zhang, Yang Xiang, Wanlei Zhou,Geyong Min; IEEE Ttransaction on Information Forensics and Security, VOL. 12, NO. 4, APRIL 2017

[20] Detecting spammer on twitter; F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida; Proc. 7th Annu. Collaboration, Electron. Messaging, Anti-Abuse Spam Conf, 2010

[21] What is twitter, a social network or a news media; H. Kwak, C. Lee, H. Park, and S. Moon; Proc. 19th Int. Conf. World Wide Web, 2010

[22] A Survey of Spam Detection Methods on Twitter; Abdullah Talha Kabakus and Resul Kara, International Journal of Advanced Computer Science and Applications, Vol. 8, No. 3, 2017

[23] Investigating the deceptive information in Twitter spam; Chao Chen et al. ; Future Generation Computer Systems (2017)

[24] WarningBird: A Near Real-Time Detection System for Suspicious URLs in Twitter Stream; Sangho Lee and Jong Kim; IEEE transactions on dependable and secure computing - 2016

[25] Reuters tracer: A large scale system of detecting & verifying real-time news events from twitter; Xiaomo Liu et al.; 25th ACM CIKM. 20

AUTHORS

Vidhi Tiwari is a fourth-year student currently pursuing her bachelor's degree in Computer Science Engineering from SRM Institute of Science and Technology. She has completed courses like Web development in Php and Data Science in R. Avid reader and learner, she is also interested in writing.

Akarsh Srivastava is a fourth-year student currently pursuing his bachelor's degree in COmputer Science from SRM Institute of Science and Technology. An active participant in the college club "Humanoid", he has handled the coding department of the same, leading him to take part in hackathons. He has completed courses such as Machine learning and is interested in deep learning and neural nets.

Mrs. P. Akhilandeswari holds the position of Assistant Professor at SRM Institute of Science and Technology in the department of Computer Science till date. She has had over 15 years of experience. Her interests in computing lie in IOT, data science and big data, and cloud computing.