Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

FAKE ACCOUNT DETECTION USING MACHINE LEARNING

K.Swathi (19NP1A1209), Ch.Navya (19NP1A1203), S.Bhavani (19NP1A1221), V.Yasaswi (19NP1A1225), Btech Student of IT Dept, , Vijaya College, Vijawada, Andhara Pradesh
Email:-swathikonakanchi0127@gmail.com
Mr.Y.sirisha, Assistant Professor **,** IT DEPT, Vijaya College, Vijayawada, Andhara Pradesh
Email:- yarlagaddasirisha1@gmail.com

Abstract
Social Networks are gaining more momentum in businesses around the world and has become one of the most used and popular platforms of digital marketing and to check the latest trends among the public and to better understand what people wants. Fake Social Profiles are increasing rapidly that spreads fake news and information over this growing channel. This paper looks at different machine learning algorithms and how they help to solve the problems related to fake social profile detection. Python is used in Jupyter Notebook along with various ML and data analytics library like Pandas, Sklearn, Numpy etc. Machine learning algorithms i.e. ANN is used in this paper and Genuine Users are detected.

Index Terms: - social networks, fake account, sklearn, numpy, ANN

I Introduction

Spam is a real threat to usefulness of the web. Spammers mask their content as useful or relevant content and hence is delivered to the user. The legitimate users consume this spam data considering it relevant to their information needs. Clay Shirky remarked that a communication channel isn't worth its salt until the spammers descend.

Spams are not easy to stop. For several years, email services like Gmail, Microsoft and others have been successfully detecting spam emails but still spam emails are in circle on the web. These services have been reporting that email spamming has been up to 90 to 95 percent of the total email exchanges. Even after successful detection of spams, companies are unable to stop spammers which ensures about the economic benefits spammers get when they trap a user clicking on a spam link. The severity of the threat posed by spamming has increased with the emergence of online social networks and twitter is one of the most popular online social network which has been highly affected by spam. twitter spamming is more threatening because its more targeted towards the trending topics of the twitter and hence bit easier to get penetrated especially because of hash-tag operator. Another fact that makes twitter a rather easier and fruitful target for spammers is its variety of audience. twitter users span across all sectors of life i.e. it can be the teachers or students, celebrities or politicians, marketers or customers or even general public. They belong to all age groups but most widely age group that uses twitter is between 55 to 64 years. There are about 60% users that access twitter from their cell phones. Twitter has 288 million monthly active members that make it widely growing social networking site. There are around 400 million tweets posted on daily bases, the average posts on twitter is 208 tweets per users account.

Due to this continuous distribution of information, a user faces many problems with search results that shares recurring and irrelevant information. This also can be very worrying at the times since a user has to scroll through the all information in direction to get an overall view of topic. Spam detection on the twitter network is difficult due to the noticeable usage of URLs, abbreviations, informal language and modern language concepts. Old-style methods of detecting spam information fall

short here. To date, study has been available on many techniques for detecting spams on twitter and blogs by using different features. After knowing the existing importance of spams on twitter, we take inspiration or motivation from

this user need and decided to design and develop improved techniques to detect spams on twitter. In this paper, we propose a spam detection approach for detecting spam tweets. This approach is based on sentimental features of a tweet. The idea is to exploit the philosophy that spammer use to force a user to click on a particular link. They definitely seek help of some motivational words (like 'the best web site', 'excellent service', etc) to make people believe in a certain tweet (examples of some spam tweets given in the table I. Results show that this exploitation of sentimental features proves fruitful.

Another approach is discussed for spam detection in twitter network. They study the propagation of spam in the network. And they want to find out whether there is a pattern that spammers used for spam proliferation through the network and to determine whether the accounts are either been compromised or overtaken by spammers or certain accounts are purely created for spam activities in the network. They examine the characteristics of the graph of spam tweets and run Trust Rank technique on the collected data. Also introduced is the features for spam tweets detection without earlier statistics of the user and use statistical presentation for the analysis purpose of language to identify spam in twitter topics.

.

2 Literature survey

Prior to 2004, email spam classification research was suffering from considerable diversity and controversy in the methods employed for spam filtering and the ways by which these methods were evaluated. It was unclear, which method was best and showed promise for improvement. Three different communities were focussing on these issues (Lynam, 2009):

- The community of developers and practitioners with the motive of developing tools for instantaneous deployment;
- The community of spam filter vendors with the motive of selling spam filters;
- The community of researchers with the motive of inventing new facts and validating existing theories and algorithms.
- Several spam filtering methods were experimented and investigated by users, practitioners, vendors and researchers and classified into three groups:
- Manual inspection,
- System oriented approaches,
- Content-based filtering.

Apart from all, Content-based filters can be further classified as -

- Ad-hoc Rule-based filters,
- Practical learning filters,
- Machine learning research.

Manual Inspection:

Email spam can be filtered by manual inspection which is one of the alternatives of automatic spam filtering. In this mechanism, an end user examines each incoming message and identifies whether it is spam or not. Such filtering always results some cost such as huge time consumption, and difficult to quantify easily (Yerazunis 2002). Alike spam filters, manual inspection is also not free from the

risk of errors.

A user may believe that he has done a better job by manual inspection. A study done by Yerazunis (2004) confirms that the error rate from manual inspection is high. In his study, he has taken a collection of emails and examined the full email (Header and body) on two different occasions and found that the disagreement rate was 0.16%. Actually, the average human error rate was much higher. Manual inspection is worthy only when spam is
infrequent. As soon as spam increases, workload and mistakes also increase. Another disadvantage of manual inspection is deletion of some important emails which are misjudged by users. The research done by Hidalgo in 2002 addresses the issues of manual inspection based filter.

For spam detection, System approaches work on the information extrinsic to the message and user. These approaches are applied before delivery of the message to the end user. Some of common methods used in this technique are, good senders list (white lists), bad senders list (black lists), or particular spam messages list (fingerprint lists). These lists are established and maintained by network administrators in collaboration with the end user who contribute to discover elements of the lists.

Apart from the methods mentioned above, Greylisting (Levine 2005; Harris 2009) captures some particular behaviour of the sender and assumes that a message is spam if this behaviour is absent. Such method introduces some cost like delay in delivery, additional network traffic and risk of message loss. It is difficult to measure the performance of such systems because they work in real-time and dynamic environment.

A white-list is identified as the list of senders (users, domains and IP addresses) which includes safe addresses that have never been used for spam sending. The incoming message that comes through the white list is classified as legitimate or ham. The problem with this technique is that the sender is always assumed to be ham when it comes from white-list so that spammers can easily spoof these white list addresses to send spam. This problem has been addressed (Leiba, Ossher, Rajan, Segal, & Wegman, 2005) by notifying that it is easy for a spammer to spoof the sender ID which is used for classifying incoming message as ham.

A black list (Cole, 2007; Micro, 2005) differs from a white list where a list of bad senders (who use their IP address for sending spams) is maintained to classify incoming messages. The complication from this method is that spams can be sent from a number of sources and it is difficult to maintain a complete effective blacklist.

Another system approach is collaborative filtering (Prakash, & O'Donnell, 2005; Kołcz, Chowdhury, & Alspector, 2004; Kołcz, & Chowdhury, 2007; Dimmock, & Maddison, 2004; De Guerre, 2007) which exploits the fact that similar email spam is sent to many end users. It
captures email spam for identifying the redundancy over many systems. If the message is received from email addresses that have never used for legitimate emails, it will consider as spam. Due to the bulky nature of spam email, it is difficult to store all messages. As the number of message will increase, decisions will be more complicated and time consuming.

Blanzieri, & Bryl in 2007 captures false positive and false negative values for measuring the performance where this classifier is predicted to be excellent. Another research (Etzold, 2013) combines *k*NN and Bayesian classifiers where the results were good. In Addition, some research reported in the literature (Soonthornphisaj, Chaikulseriwat, and Tang-On 2002; Bashiri, Oroumchian, and Moeini, 2005; Chan, Tony, Jie and Zhao.) show the weak performance of this classifier.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

3 Implementation Study

investigated issues of detecting spammers on Twitter. The proposed method combines characteristics withdrawal from text content and information of social networks. The authors used matrix factorization to determine the underline feature matrix or the tweets and then came up with a social regularization with interaction coefficient to teach the factorization of the underline matrix. Subsequently, the authors combined knowledge with social regularization and factorization matrix processes, and performed experiments on the real-world Twitter dataset, i.e., UDI Twitter dataset.

Washha *et al.* [31] described the Hidden Markov Model for filtering the spam related to recent time. The method supports the accessible and obtainable information in the tweet object to recognize spam tweets and the tweets that are handled previously related to the same topic.

Jeong *et al.* [17] analyzed the follow spam on Twitter as an alternative of dispersion of provoking public messages, spammers follow authorized users, and followed by authorized users. Categorization techniques were proposed that are used for the detection of follow spammers. The focus of the social relation is cascaded and formulated into two mechanism, i.e., social status filtering and trade significance profile filtering, where each of which uses two-hop sub networks that are centered at each other. Assemble techniques and cascading filtering are also proposed for combining the properties of both trade significance profile and social status. To check whether a user is fake or not, a two-hop social network for each user is focused to gather social information from social networks.

Meda *et al.* [21] presented a technique that utilizes a sampling of non-uniform features inside a machine learning system by the adaptation of random forest algorithm to recognize spammer insiders. The proposed framework focuses on the random forest and non-uniform feature sampling techniques. The random forest is a learning algorithm for the categorization and regression that works by assembling several decision trees at preparation time and selecting the one with the majority votes by individual trees. The scheme integrates bootstrap aggregating technique with the un-planned selection of features.

3.1proposed methodology

In the proposed system, the system elaborates a classification of spammer detection techniques. The system shows the proposed taxonomy for identification of spammers on Twitter. The proposed taxonomy is categorized into four main classes, namely, (i) fake content, (ii) URL based spam detection, (iii) detecting spam in trending topics, and (iv) fake user identification. Each category of identification methods relies on a specific model, technique, and detection algorithm.

The first category (fake content) includes various techniques, such as regression prediction model, malware alerting system, and Lfun scheme approach. In the second category (URL based spam detection), the spammer is identified in URL through different machine learning algorithms. The third category (spam in trending topics) is identified through Naïve Bayes classifier and language model

$$P(Y \mid X) = \frac{P(Y)\prod_{i=1}^{d} P(X_i \mid Y)}{P(X)} \quad (2)$$

divergence. The last category (fake user identification) is based on detecting fake users through hybrid techniques.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

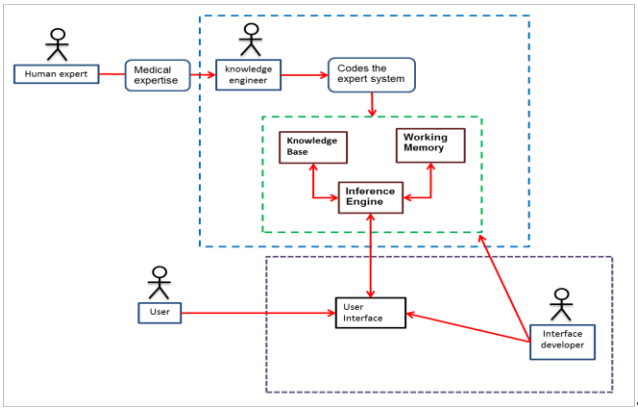

Fig 1: - proposed model

The proposed approach is divided into three stages

Spam Detection

a. Spam Detection Based on the above identified features, we proceed to use traditional classifiers to help detect spammers. In this work, several classic classification algorithms such as Random Forest, Naïve Bayesian, Support Vector Machines, and Knearest neighbours are compared. The Random Forest classifier is known to be effective in giving estimates of what variables are important in the classification. This classifier also has methods for balancing error in class population unbalanced data sets. The naïve Bayesian classifier is based on the well-known Bayes theorem. The big assumption of the naïve Bayesian classifier is that the features are conditionally independent, although research shows that it is surprisingly effective in practice without the unrealistic independence assumption. To classify a data record, the posterior probability is computed for each class

Is a normalized factor which is equal for all classes, only the numerator needs to be maximized in order to do the classification for the Naïve Bayesian classifier. The Support Vector Machine method we used is the SMO scheme implemented in the Python progaming. This SMO scheme, designed by J.C. Platt [16], uses a sequential minimal optimization algorithm to train a support vector classifier using polynomial or RBF kernels. The SMO classifier has been shown to outperform Naives Bayesian classifier in email categorization when the number of features increases. The K-Nearest Neighbour method implemented in the Python progaming is the IBK classifier.

Data Collection

We downloaded tweets from an online source using **Twelts** which download the data in a csv file and covert it to txt. It gives us the list of all followers, followings, tweets of the particular selected account. After basic pre-processing, we are left with around 70k tweets which are classified into following (a) Legit Users (b) Legit User Tweets (c) Spammer Tweets (d) Spammer Users. Manual annotation of these tweets was done with spam or not-spam labels using two annotators A and B. Kappa score for this annotation was found satisfactory (0.82) to proceed with the experiments. We decide to use standard metrics for measuring the usefulness of our approach and hence precision, recall, and F-measure are used.

a. Features Performance Comparison

Here we will discuss our proposed features spam detections performance by using five selected

classifiers (SVM, Random Forest, Naive Bayes, Bays Network and J48). We have compared the performance of different features by making different combinations, We have discussing just one combination" all proposed features with baseline features combination"

2 ALGORITHMS

Spam Filter Algorithm Steps

- **Handle Data**: Load the corpus file and split it into training and test datasets.
- **Summarize Data**: summarize the properties in the training dataset so that we can calculate probabilities and make predictions.
- **Make a Prediction**: Use the summaries of the dataset to generate a single prediction.

Make Predictions: Generate predictions given a test dataset and a summarized training dataset.
Evaluate Accuracy: Evaluate the accuracy of predictions made for a test dataset as thepercentage correct out of all predictions made.
Tie it together: Use all of the code elements to present a complete and stand alone implementation of the Naive Bayes algorithm.

Naïve Bayes Classifier

The Naive Bayes algorithm is a simple probabilistic classifier that calculates a set of probabilities by counting the frequency and combination of values in a given dataset [4]. In this research, Naive Bayes classifier use bag of words features to identify spam e-mail and a text is representing as the bag of its word. The bag of words is always used in methods of document classification, where the frequency of occurrence of each word is used as a feature for training classifier. This bag of words features are included in the chosen datasets.

Naive Bayes technique used Bayes theorem to determine that probabilities spam e-mail. Some words have particular probabilities of occurring in spam e-mail or non-spam e-mail. Example, suppose that we know exactly, that the word Free could never occur in a non-spam e-mail. Then, when we saw a message containing this word, we could tell for sure that were spam spam users. Bayesian spam filters have learned a very high spam probability for the words such as Free and Viagra, but a very low spam probability for words seen in non-spam e-mail, such as the names of friend and family member. So, to calculate the probability that e-mail is spam or non-spam Naive Bayes technique used Bayes theorem as shown in formula below.

Where:

(i) P(spamword) is probability that an e-mail has particular word given the e-mail is spam.
(ii) P(spam) is probability that any given message is spam.
(iii)P(wordspam) is probability that the particular word appears in spam message. (iv)P(non − spam) is the probability that any particular word is not spam.
(v) P(wordnon − spam) is the probability that the particular word appears in non-spam message.

To achieve the objective, the research and procedure is conducted in three phases. The phases involved are as follows:

1. Phase 1: Pre-processing
2. Phase 2: Feature Selection
3. Phase 3: Naive Bayes Classifier

The following sections will explain the activities that involve in each phases in order to develop this

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

project. Figure 2 shows the process for e-mail spam filtering based on Naive Bayes algorithm.

4 Results and Evolution Metrics

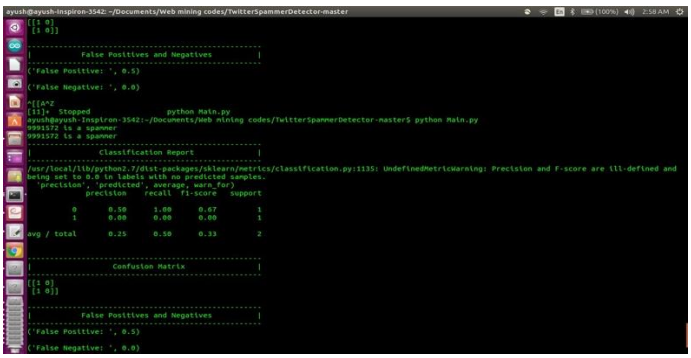

Fig 2:.evluation metrics of the algorithm and dataset

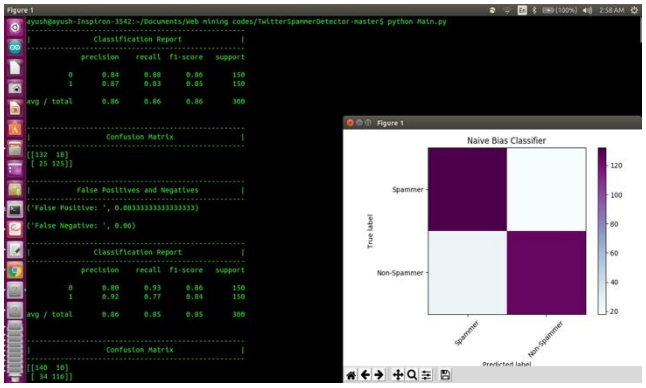

Fig 3:-navie bayes confusion matrix report

5 Conclusion

In this project, we have suggested some user-based and content-based features that can be used to distinguish between spammers and legitimate users on Twitter, a popular online social networking site. These suggested features are influenced by Twitter spam policies and our observations of spammers' behaviours. Then, we use these features to help identify spammers. We evaluate the usefulness of these features in spammer detection using traditional classifiers like Random Forest, Naïve Bayesian, Support Vector Machine, K-NN neighbour schemes using the Twitter dataset we have collected. Our results show that theRandom Forest classifier gives the best performance. Using this classifier, our suggested features can achieve precision and F-measure as we have mentioned in the images. Based on our dataset, our features provide slightly better classification results. Our next step is to evaluate our detection scheme using larger Twitter dataset as well as possibly wall-post datasets from other online networking sites like Facebook. We also hope to include the content similarity metric in our near future work.

6 References

1. How to;5Top methods & applications toreduce Twitter
2. Spam http://blog.thoughtpick.com/2009/07/how-to-5-topmethods-applications-to-reduce-twitter-spam.html
3. Rish. “An empirical study of the naïve bayes classifier”. Proeedings of IJCAI workshop on Empirical Methods in Artificial Intelligence, 2005.
4. L. Bilge et al, “All your contacts are belong to us: automated identifty theft attacks on social networks”, Proceedings of ACM World Wide Web Conference, 2009.
5. T.N. Jagatic et al, “Social Phishing”, Communications of ACM, Vol 50(10):94-100, 2007.
6. S. Yardi et al, “Detecting Spam in a Twitter Network”, First Monday, Vol 15(1), 2010.
7. G. Stringhini, C. Kruegel, G. Vigna, “Detecting Spammers on Social Networks”, Proceedings of ACM ACSAS’10, Dec, 2010.