International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-7, Issue-6, March 2019

MONO-Spam: An Intelligent Spam Detector Based On Natural Language Processing

Eshwar.S, Lavanya. K

Abstract: ***With the evolution of "social" world, people produce a lot of data. Data is being produced everywhere without the inherent knowledge of the people. And, with the incremental usage of social media and e-commerce sites etc., a user produces and consumes a lot of data. The 'data' referred to here is not the bandwidth but the text. This text can be in the form of comments, reviews, emails, names, identities, birth dates, offers, claims etc. The problem here is the integrity of data and where its end point is and the sanity. Integrity, although solved by cryptography algorithms, the sanity is always a question mark. Checking if a data is clean is the most crucial part or else a lot of space and valuable resources are wasted. In this paper, we provide a novel way of using Natural Language Processing and Multinomial Naive Bayes algorithm to filter spam before insertion. The model filters spam with an accuracy of about 96 percent***

Index Terms: Spam Classifiers, Natural Language Processing, Bag Of Words, TF-IDF, Corpus, Multinomial Naive Bayes classifier

I. INTRODUCTION

The quantity of data that an user produces on a daily basis is an estimate that is prodigious. There are about 2 quintillion bytes of data created every day by the people. The sources of these data come from e-commerce, promotion, e-marketing, computers, branding, services, projects, statistics, apps, organizations, IT sector, social network, media etc. Another important aspect is searching. There are nearly 4 billion searches made on popular sites like google and Duck Duck Go per day.

As per statistics, every 60 seconds,

- Snap users post 500,000+ photos
- 100+ new LinkedIn accounts
- YouTube videos -4,000,000+
- Instagram photos - nearly 50,000

We can interpret a lot of questions from the above survey. Why is so much data produced? What are the new protocols that need to be designed to store and preserve these data? Which among these data can be completely formatted to get more user space? How much of this data is really meaningful ?and so on. The answer to 'Which of these data can be formatted?' is spam detection and filtering based on Natural language processing. Humans have crossed the days of comprehending spam from just seeing and hence we need the help of machines. Spam messages, if accessed can lead to cyber attacks like computer vandalism, which can in turn lead to Cyber Defamation. In this paper, we realize that the meaning of the word spam has changed and use NLP techniques to prevent it and in turn stop the consequences Spamming can cause.

A. Business

With the advent of a generation where a critical amount of critical business is all done via the internet, the importance of Spam detection and it's filtering and the security of data has a much higher quantitative risk to address. Nowadays the word spam means no longer just spam, a spam message contains threat, possibly to an extent where it can damage a company's profit by more than a half percentage. It's no more just an e-garbage that one can choose to delete.

A business can have employees and even if one employee is careless and clicks on a "Threat Accompanied Spam", access to client's private data, the company's private data becomes a big loophole. This can in turn affect the business reputation and bring down the organization's value.

Ways how NLP based spam detection can help a business:

- Block potential threats because, in this paper, the algorithm is intelligent enough to guess whether a block of text is spam or not
- Assurance of legit because, the spam filter has no effect of legitimate messages, and users can read them.

B. NLP AS A TOOL

Unlike other Machine Learning Prediction problems where features are recorded from some dataset and the analyst makes a statistical approach to decide which set of features could be taken valid and which feature could be taken as label, Unfortunately, Spam detection does not respond well to these methods. It involves language that is used by humans in daily life. It takes a considerable amount of time for humans to learn a language and then to process it and in turn find the meaning of a sentence. In a spam text, there might be some word that contributes a lot to make a decision that the word is spam, and the same word could be in a hundred percent pure text which is not a spam. The problem starts here, where only a set of features cannot decide what a word in a sentence would really mean. This domain is filled by Natural Language Processing. The inherent combination of Natural Language processing and Machine Learning is emerging as a new Cyber Security Technique. This has been around for years combining the art of CS, language, and AI. It makes the language manipulable and modifiable, by using various techniques like lemmatization, pluralization etc.

Revised Manuscript Received on 30 March 2019.
* Correspondence Author
Eshwar. S, Private university in Vellore, (Tamil Nadu), India
Lavanya. K, Private university in Vellore, (Tamil Nadu), India

There a few concepts that are used in our way of spam detection. Sentences, words make the heart of any NLP library. This film is rubbish, also, watch my movie at this theatre SPAM The films screenplay is not that good, one time watchable Not a spam Here, spam and not spam is

C. NLI - Natural Language Interpretation

As mentioned earlier in the paper, language comprehension as done by humans, if done by the computers, there can be a dominant change in the world of spam, phishing and hacking. NLI being a subset of NLP and AI deals with this domain. It is used as a chatbot, translator, FAQ. In this paper, we use it for spam detection.

II. LITERATURE SURVEY

Spam detection can be mapped to a human's interactive dialogue. Spam can be assumed as a prank by humans. This is mentioned as a challenge by Jaime et al[1] where they have mentioned Applications where one accesses the database, where humans have the problem solving and decision making abilities is still a miracle to scientists as they are recognizing how the neurons work. Another concern is Out of Vocabulary words (OOV)[2] as mentioned by Tom Young et al. Previously the evaluation of cards and batch processing of human language was nearly 7 minutes and now Google search returns results less than a second. This has been the growth of Deep Learning based NLP[3]. Word tagging, machine language translation and POS (part-of-speech) tagging are all turning out to be an inherent capacity that the field of science has achieved. Works have been carried out in the field of NLP where Neural Networks are used which makes faster computation involving automatic feature extraction which is totally in different to the manual feature extraction, where the users have to analyze the data and handcraft the solutions and make predictions based on what they want to represent. Though RNN models have proved to be having a more accuracy[2], this is negligible for NOSQL databases. Deep Neural Networks can be used while working with a large amount of data like big data. Previously Spam filtering was so simple that an user, just be looking at the text can recognize that it the text is a spam and further action can or cannot be taken, or a repetition or a sequence of messages with the same pattern from the same IP address or a cluster of IP addresses, or certain typical attributes can contribute to the message being spam or not. But days have changed and hackers and advertisers have evolved, like how certain types of viruses have adopted to the likes of many anti viruses, the way of spam texting has also changed. Spam nowadays seem more saline than messages that are really considered to deliver the meaning of a message. After the Machine Learning era, Spam detection eventually turned out to be an automated task where the message is translated into a set of feature vectors and this vector is applied on by Linear or Logistic regression and the output is predicted[4](Fig.1) The google data center applies 100s of conventions before an mail of text reaches in your inbox there

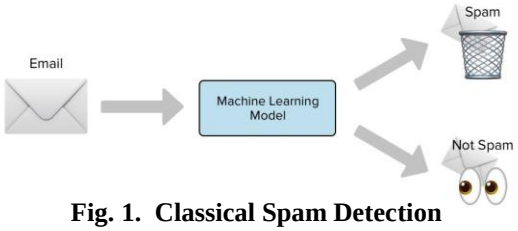

Fig. 1. Classical Spam Detection

by doing the spam detection process at the data centers[4]

TYPES OF SPAM FILTERING INCORPORATED BY GOOGLE[8]

- **Null Sender Dispose** - A user shall not send a message to an email address that does not exist, or does not attach a Simple Message Transfer Protocol
- **Blatant Blocking** - An email or text if detected as a spam, is deleted even before it is brought to the client
- **Category Filters** - An user shall decide which message or category of texts is considered as spam for him. This is akin to the dataset vendors labelling the dataset and the ML algorithms predict the outcome.
- **Bulk Email Filters** - Self descriptive. Bulk email sent from an IP is spam
- **Digital Signature Based** - Validate emails by authentication

There are several spam detection methods that has been incorporated by google, microsoft etc. Microsoft security patches are the new updates done to Windows Security most often, time to time. Here we review a few of the well established spam detection methods. List-Based Filtering There are ways where spam detection can be based on the sender. Certain IP addresses are black listed and those senders messages would not reach anyone's in-box. List based filtering also applies to categorizing an user as trusted or not. This can be user-based, where the user chooses from whom he receives a message from. This technique has been around for years, where a specific email id is specific to senders. Facebook's CEO Mark Zuckerburg first created his social media website which was specific to people with the Harvard attached email id. This kind of approach is called List based filtering and it is of two types.

- Black List Based
- White List Based

Black-List based Filtering This technique is similar to list based filtering but the difference is that this technique is not user based. There is a preset of Internet Protocol address set that is already held by the vendor or the organization. The true positives of this or Microsoft store and change his or her IP address. Though blacklist might later adapt to the vulnerability by newly added IPs but it not convincingly safe. White-List based Filtering This kind of filtering is very similar to user authentication and is exactly opposite to Black-List filtering where the user chooses to block senders, in white-box user chooses a preset of preferred senders. This might be a group on Google or a Facebook group or a Facebook friend list.

This type of listing is also done at the user level. Sometimes white list based filtering can be automated where the sender has a history stored in a database. The database has a history of the users past activities. If a sender sends a message and he/she does not have an active record of spam in the database, then the message is allowed. But some white lists are very restrictive that it does not allow any message out of the list. This kind of white list based filtering is called restrictive filtering and is very common nowadays because it is the concept used in social media and most of the organization. The users of the social media don't choose to block friends (which might happen after a friend request), but instead users add friends and they choose to receive messages from them.

Collaborative filtering This is the best manual user best filtering method as it can recognize spam within a few seconds of its outbreak, but this technique involves a very huge and active user base. A community of user groups starts marking an email or a sender or spam and if this is scaled to a larger group, then that sender is marked as spam. This approach is incorporated in the True Caller android app, which uses collaborative filtering (user based) to mark calls from a number as spam or not. The app is also transparent with it's system as it displays how many users have marked a particular number as spam or not. The flags sent by these users are called annotations and are labeled in a central database. But this approach is restricted to a community. The email is only restricted from the other members of the community. For example, only the users of true caller are alerted when they receive a spam call, for other users, this is a normal call. This is the only drawback in this approach, but yet it is quite effective. This type of filtering is also used in recommender systems.

Word based filtering This kind of approach is used from the old days. It is akin to the human approach. Any vulgar or harassing word is uttered in a text, it is blocked as spam. But this kind of approach is not practical nowadays as many friends use these kinds of words within their community. But those words compiled as sentence if detected as spam, the respective actions are taken. It is up to the users to either report the message or leave it aside. But spam types are not discussed yet, and those are not negligible either. Spam can affect our daily lives and can range from simple irritation like one-time-investment to a server crawl and other major virus attacks A survey done by the Federal Trade Commission has listed that generally spam might fall in the following sections and people should be aware of it(Fig 2). In this paper, we will address all these domains by means of natural language processing with the words of each sentence which in turns maximizes the chances of detecting the spam.

Type of Offer	Description
Investment/Business Opportunity	work-at-home, franchise, chain letters, etc.
Adult	pornography, dating services, etc.
Finance	credit cards, refinancing, insurance, foreign money offers, etc.
Products/Services	products and services, other than those coded with greater specificity.
Health	dietary supplements, disease prevention, organ enlargement, etc.
Computers/Internet	web hosting, domain name registration, email marketing, etc.
Leisure/Travel	vacation properties, etc.
Education	diplomas, job training, etc.
Other	catch-all for types of offers not captured by specific categories listed above.

Fig. 2. Sections of spams and it's impact on entities

III. PRELIMINARIES

A. An Overview Of Classifiers

Classifiers are machine learning algorithms that classify a problem based on a history dataset. It can be the use ofstatistics or probabilistic models. The concept of classificationis termed around random variables. A random variable is a numeric representation of real world entities. Let X denote the number of heads, then the probability of getting one head in two tosses would be

$$P(X=1) = 1 / 2$$

Random variables is very useful while making a vector representation of an element. Each random variable when assigned a probability gets a probability distribution table and this table is used to get the Probability mass function. Random variables can be continuous or discrete, for continuous variables we have Probability Distribution Function (PDF). Using this pdf we can derive the probability of the random variable X being anywhere between a predefined interval. Random variables form the central base for classification algorithm. The problem of spam detection comes under Supervised Learning approach where we have our history dataset and obtain a classifier for it.It is very important to note that our problem is not clustering. Classification approach is the concept where we try to make a prediction of the target class. This is done based on boundary conditions. The boundary conditions are obtained based on the datatset provided. The dataset contains the history of the data and their corresponding class labels and the analyst has to decide which class is the target class and which classes are the features that will in turn define the boundary conditions for the target class and in turn helps us to predict the target class. This is the basic approach of classification. In clustering we determine if a part of a data belongs to a community. There could be several clusters but classification is binary. Either spam or not spam. This approach is binary and here spam is not a community. Different countries can represent communities of people; this can be a clustering problem. Classification has evolved into different types where nowadays just making a yes or a no decision is not just classification. For example, classification of a fruit into fig, apple, melons etc. Therefore,

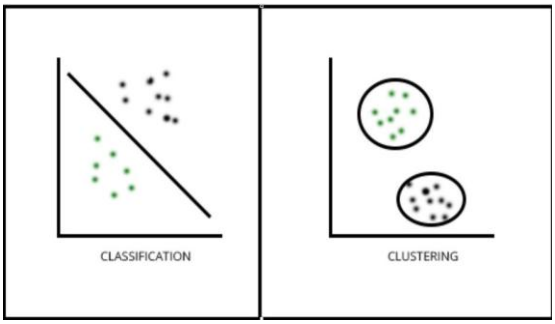

Fig. 3. Classification and Clustering

clustering is grouping data with similar characteristics and classification is predicting the target class. Our spam detection problem is more of a classification problem than a clustering problem because our target class here is either the text is spam or it is not spam. Also, detecting spam is done from a previous history of data where a text or sentence or line is labelled as spam or not spam. These informations leads us to a very important conclusion. Spam detection is a classification and supervised machine learning problem

B. Different Learning Algorithms

Over the last few years there has been a bucket of Learning algorithms that has evolved where the most famous and simplest one is linear regression. for Y = aX + b, linear regression uses the concept of a straight line equation to predict the output.

1) *Linear Algorithms*: Linear regression is a statistical approach to the prediction problem. It is shifted by the parameters a,b as shown in Fig 4. Regression analysis is the

$$a = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2}$$

$$b = \frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2}$$

Fig. 4. Equations to compute the constants of Linear Regression

process where we find the values of parameters that best fit the data. Given a problem, regression analysis is more a manual task where the user has decide which features would actually contribute to the continous output class. The data from the dataset i.e. Features are vectorized and the vector is used as X and Y values and using the concept of Multiple Regression, the prediction is obtained. In a simple linear regression model X is one vector. But with multiple regression X = {X1, X2, X3…..,Xn-1, Xn}, Y={ class }. *Linear regression is an algorithm that will help find out the values of Y given Y is dependent on X and Y is continuous.*It is very helpful in predicting the stock values, house price values, changing temperature, rate of change.

This cannot be used in Spam detection, as { spam, not Spam} is discrete.

Naive Bayes Classifier is another supervised learning algorithm that helps to predict the output class label. The Naive Bayes Classifier is not a regression analysis model. Here the outcome is a class label which is based on the Bayesian Theorem of conditional probability. P(A/B) is the probability of event A given the event B has occurred. Here the event B is called as the evidence event. B is referred to as evidence because only then, A has occurred. (Fig 5). The terminologies in Bayesian Classification involved the Feature matrix and Response vector.

The Feature Matrix comprises of the rows or vectors of the dataset which contains the values of features that are considered dependent. The Response vector is another vector or row that comprises of the value of the class variable for each of the row of the feature matrix. This can be a yes or no solution too. This kind of problems comes under decision making but can also be under classification considering 'yes' as one label and 'no' as another label. In this paper, in the methodology section the working of the proposed spam detector based on Naive Bayes and there are other linear classification models that are available too.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} = \frac{P(B|A)P(A)}{\sum_{i=1}^{n}[P(B|A_i)P(A_i)}$$

Fig. 5. The equations of the Bayes Theorem

Support Vector Machines: Support Vector Machine un-like regression and Naive Bayes, it used for both classification and regression problem, Support Vector Machines predicts the outcome of random experiments of a smaller dataset, but it is a much efficient way than the regression method. It is a supervised machine learning algorithm, Majority of the analysts use it for classification problems. Each data in a vector is given a point in an n-dimensional space. Here n will

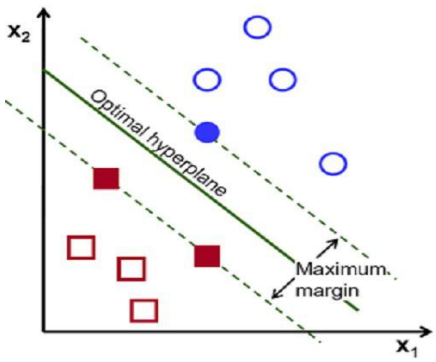

Fig. 6. The optimal hyper plane defined for classification two classes using Support Vector Machine

be the number of features that the analyst considers. The classification is done by plotting a corresponding hyper plane that has to the ability to segregate the given number of classes.

The goal of the SVM is to design an hyperplane that perfectly classifies the two classes, sometimes there can be more than one hyperplane that intersects with each other. The maximum margin of the hyperplane is calculated as the:

Closest Element of CLASS A - Closest Element of CLASS B

The hyperplane is not a line, it is rather a plane with the equation W exp T = X;

B. Issues in NLP : Semantics of Language

With the assumption that the message or text will bein English, the paper is focused on the semantics of English grammar, starting from words, sentence, tags, nouns, prepositions, Part Of Speech, verbs, adjective. The NLP libraries nowadays are intelligent enough to identify the role a word plays in a phrase. Research is being done in other languages like Korean (KONPLY), Chinese (Jieba) and Spanish (Pattern)

1) *Tagging:* The texts of the input vector are split at the spaces. The lines are split at the periods. This leads us to

2) *Sentences:* A sentence is made up of words. It is vital and critical to know that the meaning of the sentence is dependent of the words used. This leads us to the concept of POS. POS is the role of a word in a sentence. The word is tagged as a adjective, proper or common noun, descriptive, denotes action (verb), determiner, adverb etc. The POS of a word is not an attribute of the word. This is looped and again dependent on the sentence and other words. For example,

I hate you

Hate is very bad thing

The word ”hate” can be tagged as a verb in the first sentence and at the same time it can be tagged as a noun in the second sentence. The takeaway point here is, sentence is dependent on words, and the word is dependent on sentencewhich is nothing but a group of words. NLP languages make their best in tagging. A word is not tagged with verb and other semantics, rather they have

TABLE 1 : Subset of the NLP tags available

predefined set of tags, which is listed Above. (Table 1)

Tag	Description
CC	Conjuction, co-ordinating
JJ	Adjective
NN S	Plural Noun
FW	A foreign word
DT	A determiner word
PDT	Pre-determiner
NN	Singular Noun

2) *Phrases:* Phrases is a subset of a sentence. The sentence is actually subdivided into a subset which is also called units. These units are represented as a basic tree structure. The tree from top down forms a meaningful sentence, but at each level we can associate a ,meaningful phrase. It consists of the NP, VP, PP. Each tag is associated with a phrase that gives meaning to the phrase. For example, ”Eshwar” is a proper noun. ”He is Eshwar” is a phrase of nouns and verbs that adds meaning to the tag. Therefore, a phrase is the word and the words that is around the word which gives a meaning to the phrase. It can be a prep. adj. etc. Noun or the subject phrases matter the most in Natural Language Processing because they tell about the context of the sentence. Natural Language Processing libs provide tools to parse a sentence into the Phrase tree and thereby acquiring the noun phrase.

TextBlob - is the library used in this prepare to implement Spam detection in NOSQL Databases.

3) *Morphology*: Lexicographically, these are the words that are termed as creepy. Morphology of a word is the forms that it can take that can change the whole meaning thereby changing the meaning of the whole sentence. This has been one of the challenging parts of the Natural language processing algorithms. These are usually caused be prefix and suffix version of a given word. The prefix or suffix although being a 2 or 3 letter word can interchange the meaning of the whole sentence thereby heavily affecting the outcome of the random experiment. In spam detection morphology of a word plays a very major role because hackers, spammers use this loophole to parse through the spam filters. But over years, NLP libraries have learned to solve this. It has been found that finding the lemma of the word is enough to tackle this situation. Lemmatization is reducing the word to its purest form without any morphology applied to it. This simplest or purest form of the word is called Lemma of the word and the process is called Lemmatization. This process reduces the word into it’s most essential form.

Word With Morphology: [doing , running, forming]

Word After Lemmatization: [do , run , form]

4) *Pluralization:* This is a simple but an essential form of morphology of a word. At the first instance one might think the process is easy, ”dog-dogs”, better,”goddess-goddesses”. These kind of simple plural forms of words can be easily devised by an algorithm that a computer can process. The issue starts with complex Pluralization words. ”child” is ”children”, but the computer may address it is as childs or childses, which is eventually incorrect. And hence, pluralization is also an issue when it comes to Natural language processing, This issue can be solved by a numerous methods, by datasets, by online look-up, by hash tables etc. But efficient conversion is still in question in Natural Language Processing.

TextBlob is the library that will be used in this paper as it has been tested to do Pluralization at a good efficiency.

IV. METHODOLOGY

We use the Naive Bayes algorithm to classify the classes. This algorithm does not work well on big data. It works well on relatively small datasets. Our goal is to protect NOSQL database insertion operation by detecting spam. Ranging from bio, comments, reviews. We will use MongoDB in our imple-mentation as it is the most flexible and supports JSON format which will in turn help our app to evolve as an endpoint to classify texts to help the general public.

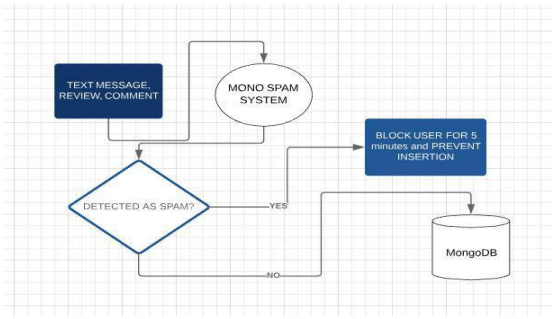

Fig. 7. High Level View Of the system

1) *MONGODB INSERTION:* In our paper we use Mongodb, and hence it is important to mention the methods involved while validating texts. Once a text is ready to insert into the database, our model acts as a firewall and validates it. If the text is classified as spam the user and the text is immediately blocked from inserting into the database. In mongoDb, there are collections in a database. Each collection has a record. It is always empty unless a data is inserted. In fact, the collection or DB is not created unless it has at least one data. And hence, by default our MONO spam would block any spam related texts.

A. Corpus – DATASET

The dataset is a 4000 element SMS text message data set taken from UCI Machine Learning Repository. It is a TSV file, where the message and the label is separated by tabs. The label is ”spam” for spam message and ”ham” for not spam message. The corpus is separated at spaces and the csv is read as Pythons panda object. It conists of 425 spam messages and 3,375 clean messages.

B. Pre Processing and Analytic

Before we process the data into feature vectors it is very important to remove features that do not contribute to the class or negligible contribution. This can be done by correlation coefficient or total Correlation or partial correlation. If Y is dependent on X keep X, else remove X. Sometimes length of the data can contribute to the final class label. We will discuss about the implementation of our paper in the implementation section.

1) BoW transform and TF-IDF transform: From the given dataset make a bag of words(BoW) by forming a Vocabulary. The vocabulary contains tokens which is called grams. One gram vocabulary contains single tokens and this can lead to the formation of a sparse matrix which is a very huge computation problem. Before, giving the solution for this, we need to address the definition of tokens and one gram vocobulary. Tokens or grams are the different words from the dataset without duplicates. One gram vocabulary is defined as the vocabulary whose tokens are a single word. An n-gram vocabulary contains tokens with n words.

1-gram Vocabulary = [”the”,”tiger”,”is”,”an”,”animal”]

2-gram Vocabulary = [”the tiger”, ”tiger is”, ”is an”, ”an animal”]

This approach would reduce the size of the matrix but is again not much efficient. and hence we pre process the data by removing a few stopwords like ”the”,”is” etc. This mechanism will be much explained in the implementation section. TF-IDF transform is done on the bag of words vector and the final feature vector is obtained.

2) *The Learning Algorithm - Naive Bayes Classifier:* As already discussed in the paper, Naive Bayes Classifier returns the probability of an event A given an evidence of event B. The classifier makes three assumptions:

- The feature vector is independent
- The response vector is dependent on all feature vectors
- All features have an equal contribution to the response vector

This assumption is very impractical in real world scenario but it fortunately works and has been proved right in many cases. The feature vectors that we obtained is from the tf-idf transform.

C. Algorithm and Flowchart

1) **Algorithm:** The MONO spam algorithm to process data and filter spam for NOSQL database insertion.

Input: The raw corpus
Output: A boolean value for NOSQL insertion

PROCESSING INVOLVED:

- The corpus is stripped on the tab spaces and converted to an array of Panda objects
- The vector is analyzed and length of each message is appended with the features of row i.e. Length is added as a feature
- Two functions are defined to tokenize(sentence to words) and lemmatize(words to it’s base form) a sentence from each feature vector.
- The list of words is vectorized using term frequency transformation where a vocabulary is defined from the data set and it contains no duplicates. For each sentence in the dataset, the frequency of each word occuring in the document is assigned. This leads to Sparse matrix problem. (which can be solved by combining words)
- Inverse Document Frequency transform is applied on the tf transformed set of feature vectors.
- The final feature vector along with the message length is obtained as shown in Table 2.
- The Multinomial Naive Bayes classifier is used to cate-gorize these vectors into their classes [Spam, not spam] The model is used on various messages that are requested to be inserted into the database
- If prediction of message is spam, return true, else false.

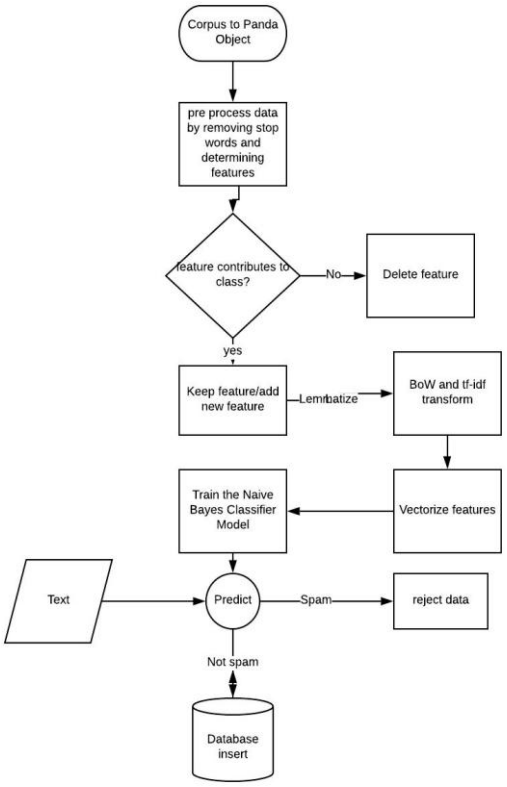

Fig. 8. Detailed Description of Processing

V. IMPLEMENTATION

Python is the language used for its code readability and simplicity and has a vast number of third party libraries to simple up the processing involved. In this paper, the system is implemented with dependencies. The dependencies are listed below.

A. TextBlob

TextBlob is a lightweight Natural Language Processing library which helps to split sentence into words, tag words, get the Part of Speech, Analyze the phrase tree, Lemmatize the words to its base form, Pluralize the words and pre process the data to exactly fit the model. It is used to tokenize the words and also remove stop words thereby removing the sparse matrix issue and greatly decreases computation time. TextBlob has one con that it is specific to the English language and cannot parse the sentences of other language. Since our paper is localized to the English language, we have chosen to use TextBlob.

B. Data Visualization and pre processing

Using matplotlib we visualize the dataset and get an his-togram of the datas on X axis to their lengths on Y axis.(Fig 9)

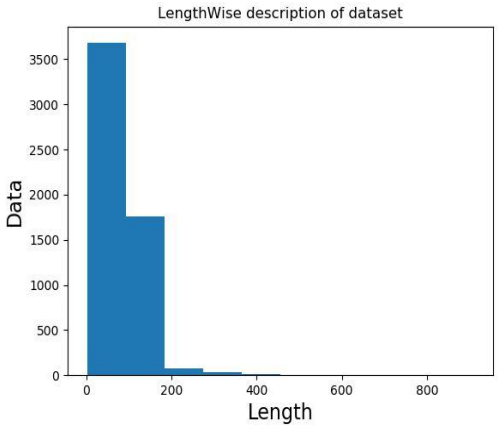

Fig. 9. Each message plotted with it's length :: Length -X axis ; Data - Y axis

There can be no information obtained from this but the data is skewed towards the right, there are many datas with more length. Texts with more length can be spam or ham? We need to visualize it. Data splitting is done where datas belonging class "spam" and datas belonging to class "ham" is separatedand we append a length feature to each of the newly obtained two vectors. Matplotlib is again used to visualize these features and check if length of the text contributes to the result. It is noted that the length indeed contributes to the outcome of spam or not spam. So length of the message is appended as a feature.(Fig 10) Each feature vector is a sentence and this sentence is split into words. The words are then lemmatized to their base forms and then the stop words are removed from the vocabulary. For each of the feature vector Term Frequency is obtained. There is an issue with term frequency. It provides the same weight to all words. Hence we extend the Tf transform to tf-idf transform which provides more weight to words that make a difference in meaning.

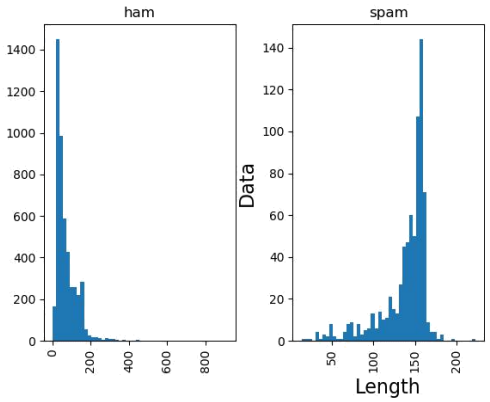

Fig. 10. Dataset is divided into spam and not spam subsets and the length is plotted, FIG 11a : Spam , FIG 11b : Ham ; Length -X axis ; Data - Y axis

Term Frequency-Inverse Document Frequency is represented as

TF-IDF is calculated from TF and IDF based on the

$$\mathrm{tf}(t,d) = 0.5 + 0.5 \cdot \frac{f_{t,d}}{\max\{f_{t',d} : t' \in d\}}$$

Fig. 11. Term Frequency

$$\mathrm{idf}(t,D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

Fig. 12. Inverse Document Frequency

$$\mathrm{tfidf}(t,d,D) = \mathrm{tf}(t,d) \cdot \mathrm{idf}(t,D)$$

Fig. 13. Term Frequency - Inverse Document Frequency Transform

C. Training the model

The output of the tf-df transformer will be the input as a feature vector to the classifier. The classifier will take the corresponding vocabulary tokens tfidf values and map it to the class labels. This feature vector is the training set. The spam and ham vectors will be the class labels. This is passed as the input to the Multinomial Naive Bayes classifier. The classifier predicts the output class labels and returns the string ”spam” or ”notSpam”. This metadata is used once a user sends a request to insert their message into the NOSQL database. The message is parsed and the algorithm is applied, if the output of the classifier is spam, the message is not inserted, else the message is stored.

Documentn Frequency	W1	W2	W3	..	..	Length	Classn Label
A	1	1	4			431	spam
B	2	1	3			312	spam
C	3	2	2			100	not spam

TABLE II: A SAMPLE REPRESENTATION OF THE TRANSFORTMATIONS

VI. RESULTS AND CONCLUSION

The model is exported and ready to test, We take a part of the dataset as a test data and apply the BoW, tf-idf transformations and test the result. If the test is passed, the data is inserted into mongodb else not inserted. We obtain the accuracy and confusion matrix to precisely calculate the efficiency of the product.

A. Accuracy

The model yielded an accuracy of 96.95 percent. Out of 100 insertions into the NOSQL database, our model would accurately reject or insert 97 percent of the times correctly. That is, true positives. Accuracy alone will not help us to decide the reliability of the model. Hence we make computations based on the confusion matrix.

B. Confusion Matrix and related calculations

We use metrics like sensitivity, specificity, precision, Neg-ative Predicted Val, FP rate, FN rate, FD rate, Correlation Coefficient.

Inference From Matrix : The model predicted real 577 spam. predicted 0 real not spam as spam, 170 real spam as not spam, 4827 real not spam as not spam.

	True Positive	True Negative
Predicted Positive	577	0
Predicted Negative	170	4827

Fig. 14. Confusion Matrix

C. Metrics Explanation

From fig 15, we can infer that,

- Sensitivity : *77 percent*, our model predicts 77 percent accurately that a message is spam, given all messages are spam.
- Specificity : *100 percent*, our model predicts 100 percent accurately that a message is not spam if it is really not spam.
- Precision: *100 percent*, our model is very reserved to the spam messages and hence it has a precision of 100 percent
- NPV *: 96.6 percent* , our model predicts not spam with 96 percent rate
- Correlation: *0.86* There is a strong relation between our class labels and feature vectors. In future we can try to skew this to 1.

Measure	Value	Derivations
Sensitivity	0.7724	TPR = TP / (TP + FN)
Specificity	1.0000	SPC = TN / (FP + TN)
Precision	1.0000	PPV = TP / (TP + FP)
Negative Predictive Value	0.9660	NPV = TN / (TN + FN)
False Positive Rate	0.0000	FPR = FP / (FP + TN)
False Discovery Rate	0.0000	FDR = FP / (FP + TP)
False Negative Rate	0.2276	FNR = FN / (FN + TP)
Accuracy	0.9695	ACC = (TP + TN) / (P + N)
F1 Score	0.8716	F1 = 2TP / (2TP + FP + FN)
Matthews Correlation Coefficient	0.8638	TP*TN - FP*FN / sqrt((TP+FP)*(TP+FN)*(TN+FP)*(TN+FN))

Fig. 15. Calculation of Metrics

- False Negative: Predicted ”Not spam”, Actual ”Spam”
- True Positive: Predicted ”spam”, Actual ”Spam”
- True Negative: Predicted ”Not spam”, Actual ”Not Spam”
- False Positive: Predicted ”spam”, Actual ”Not Spam”

VII. FUTURE WORKS

The model is already made for scalable purpose as it is used on MongoDB which represents data in the format of JSON. JSON is evolving as an API standard data fetch format. But since Mongodb is not considered to store big data. The idea can be used to develop a model for the big data domain using Deep learning algorithms and Neural Networks. Machine learning works well on small datasets, deep learning requires big data sets to provide a better accuracy, But the trade off is computation time. Deep learning algorithms require powerful supercomputers as it should be run on clusters to generate the model. Simple big data processing can be made on Apache Spark which uses Resilient Distributed Datasets as it distributes the big data onto the clusters in the datacenters. Similiar to the MapReduce approach invented by google, In deep learning, the reduce operation could be the linear combination operations to find out combination constants to predict the output class labels. This can be a future enhancement.

REFERENCES

1. White Paper on Natural Language Processing by Jaime Carbonell et al
2. Recent Trends in Deep Learning Based Natural Language Processing by Tom Young et al, Nov 2018
3. Spam Detection with Logistic Regression by Natasha Sharma, May 2018
4. Spam Detection techniques: A review by Gurjot Kaur, 2013.
5. Filtering Spam with Behavioral Blacklisting by Nick Feaster, 2017
6. Survey of Collaborative Filtering Algorithms for Social Recommender Systems, 2016.
7. A review on Evaluation Metrics For Classification Evaluations BY HOSSIN. M ET AL, 2015.
8. The War Against Spam: A report from the front line by Bradley Taylor et al, Google

AUTHORS PROFILE

Eshwar.S, is currenty pursuing Btech. CSE(3rd Year) in VIT University, Vellore. Email id: eshwar.s2016@vitstudent.ac.in . His current research interests includes Data science, Machine learning, Web development and Virtualization.

Dr. K.Lavanya is currently working as an Associate Professor in the School of Computer Science and Engineering(SCOPE) in VIT, Vellore. She received her Ph.D. degree in Computer Science and Engineering from VIT University, Vellore, on August 2015 [July 2011 - August 2015]. She completed ME in Computer Science and Engineering from VIT University, Vellore, in the year 2011. She also received BE degree in Computer Science and Engineering from the Anna University, India, in 2005. Her current research interests include *Computational Intelligence, Data Science, NoSQL databases, Data Mining and Warehousing, Machine Learning*.