A Novel Method for Detecting Spam Email using KNN Classification with Spearman Correlation as Distance Measure

Ajay Sharma
M.Tech Scholar
Samrat Ashok Technological Institute
Vidisha,M.P.

Anil Suryawanshi
Assistant Professor
Samrat Ashok Technological Institute
Vidisha,M.P.

ABSTRACT
E-mail is the most prevalent methods for correspondence because of its availability, quick message exchange and low sending cost. Spam mail appears as a serious issue influencing this application today's internet. Spam may contain suspicious URL's, or may ask for financial information as money exchange information or credit card details.

Here comes the scope of filtering spam from legitimate e-mails. Classification is a way to get rid of those spam messages. Various researches are proposed for spam filtering by classifying them into labels of spam and business messages.

Bayesian classification based spam filtering technique is a popular method. Also SVM based classifications are also used. K-nearest neighbour classification is simple, straightforward and easy to implement and has high F-measure compare to Bayesian and SVM classification. But accuracy of traditional KNN is lower than Bayesian classification.

In this work a detection of spam mail is proposed by using K-nearest neighbour classification method by combining Spearman's correlation coefficient as distance measure rather than traditional Euclidean distance. Experimental results present a significant improvement in accuracy with higher F-measure compare to traditional algorithms.

Keywords
Bayesian classification, SVM Classification, spam, Email, KNN classification, Spearman correlation, Spam Filtering, Accuracy, F-measure.

1. INTRODUCTION
Electronic mail, most usually called email or email subsequent to around 1993 is a strategy for trading computerized messages from a creator to one or more beneficiaries. Email works over the Internet or other Ecosystem Email is electronic device .it is method of exchange message from source to destination. Email is very fast furthermore, dialect utilized as a part of messages is basic can be formal or informal. There is no paper work while using email. Some early email systems required the maker and the recipient to both is online meanwhile, in a similar way as messaging. Today's email systems rely on upon a store-and-forward model. Email servers recognize, forward, pass on, and store messages. Neither the customers nor their PCs are required to be online all the while; they require associate just quickly, regularly to a mail server, for whatever time span that it takes to send or get messages. Genuinely, the term electronic mail was used nonexclusively for any electronic record transmission. Case in point, a couple creators in the mid 1970s used the term with the more particular significance it has today. An Internet email message comprises of three segments, the message envelope, the message header, and the message body. The message header contains control data, including, negligibly, an originator's email address and one or more beneficiary locations. Generally enlightening data is likewise included, for example, a subject header field and a message settlement date/time stamp. At first and ASCII content just correspondence medium, Internet email was extended by Multipurpose Internet Mail Extensions (MIME) to pass on substance in other character sets and multi-media content associations. All inclusive email, with internationalized email addresses using UTF-8 have been regulated, yet not generally received [1, 2].

1.1 Privacy Concerns
Today it can be essential to recognize Internet and interior email frameworks. Web email might travel and be put away on systems and PCs without the sender's or the beneficiary's control. Amid the travel time it is conceivable that outsiders read or even change the substance. Inside mail frameworks, in which the data never leaves the hierarchical system, might be more secure, in spite of the fact that data innovation faculty and others whose capacity might include observing or overseeing might be getting to the email of different representatives.

Email protection, without some security safeguards, can be traded off on the grounds that:

- Email messages are for the most part not encoded.
- Email messages need to experience middle of the road PCs before coming to their destination, which means it is generally simple for others to block what's more, perused messages.
- Numerous Internet Service Providers (ISP) store duplicates of email messages on their mail servers time as of late they are passed on. The fortifications of these can stay for up to a while on their server, in dislike cancellation from the letter drop.
- The "Got:"- fields and other data in the email can regularly recognize the sender, averting unknown cores. [5, 6, 7]

2. LITERATURE SURVEY
Ommera Jan, Heena Khana, the filtered mails are further filtered to measure the misclassification using different data mining techniques. The results show that the decision tree is the best classifier. It is easy to interpret and explain the executives. In comparison to random forests are time efficient. Decision tree requires relatively less effort from users for data preparation [8].

Tarjini Vyas ,Payal Prajapati consider diverse arrangement systems utilizing WEKA to channel spam sends. Result demonstrates that Naive Bayes method gives great precision (close to most astounding) and set aside minimum time among different strategies. Likewise a similar investigation of every method as far as exactness and time taken is given. It can be concluded that from all techniques that have been used here, Naive Bayes technique gives faster result and good accuracy over other techniques (except SVM and ID3). SVM and ID3 give better accuracy than naïve Bayes but take more time to build a model. There is a trade-off between time and accuracy. So which technique is used depends on the application at hand. [9].

In tending to the developing issue of garbage E-mail on the Internet, Mehran Sahani, Susan Dumais inspect techniques for the robotized development of channels to dispense with such undesirable messages from a client's mail stream. By throwing this issue in a choice theoretic system, there is a plausibility to make utilization of probabilistic learning routines in conjunction with a thought of differential misclassification expense to create channels which are particularly fitting for the subtleties of this assignment. While this might show up, at to start with, to be a straight-forward content grouping issue, it demonstrates that by considering space particular components of spam separating notwithstanding the crude content of E-mail messages, a great deal more exact channels can be produced.[10]

Wenjuan Li ,Weizhi Meng identify that larger studies should be conducted to explore the practical performance of SML in different environments. In this work, an empirical study is performed with three different environments and over 1,000 participants regarding this issue. It is found that decision tree and SVMs are acceptable by most users in real environments and that environmental factors would greatly affect the performance of SML classifiers. [11]

Jitendra Nath Shrivastava, Maringanti Hima Bindu a Genetic Algorithm based email spam arrangement calculation is proposed. In this work some essential results are introduced. This calculation effectively recognizes spam and ham messages. The proficiency of the procedure relies on upon the dataset and GA parameters. The productivity of the calculation is more than 82%. [12]

The implemented work results in the improvement of the accuracy and time of the classification process and hence, the work of spam detection can be done easily but the features which are identified here are just related to the spam data like no. of URL in the tweet or number of spam words etc [13].

Eman M. Bahgat , Sherine Rady, Walaa Gad,an email filtering approach using classification techniques is proposed and studied. Two ways of selecting features are suggested. In the first, features are extracted from body content based on web document analysis methods. In the second way, dimensionality of these extracted features is reduced by selecting the determined (meaningful) terms only using a constructed dictionary. Experimental studies have been conducted using several classifiers and compared to existing related work using the same dataset. The recorded results prove the efficiency of the proposal filtering approach. The dictionary based filtering had an acceptable performance with faster filtering execution. [14]

Tao Ban ; Shimamura, J. at el. propose another online framework that can rapidly recognize vindictive spam messages and adjust to the adjustments in the email substance and the Uniform Resource Locator (URL) joins prompting malevolent sites by redesigning the framework day by day. To break down email substance, we embrace the Bag of Words (BoW) approach and create highlight vectors whose traits are changed taking into account the standardized term recurrence opposite report recurrence (TF-IDF). The outcomes affirm that the proposed spam email discovery framework has capacity of identifying with high recognition rate. [15]

3. RELATED WORK

3.1 Bayes Theorem

Thomas Bayes, it is known as Bayes' theorem, a nonconformist English clergyman who has started work in the field of probability and the theory of decision amid the eighteenth century. Suppose X is a data tuple. In terms of Bayesian, X is taken as "evidence." of course, it is portrayed by estimations made on an arrangement of n attributes. Suppose H be a theory, for example, that the data tuple X fits in with a predetermined class C. For classification issues, we need to focus P (H|X), the probability that the theory H holds given the "evidence" or watched data tuple X. At the end of the day, we are searching for the probability that tuple X has a place with class C, given that we recognize the attribute depiction of the X.

P (H|X) is the back likelihood, or a posteriori likelihood, of H adapted on X. For instance, assume our universe of data tuples is bound to clients depicted by the attributes, age and wage, individually, and that X is a 35-year-old client with a wage of $40,000. Assume that H is the speculation that our client will purchase a personal computer. At that point P (H|X) reflects the probability that client X will purchase a personal computer given that we know the client's age and pay.

Conversely, the prior probability is P (H), or from the prior probability, of H. For our sample, this is the probability that any given client will purchase a personal computer, paying little respect to age, wage, or some other data, so far as that is concerned. , P (H|X) is the posterior probability, depends on more data (e.g., client data) than the P (H) is prior probability, which is not dependent on X.

Essentially the posterior probability of X is (X|H) molded on H. That is, it is the probability that clients, X, are 35 years of age and procures $40,000, given that we know the client will purchase a personal computer the prior probability of X is P (X). Utilizing our sample, the probability a man from our set of clients is 35 years of age and wins $40,000. P (H), P (X|H), and P (X) may be evaluated with the help of the information that is given. Bayes' theorem is helpful in that it gives a method for estimating the posterior probability (H|X), from, P (H), P (X|H) and P(X). Bayes' theorem is

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)}$$

The working of the naïve Bayesian classifier, or basic Bayesian classifier, is as follows:

1. Suppose an arrangement of preparing of tuples is D and their related class labels. Obviously, every tuple is represented through a n-dimensional attribute vector, $X = (x_1, x_2, \ldots\ldots x_n)$ delineating n estimations prepared on the tuple from n attributes, correspondingly $A_1, A_{2.} \ldots\ldots A_n$.
2. Assume that there are m classes. Given a tuple, X, the classifier will anticipate that X fits in with the class having the most elevated posterior

probability, molded on X. That is, the naive Bayesian classifier predicts that tuple X has a place with the class C_i if and only if

$$P(C_i \mid X) > P(C_j \mid X) \quad for I \le J \le m, j \ne i, \tag{2}$$

Bayesian classifiers are likewise helpful in that they give a hypothetical support to different classifiers that don't expressly utilize Bayes' theorem. For instance, under specific suspicions, it can be demonstrated that numerous neural system and bend fitting calculations yield the most extreme posteriori hypothesis; the naive Bayesian classifier does the same.

3.2 Support Vector Machine (SVM)

SVM is a set of supervised learning technique with, associated with learning algorithms that is utilized for classification, clustering and regression. Given a set of training examples, each marked for belonging to one of two categories ,an support vector machine training algorithm builds a model that assign new example into one category on the other , making it an on-probabilistic binary linear classifier. Support vector machine model is representations of the separate classifications are partitioned by clear hole that is as wide as possible. That same space and anticipated to fit in with class taking into account which side of the crevice they fall on.

It has been shown by several researchers that SVM is also an accurate algorithm for classification. It is also widely utilizing in Websites page classification and bio-informatics applications.

SVM has been functioning with achievement to the information retrieval problem. SVM is a machine learning technique which is based on vector space where the purpose is to establish a decisive edge between two classes which is maximally for a form a few positions in the training data.

$$D = \{(X_i, y_i) | X_i \in R^P, y_i \in \{-1, 1\}\}_{i=1}^{n}$$

Where the value of Y_i belonging between1and−1,representing the class to which the point xi belongs. Here every xi is a p-dimensional actual vector and we discover the high-margin hyper plane that divides the points having Y_i =1 from those having Y_i =−1.

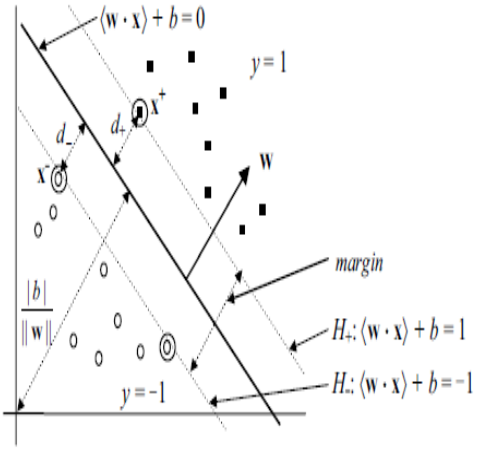


Figure 1: Basic Architecture of SVM

Figure 4.2 shows the basic architecture of SVM. Highest margin hyper plane and margins for an SVM with samples from two classes. Samples on the margin are known as support vectors.

Any hyper plane can be described as the set of points x fulfilling. A separating hyper plane is described by the regular vector w and the offset b:

$$w.x + b = 0$$

Where • denotes the dot product. W is also known as the regular vector of the hyper plane. Exclusive of change the regular vector w, unstable b moves the hyper plane parallel to itself. While SVM maximizes the margin between positive and negative data points, let us discover the margin. Let d_+ (correspondingly d_-) be regular the shortest distance from the extrication hyper plane (<w. x>+ b = 0) to the closest positively (negative) data position. The margin of the extrication hyper plane is $d_+ + d_-$.

Let us consider a positive data point (x+, 1) and a negative (x-, -1) which is very close to the hyper plane <w. x> + b = 0. We describe two parallel hyper planes (H+ and H-) that pass by x+ and x- correspondingly. H+ and H- are also parallel to <w. x> + b = 0. We can rescale w and b to achieve

$$H_+ : < w.x^+ > + b = 1$$

$$H_- : < w.x^- > + b = -1$$

The space between the two margin hyper planes H_+ and H_- is $(d_+ + d_-)$. Distance from a point x_i to a hyper plane <w. x> + b = 0 is:

$$\frac{|<w.x_i>+b|}{||w||}$$

Therefore, the decision edge <w. x> + b = 0 lies, half way between H_+ and H_-. The margin is Therefore

$$margin = d_+ + d_- = \frac{2}{||w||}$$

Consider the training sample $\{(x_i, d_i)\}$, where x_i is the input sample, d_i is the preferred output

$$W_0^T X_i + b_0 \ge +1, for\ d_i = +1$$

$$W_0^T X_i + b_0 \le -1, for\ d_i = -1$$

3.3 K-Nearest-Neighbor Classifiers

Do not include headers, footers or page numbers in your The k-nearest-neighbor technique was initially portrayed, in the mid 1950s. The technique is working seriously when given extensive training sets, and did not pick up popularity until the 1960s when expanded calculating power got to be accessible. It has subsequent to being generally utilized as a part of the pattern recognition.

Closest neighbor classifiers depend on learning by relationship, that is, by contrasting a given test tuple and preparing tuples that are like it. The preparation tuples are depicted by n traits. Each tuple speaks to a point in a n-dimensional space. Thusly, all the preparation tuples are put away in a n-dimensional example space. At the point when given an obscure tuple, a k-closest neighbor classifier looks the example space for the k preparing tuples that are nearest to the obscure tuple. These k preparing tuples are the k "closest neighbors" of the obscure tuple.

"Closeness" is characterized as far as a separation metric, for example, Euclidean separation. The Euclidean separation between two focuses or tuples, say, $X_1 = (x_{11}, x_{12}.... x_{1n})$ and $X_2 = (x_{21}, x_{22}.... x_{2n})$ is

$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2}$$

In another way, for every numeric attribute, we take the distinction between the relating estimations of that attribute in a tuple and in tuple, square this distinction, and aggregate it. The square root is taken of the aggregate amassed distance count. Regularly, we standardize the estimations of every quality before utilizing Eq. (9.22). Helps prevent attributes with at first expansive reaches (e.g., salary) from exceeding attributes with at first smaller extents (e.g., binary attributes). Min-max standardization, for instance, can be utilized to change a value v of a numeric attribute A to v' in the extent [0, 1] by calculating

$$v' = \frac{v - \min_A}{\max_A - \min_A}$$

Where $\min_A$ and $\max_A$ are the least amount and greatest values of attribute *A*. Chapter 3 defines different techniques for data standardization as a form of data transformation.

For k-nearest-neighbor classification, the obscure tuple is appointed the most commonly identified class between its k nearest neighbors. To the position when k = 1, the obscure tuple is appointed the class of the training tuple that is nearest to it in pattern space. Nearest neighbor classifiers can likewise be utilized for numeric prediction, that is, to give back an actual estimated prediction for a specified obscure tuple. For this condition, the classifier precedes the normal estimated value of the actual valued labels connected through the k-nearest neighbors of the tuple that is not known.

The past discussion supposes that the attributes utilized to define the tuples which are numeric. For nominal attributes, a straightforward strategy is to analyze the relating estimation of the attribute in tuple with that in the tuple. On the off chance that the two are indistinguishable (e.g., tuples X_1 and X_2 both have the shading blue), then the distinction among the two is taken as 0. On the off chance that the two are distinctive (e.g., the tuple X_1 is blue yet tuple X_2 is red), then the distinction is thought to be 1. Different techniques may include more modern plans for differential reviewing (e.g., where a bigger distinction score is allocated, state, for blue and whiter than for blue and black).

By and large, if the estimation of a given attribute A is lost in a tuple X_1 and/or in tuple X_2, we expect the most extreme conceivable distinction. Assume that each of the attributes has been mapped to the extent [0, 1]. For ostensible attributes, we take the distinction quality to be 1 if either one or together of the relating estimations are absent. On the off chance that A is numeric and omitted from tuples X_1 and X_2, then the distinction is additionally taken to be 1. On the off chance that a single estimate is omitted and the (that we will call) is accessible and consistent, then we can acquire the dissimilarity to be either $|1 - v'|$ or $|0 - v'|$ then again (i.e., $1 - v'$ or v'), whichever is bigger.

This can be resolved tentatively. Beginning with K=1, we utilize a test set to evaluate the error rate of the classifier. This procedure can be replicated every time by augmenting k to take into account one more neighbor. The k estimate that specifies the minimum error rate might be chosen. When all is said in done, the bigger the quantity of training tuples, the bigger the estimation of k will be (so that classification and numeric prediction choices can be found on a bigger bit of the put away tuples). As the quantity of training tuples approaches infinity and K=1, the error rate can be no more awful than double the Bayes error rate (the recent being the hypothetical least).

Nearest neighbor classifiers utilize distance-based comparisons that characteristically allocate equivalent weight to every attribute. The strategy, be that as it may, has been altered to join attribute weighting and the pruning of uproarious data tuples. The decision of a distance metric can be basic. The Manhattan (city square) separation), or other distance estimations, might likewise be utilized.

Nearest neighbor classifiers can be to a great degree moderate when classifying test tuples. On the off chance that D is a training database of |D| tuples and K=1, a then O (|D|) correlation is required to classify a specified test tuple. By presorting and organizing the put away tuples into search trees, the quantity of correlations can be diminished to O (log |D|). Parallel usage can diminish the running time to a steady, that is, O (t), which is free of |D|.

Different strategies to accelerate classification time incorporate the utilization of distance calculations and altering the put away tuples. In the partial distance strategy, we process the distance based on the view of a subset of the n attributes. The altering technique uproots training tuples that demonstrate futile. This technique is additionally alluded to as pruning or gathering on the grounds that it diminishes the aggregate number of tuples put away.

4. PROPOSED METHODOLOGY

4.1 Spearman's Correlation Coefficient

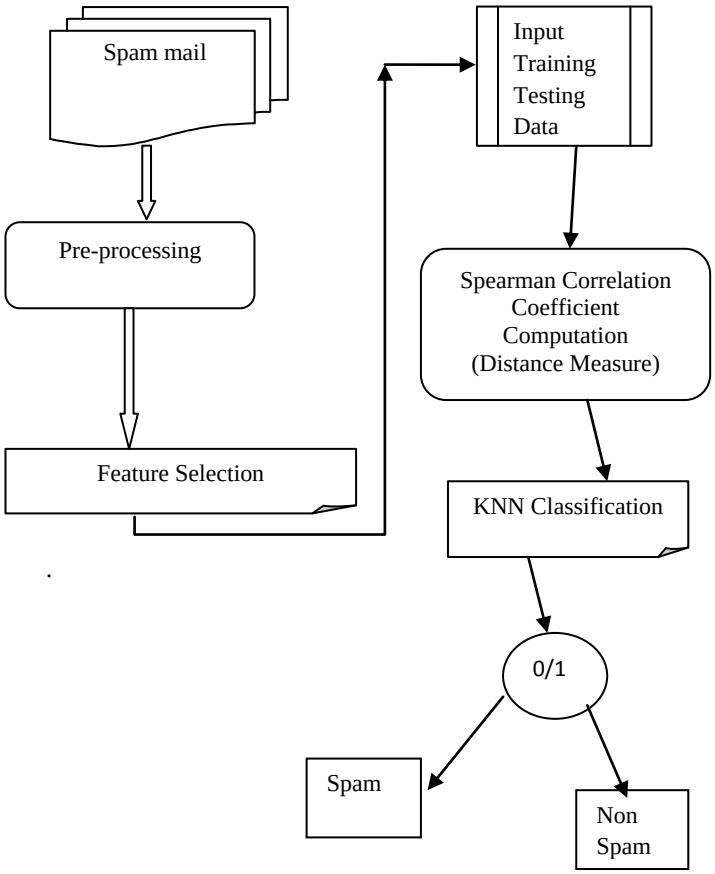
Spearman's connection coefficient is a factual measure of the quality of a monotonic relationship between matched information. Spearman's correlation coefficient is a measure of a monotonic relationship and thus a value of do not imply there is no relationship between the variables. For example in the following scatter plot which implies (monotonic) correlation however there is a perfect quadratic relationship.

Before learning about Spearman's correlation it is important to understand Pearson's Correlation which is a factual measure of the quality of a straight relationship between matched information. Its computation and ensuing essentialness testing of it requires the accompanying information presumptions to hold:

- interval or ratio level;
- linearly related;
- Vicariate typically circulated.

In the event that information does not meet the above presumptions then utilize Spearman's rank connection

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}.$$



.

Figure 2: Proposed Methodology

4.2 Methodology

KNN algorithm with Spearman Correlation

a. Initialize input from data set: Test Tuple

b. Compute spearman correlation coefficient between test tuple with training tuple.

If X and Y are training and testing tuple respectively then Spearman's correlation can be computed as-

$$d_{ij} = 1 - \frac{6 \sum_{i=1}^{n} (rank(x_i) - rank(y_j))^2}{n(n^2 - 1)}$$

c. Compute neighbor set to the tuple X where number of element in neighbor set is k.

Here k=3;

d. Determining the majority class by finding the closest neighbor to the test tuple X.

e. Test tuple is assigned the class of nearest neighbor.

f. Output : Class Label for Test tuple (0 or 1)

5. EXPERIMENTAL SET UP & RESULTS

The arranged work is actualized in MATLAB r2010b bundle. All investigation and diagrams are planning on utilizing MATLAB. MATLAB gives apparatuses to collect, dissect, and envision data, endorsing you to acknowledge understanding into your data in a small amount of the time it would take utilizing spreadsheets or customary programming dialects. Moreover, record and comes about sharing through plots and reports or as uncovered MATLAB code is likewise conceivable in MATLAB.

5.1 Dataset

Spambase dataset [17] is used to simulate the proposed work.

Dataset is available at UCI machine learning repository. The data set has 4601 instances in which 1813(39.4%) are spam. Each tuple has 58 attributes in which 57 constants define features in Email and one is ostensible class mark. The email with class mark 1 is known as spam and 0 as non spam. Here are the meanings of the characteristics:

48 ceaseless genuine [0,100] qualities of sort word_freq_WORD = rate of words in the email that match WORD,

All out number of words in email. A "word" for this situation is any string of alphanumeric characters limited by non-alphanumeric characters or end-of-string. 6 ceaseless genuine [0,100] qualities of sort char_freq_CHAR

= rate of characters in the email that match CHAR,

1 consistent genuine [1,...] characteristic of sort capital_run_length_average

= normal length of continuous groupings of capital letters

1 consistent whole number [1,...] quality of sort capital_run_length_longest

= length of longest continuous grouping of capital letters

1 consistent whole number [1,...] quality of sort capital_run_length_total

= whole of length of continuous groupings of capital letters

= complete number of capital letters in the email

1 ostensible {0,1} class characteristic of sort spam

= indicates whether the email was considered spam (1) or not (0),

i.e. spontaneous business email.

5.2 Results

In first section three traditional algorithm Bayesian classification, SVM classification and KNN algorithm with Euclidean distance measure.

Later KNN algorithm with Euclidean is compared with KNN algorithm with spearman's correlation as distance measure.

Following evaluation parameters are used to evaluate and compare techniques-

The system is evaluated using the F-Measure, Precision, Recall and Accuracy, given as follows:-

$$F - measure = \frac{2 \times P \times R}{P + R}$$

Where *P and R* are defined as:

$$P(precision) = \frac{TP}{TP + FP}$$

$$R(recall) = \frac{TP}{TP + FN}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

5.2.1 Comparative Study of Bayesian, SVM and KNN Classification

A. Precision

Table 1 Comparison of Classifier (Precision)

Classification Technique	Train-Test			
	50-50	*60-40*	*70-30*	*80-20*
Bayesian	0.1997	0.1463	0.6561	0.4324
SVM	1	1	1	1
KNN	0.9290	0.9122	0.8571	0.9103

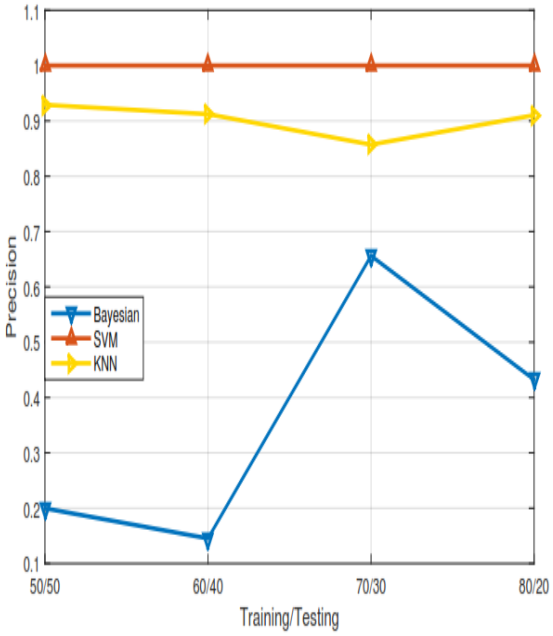


Figure 3: Comparison of Classifier (Precision)

B. Recall

Table 2: Comparison of Classifier (Recall)

Classification Technique	Train-Test			
	50-50	*60-40*	*70-30*	*80-20*
Bayesian	0.6624	0.7500	0.2404	0.2388
SVM	0.3504	0.2729	0.2195	0.1773
KNN	0.3771	0.3278	0.3193	0.3080

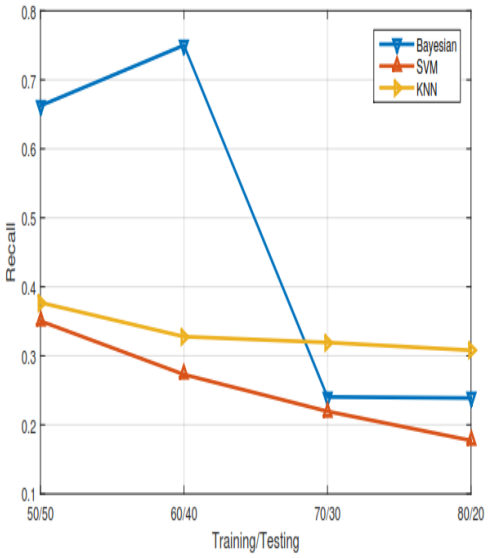


Figure 4: Comparison of Classifier (Recall)

C. F-Measure

Table 3: Comparison of Classifier (F-Measure)

Classification Technique	Train-Test			
	50-50	*60-40*	*70-30*	*80-20*
Bayesian	0.3069	0.2434	0.3518	0.3077
SVM	0.5189	0.4288	0.3600	0.3012
KNN	0.5364	0.4823	0.4653	0.4603

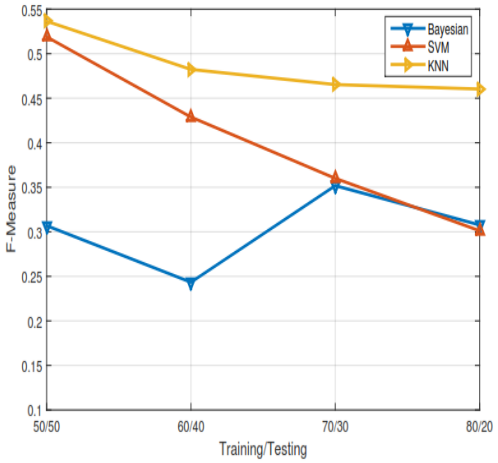


Figure 5: Comparison of Classifier (F-Measure)

D. Accuracy

Table 4: Comparison of Classifier (Accuracy)

Classification Technique	Train-Test			
	50-50	*60-40*	*70-30*	*80-20*
Bayesian	0.6766	0.7347	0.4596	0.6697
SVM	0.3639	0.2749	0.2252	0.2139
KNN	0.4491	0.4671	0.5706	0.6384

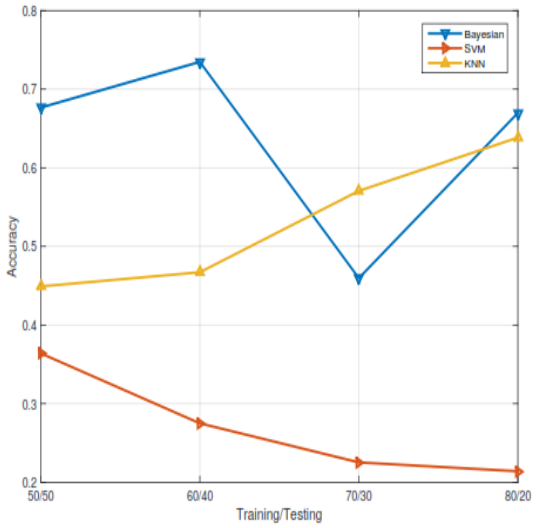


Figure 6: Comparison of Classifier (Accuracy)

5.2.2 Comparative Study of KNN Classification with Euclidean and KNN classification with Spearman Correlation

E. Precision

Table 5: KNNe vs KNNs (Precision)

Classification Technique	Train-Test			
	50-50	*60-40*	*70-30*	*80-20*
KNN with Euclidean	0.9290	0.9122	0.8571	0.9103
KNN with Spearman	0.9772	0.9721	0.9568	0.9744

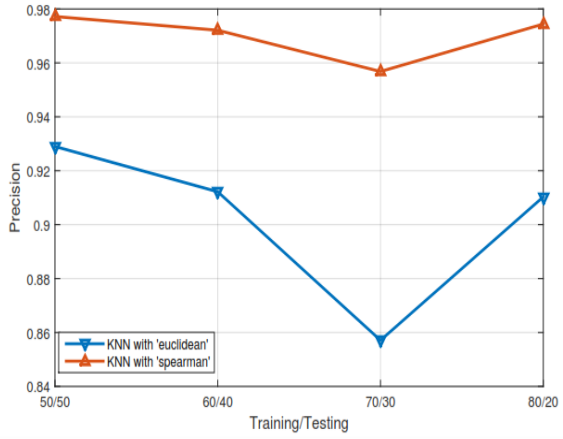


Figure 7: KNNe vs KNNs (Precision)

F. Recall

Table 6: KNNe vs. KNNs (Recall)

Classification Technique	Train-Test			
	50-50	*60-40*	*70-30*	*80-20*
KNN with Euclidean	0.3771	0.3278	0.3193	0.3080
KNN with Spearman	0.9352	0.9276	0.9320	0.8889

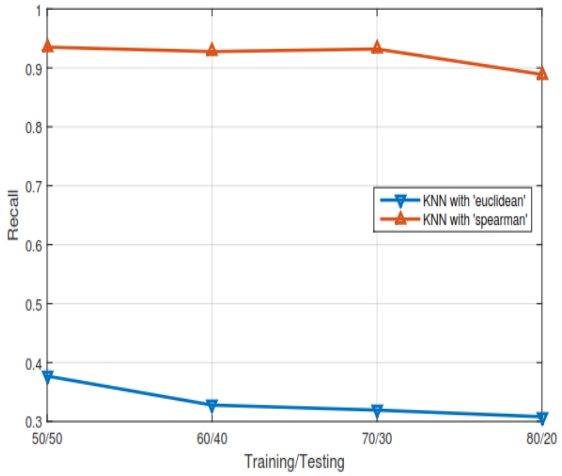


Figure 8: KNNe vs KNNs (Recall)

G. F-Measure

Table 7: KNNe vs KNNs (F-Measure)

Classification Technique	Train-Test			
	50-50	*60-40*	*70-30*	*80-20*
KNN with Euclidean	0.5364	0.4823	0.4653	0.4603
KNN with Spearman	0.9560	0.9493	0.9443	0.9297

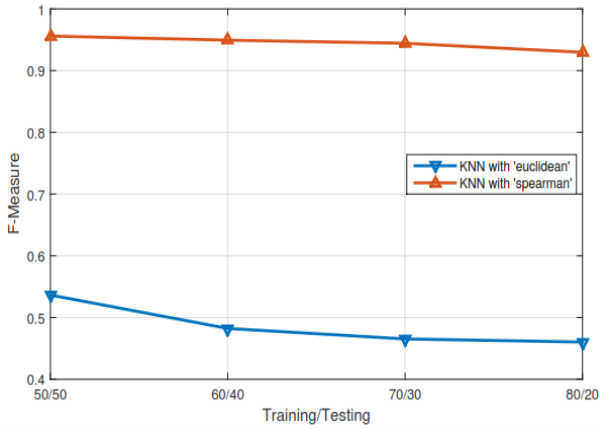


Figure 9: KNNe vs KNNs (F-Measure)

H. Accuracy

Table 8: KNNe vs KNNs (Accuracy)

Classification Technique	Train-Test			
	50-50	*60-40*	*70-30*	*80-20*
KNN with Euclidean	0.4491	0.4671	0.5706	0.6384
KNN with Spearman	0.9691	0.9718	0.9754	0.9750

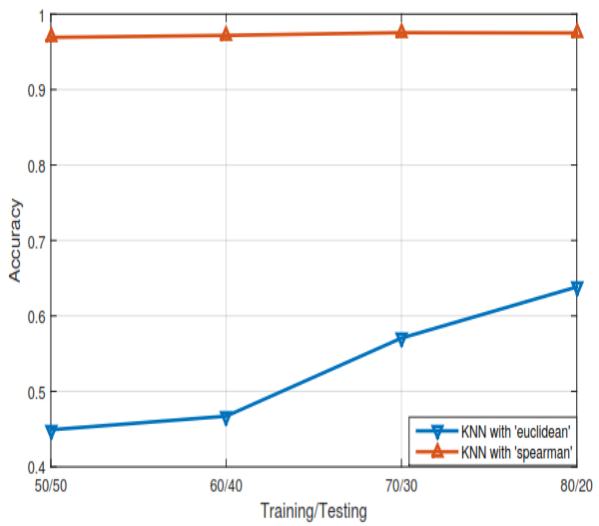


Figure 10: KNNe vs KNNs (Accuracy)

6. CONCLUSION & FUTURE WORK

Email is one of the most common techniques for communication. Spammers use forge mails containing malicious url's, asking for monetary information or personal information which may cause loss in terms of money or leakage of very personal information.

Various techniques are proposed for detecting spam or spam filtering. In many researches Bayesian classification technique is used for spam filtering. SVM classification and KNN classification techniques are also very popular. In this paper above three algorithms are compared.

In next section KNN classification with Spearman's correlation is used for detecting suspicious mail or spams. The proposed algorithm achieves higher accuracy and F-measure compare to above specified techniques.

Spearman correlation coefficient is used as distance measure in KNN classification technique. This can be combined with other filtering technique and may provide better results. Also with large dataset K-nearest neighbor algorithm may face issues in terms of execution time. So further research is required.

7. REFERENCES

[1] Liu, Bing. Web data mining: exploring hyperlinks, contents, and usage data. Springer Science & Business Media, 2007.

[2] M. Tariq Banday. Effectiveness and limitations of E-mail security Protocols. International Journal of Distributed and Parallel Systems (IJDPS) Vol.2, No.3, May 2011

[3] Di Liu. A Spearman correlation coefficient ranking for matching-score fusion on speaker recognition. Browse Conference Publications> TENCON 2010 - 2010 IEEE Regio

[4] MEI paper on Spearman's rank correlation coefficient. December 2007. "Spearman's rank correlation"

[5] Volume 14, Supplement 1, August 2015 "Privacy-preserving email forensics"

[6] T. Pranav Bhat, C. Karthik A Privacy Preserved Data Mining Approach Based on k-Partite Graph Theor Volume 54, 2015

[7] Volume 27, Issue 1, January 2015" Clustering and classification of email contents"a

[8] Ommera jan ,heena khana "An analysis of misclassification error detection in mails using data mining techniques"MAY 2015

[9] Tarjini vyas ,payal prajapati "A survey and evalution of supervised machine learning techniques for spam E-mail filtering" 978-1-4799-608S-9/1S/$31.00©201S IEEE

[10] Mehran sahani ,susan dumais "A Bayesian approach to filtering junk E-mail"

[11] "An empirical study on email classification using supervised machine learning in real environments "EEE ICC 2015 - Communication and Information Systems Security Symposium

[12] "E-mail spam filtering using adaptive genetic algorithm" I.J. Intelligent Systems and Applications, 2014, 02, 54-60

[13] Dr.sanjeev dhawan, jyoti verma "social networking spam detection using R package and k-nearest neighbor classification"www.iasir.net

[14] Emam M.baghat, sherine rady "An email filtering approach using classification techniques"

[15] Tao ban "An online malicious spam mail detection system using resource allocating network with locality sensitive hashing" Received 25 February 2015; accepted 20 April 2015; published 22 April 2015

[16] Kishor, N. Ratna. "International Journal of Advance Research in Computer Science and Management Studies." International Journal 2, no. 3 (2014).

[17] http://www.ics.uci.edu/~mlearn/MLRepository.html (data set)