Improved Methods for Classification of Emails

Snehal Prakash Chaudhari[1], Nikita Balasaheb Dhumal[2], Rupali Ganesh Watamkar[3], Gauri Shantaram Ikade[4]
[1,2,3] *Information Technologies NDMVP College of Engineering, Nashik*

***Abstract*- This system proposed for a hybrid solution of spam email classifier using context based email classification model as main algorithm complimented by information gain calculation to increase spam classification accuracy and Machine Learning algorithms. Previous solution consists of three stages email pre-processing, feature extraction and email classification. We use Naives Bayesian classification and k-mean clustering algorithm to classify the data in the mails. The study has shown that implementing the spam filter in the context –based email classification model is feasible and can be improved.**

Index Terms- Email classification; graph mining algorithm; spam; email classifier

I. INTRODUCTION

Email Filtering, in the context of our application, refers to the classification of an account's emails based on two types of emails (unless keywords specified by the user): 1. Spam and 2. Non-Spam. The user first registers with the application by selecting an available username and setting a password for the account. He then logs in to his account using the registered id and the corresponding valid password. Upon logging in, the user's mails are fetched in the database and are classified into spam and non-spam. The user can also create custom labels which are classified using keywords provided by the user. Also, he can browse for the unread and read emails. This makes the mail service easy and user friendly. A basic task in email filtering is to mine the data from an email and to classify it into the different categories using data mining classification algorithms. Email Filtering involves spam filtering, generalized filtering and segregation and filtering of inbound emails. Spam mails are filtered since they are not important to most of the users. Generalized filtering and segregation of emails is segregation of the mails into different categories as specified by the user using custom labels. Companies filter outbound emails so that sensitive data regarding the working of the company does not leak intentionally or accidentally by emails. To summarize email filtering: 1) Segregates inbound mails into different categories. 2) Filters outbound mails so as not to leak sensitive information.

Email is a cost-effective method of communication commonly found in all areas of industries. Education industry is not an exception. Workforce in education industry spends fair amount of time in front of computer chasing up on emails. This is more so with jobs that deal with high volume of emails each day such as administrator in education industry. Managing incoming email is a critical matter to many because emails can herald important meetings, work messages, lunch, industry related information, upcoming events which many cannot afford to miss. Also, email is a means to transfer important documents in education agency. Often the documents contain international student's private information and scanned copy of application to apply for admission into education institution such as Universities, TAFEs and private colleges. At present we still find important work related emails in spam folder. Therefore there is still a need to improve accuracy of email classifiers using new and existing algorithms. One possible solution to improving spam classification algorithm is using a spam filter named Linger IG implemented in 2003 in an email classification system named Linger.

The basic principle of how this spam filter works bases on calculating information gain. However the problem with this solution is its accuracy in classifying non-spam emails into folders. Out of many email learner used by Linger, at best, Widrow-H off gives unstable accuracy which moves between 82.40% ~ 48.50% [1] when classifying emails into folders. Current solution such as context based email classification model [2] has been developed to better adapt at classifying emails into homogenous groups.

II. LITERATURE REVIEW

Study of literatures regarding automated email classification has found there are at least four different types of approaches to automated email classification: Traditional approach, Ontology-based approach, Graph-mining approach, Neural-Network approach. Among many solutions proposed by other researchers, Linger and context based email classification model were notable discoveries.

A. Traditional Approaches to email classification

Text classification algorithms have been adopted to email classification systems [3][4][5]. These includes Naïve Bayes algorithm [4] and Support Vector Machine [3] which tokenize the email for calculation determining similarity of emails to either spam or other useful type of email. Another famous and traditional approach to text categorization is NB. It learns training examples in priori probability given unseen examples. Basic concept is to calculate the probability it classifies documents based on learn advance before given unseen examples of categories and probabilities that attribute values belong to categories. The assumption that attributes are independent of each other underlies on this approach. Even though this theory violates the fact that attributes are dependent on each other, its performance is feasible. In text categorization [15] For vectorization performance of Naïve Bayes is very poor when features are co related to each other it is used popularly not only for text categorization, but also for any other classification problems, since its learning is fast and simple. Support vector machines is a method for classification of linear and non linear data. This algorithm uses non linear mapping to transform training data into higher dimension and then it search for linear optimal separating hyper plane. SVM optimizes the weights of the inner products of training examples and its input vector, called Lagrange multipliers, instead of those of its input vector, itself, as its learning process .It provides a compact description of the learned model. A major research goal is in SVM is to improve the speed in training and testing so that it become feasible option for large data set. In 1998, it was initially applied to text categorization by Joachims [16]. He explains the SVM in text categorization by comparing it with KNN and NB. Drucker et al. used SVM for implementing a spam mail filtering system and then compared it with NB in implementing the system in 1999 [17]. They conclude empirically that SVM was the improved approach to spam mail filtering than NB. In 2000, Cristianini and Shawe-Taylor presented a case of applying SVM to text categorization in their textbook [18].

Experiment conducted by Alsmadi and Alhami [3] have found that removing stop words in emails improve accuracy of email classification. Jason D. M Rennie [4] performed email classification using a Naïve Bayes algorithm in an email classification system named in file. An email classification method named Three-Phase Tournament method devised by Sayed et al [5] has shown very unstable accuracy ranging from 2% to 95%.

B.Ontology-based Approaches to email classification

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

Ontologies are proposed for several purposes related to the reusability of knowledge, knowledge sharing and analysis and also to separate commonalities from differences in the different knowledge areas. In the specific research subject of ontology classification or knowledge extraction of Email contents, there have been some research papers that tried to propose and introduce concepts usually found in Email contents. Such ontology can be also used for email validation or spam detection. For example, Taghva et al.'s (2003) paper proposed email concepts' extraction using Ecdysis Bayesian email classifier. Authors extracted email contents based on features collected from the extracted or trained data and also from DOE inclusionary or exclusionary records (Office of Civilian Radioactive Waste Management, 1992). Inclusionary concepts include: Organization, Department, Email Agent, and Message Topics. Exclusionary concepts include: Email Characteristics, Count Characteristics, and Attachment Type Characteristics. Each one of those entities includes

several related attributes. Protégé ontological tool (http://protege.stanford.edu/) was used to build and show the ontology. In our case, MIME parser is used to parse from emails many attributes of those described in Taghva et al. ontology.

Yang and Callan (2008) in 2008 presented also ontology to extract concepts from a corpus of public comments (Mercury and Polar Bear datasets). NGram mining is used to identify candidate concepts. Wordnet and surface text pattern matching are used to identify relationships among the concepts. Wordnet keywords are used to guide organization of concepts into intended hierarchal relationships. Part of Speech (POS) tagger from Stanford University is used as a text parser. Authors then used NGram based on words. Beseiso et al.'s (2012) paper proposed a method for concepts' extraction from email systems. Authors discussed one of the challenges of emails concepts' extraction as in most cases; users' emails are domains specific and highly dependent on the person, their profession, interests, etc. Authors extended NEPOMUK Message Ontology and defined email general concepts and domain specific concepts. Authors used Enron and custom email datasets for evaluation. Aloui and Neji's (2010) paper proposed a system for automatic email classification and question answering. The approach proposed three clusters of emails based on their general subjects: Procedural, social and cognitive functions. The paper extended an approach in the paper of Lê and Lê (2002). The 10 categories include: Requesting, Thinking, Discussing, Confirming, Referring, Clarifying, Complimenting, Complaining, Greeting and Sharing. Text clustering and classification can be used for a wide spectrum of applications. For example, Altwaijry and Algarny's (2012) paper used text classification methods to classify network income data and traffic and classify such data into threat (harmful) or non-threat data. A Naive Bayesian (NB) classifier is used. Such classifier is proved to be effective for classification in several different areas. Authors used public KDD IDS dataset for testing and training. Another major application area for classification especially in information retrieval systems includes image classification (De and Sil, 2012). In this specific paper, authors used fuzzy logic to assign soft class labels to the different images in the collected dataset. Such image classification can be used for search engines query and in most cases images are associated with embedded text or text located around those images. [3].

C. Graph-mining approaches to email classification.

Graph-mining approaches to email classification take advantage of semantic features and structure in emails by converting emails into graphs and matching template graphs with graphs made from each emails [8][9][10]. Typical graph mining algorithm converts emails into graphs. Substructures of graphs are then extracted from graphs. Parameters prune substructures. Representative substructures remain. Substructures are ranked just so that in case an email graph matches more than two representative substructures, emails go into a folder which the matched representative with higher rank. eMail Sift is a graph mining algorithm devised by Aery and Chakravarthy [8]. Aery and Chakravarthy have reported the email classification accuracy increased from 80% to 95% as the number of inputted emails increased from 60 to 370 [8]. On the contrary, a later work by Chakravarthy et al [9] named m-InfoSift showed that email classification accuracy decreased as number of folders increased. Accuracy of the email classification decreased from 100% to 91% as number of folders increased from 2 to 4 [9].

D. Current Best Selected Solution.

Graph-mining algorithm named Context-based Email Classification System was proposed by Wasi et al [10]. It consists of graph mining algorithm and Event Identification System. As shown in the Table 3, Accuracy of email classification was 80% when 300 emails were used for training. Accuracy of email classification rose to 85% when 750 emails were used [10]. Accuracy reached 88% when 1500 emails used. Accuracy became 93% as number of training emails counts 3000. As this result shows, it took 10 times more emails to raise accuracy from 80% to 93 % . A major flaw in this system bases on the huge number of emails it requires to reach accuracy of 100%. Also, the context-based email classification model does not have spam filter even though the model addresses clustering of homogenous emails into groups. Proposed work therefore needs to address this insufficiency to improve the model. An email classification system named Linger was developed by Jason Clark, Irena Koprinska and

Josiah Poon at University of Sydney. Linger uses neural network [1]. Linger uses a spam filter named LingerIG (Information Gain). As shown in the Table 3, result of their experiment showed that when LingerIG was used, Linger showed 100% accuracy at spam email classification.

III. SYSTEM DESIGN

1. SFECM: Components and Implementation

The system consists of three stages: Email Preprocessing, Feature Extraction and Email Classification. The proposed system runs POS Tagger on email in email preprocessing stage to turn email texts into email features. At feature extraction stage, proposed system filters Spam from a set of inputted emails. Then from filtered emails, sign-off words, greeting words, keywords are extracted to form email graph. At this stage, template graphs update using new email graphs. Template graphs are then ranked in email classification stage to be assigned to represent relevant folder. Then email graphs are matched to representative template graphs and placed to folder of the representative template graph that graph matches most. Detailed diagram of this proposed work is presented in Figure 1.

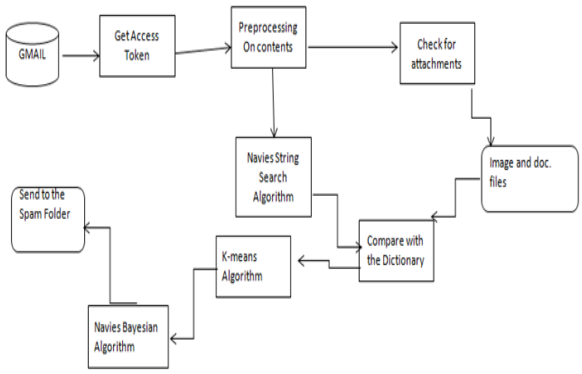


Figure 1 System Architecture

2. Spam Filter: Algorithm

The proposed solution's algorithm at spam filtering event is presented below:

Algorithm(1) : Proposed Spam Filter

INPUT: Test email samples (E) = {E1, E2, … , EN}

OUTPUT: Classified emails (E')= {E1',E2',…,EN'}

BEGIN

Step 1: USE a set of test emails as sample emails.

Step 2: INPUT a sample email into proposed solution's spam filter.

Step 3: Start a loop

For each email feature in EN if Email contains email feature that matches feature in spam feature list, increment count. Keep counted number of matches as numMatchSpam; number of email features in the email that match features from spam filter's list of spam email features.

END FOR

Step 4: Start another loop

For each email feature in EN if Email contains email feature that matches feature in non-spam feature list, increment count. Make integer variable in source code numMatchWork; number of email features in the email that match features from spam filter's list of work email features. Keep counted number of matches as numMatchWork.

END FOR

Step 5: Calculate Entropy/Impurity

Calculate the spam email features are contained in email by dividing number of spam email features by number of email features in the email. Call this impuritySpam(impSpam).

Calculate impSpam= numMatchSpam / number of Features In Email;

Find the work email features are contained in email by using below formula.

Call this impurityWork (impWork) impWork = numMatchWork/ number of Features In Email;

Step 6: Move email to either spam or keep email in inbox.

If impSpam > Average Information Gain of all Spam emails

Move the email to spam folder directory.

If impSpam > impWork Move the email to spam folder directory.

If impWork > Average Information Gain of all non-spam emails Keep the email in inbox.

If impWork > impSpam Keep the email in inbox.

Step 7: End of Algorithm

The below figure specified the system architecture of our project.

IV.E-MAIL SPAM FILTERING: THE ALGORITHMS

A. Naive Bayes (NB)

The Na¨ıve 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, Na¨ıve 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. Na¨ıve 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 email. 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 Na¨ıve Bayes technique used Bayes theorem as shown in formula below.

$$P(spam \mid word) = \frac{P(spam).P(word|spam)}{P(spam).P(word|spam) + P(non-spam).P(word|non-spam)}$$

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.

B. K-means Clustering

Many existing databases or datasets are unlabeled, because large amounts of data make it difficult for humans to manually label the categories of each instance. Hence, unsupervised learning is needed. Besides being unlabeled, several applications are characterized by high dimensional data (e.g., text, images). Unsupervised learning means there is no teacher in the form of the class label. One type of unsupervised learning problem is clustering. The goal of clustering is to group similar data together. In database management, clustering data is the process of dividing data element (input data) into "similar" groups so that items in the same group are as similar as possible, and items in different group are as dissimilar as possible. It is one of the most useful methods in data mining for detection of natural groups in a dataset-Means clustering algorithm, and group's data based on their feature values into K clusters. In the classification, the objects are assigned to predefined classes, whereas in clustering the classes are formed. There are general categories of cluster analysis methods such as Tree clustering, block clustering, EM clusters and Kmeans clustering. Clustering methods may be divided into two categories based on the nature of the data and the purpose for which clustering is being used such as fuzzy clustering (each data element can belong to more than one cluster and is a mathematical method for classification such as expectation maximization method) and hard clustering (each data is divided into distinct cluster where data elements belong to exactly one cluster such as K-means clustering). K-means algorithm, is numerical and one of the hard clustering methods, this means that a data point can belong to only one cluster (group). This paper utilized the K-means clustering algorithm to group the messages (emails) based on the similarity of their attributes or features into K disjoint groups. K is a positive number initialized early, before the algorithm start, to refer to the number of required clusters (groups). Basically, K-means clustering inspects the feature of each object, such that the objects within each cluster are similar to each other and distinct from objects in other clusters. K-means is an iterative algorithm, it starts by defining an initial set of clusters and the clusters are repeatedly updated until no more improvement is possible (or the number of iterations exceeds a specified limit). The use of SVM algorithm for spam detection using massive data, are time and memory consuming. Therefore, the researcher used a K-means clustering to solve the problem of time and memory consuming, by dividing the huge data into subgroups according to similarity, to improve the accuracy of spam detection. The steps of Kmeans clustering algorithms are seen in Figure 4 showed K-means clustering step. The K-means algorithm starts with initial K centroids, then it assigns each remaining point to the nearest centroid, updates the cluster centroids, and repeats the process until the K centroids do not change .Standard K-means clustering utilizes Euclidean distance to measure the difference between email messages (or Euclidean

distance is used as a measure to describe the similarity between data objects).

$$d(X, y) = \sqrt{\sum_{i=1}^{n} (X_i - y_i)^2}$$

The position of a point in a Euclidean n-space is a Euclidean vector. So, X (X ,X , ., X) and Y (Y , Y , ..Y) are Euclidean vectors, starting from the origin of the space, and their tips indicate two points.

C. Optical Character Recognition Technique

Through the scanning process is the digital image of the original document is captured. Whereas OCR optical scanners generally consist of a transport mechanism plus a sensing device that converts light intensity into gray-levels. Printed documents usually consist of black print on a white background; hence, when performing OCR, it is common practice to convert the multilevel image into a bi-level image of black and white. Often, this process is known as thresholding, is performed on the scanner to save memory space and computational effort.

Location and segmentation Segmentation is a process that determines the constituents of an image, it is necessary to locate the regions of the document where data have been printed and distinguish them from figures and graphics. For instance, when performing automatic mail-sorting, the address must be located and separated from other print on the envelope like stamps and company logos, prior to recognition.

Pre-processing The image resulting from the scanning process may contain a certain amount of noise depending on the resolution of the scanner and the success of the applied technique for thresholding, the characters may be smeared or broken. Some of these defects, which may later cause poor recognition rates, can be eliminated by using a preprocessor to smooth the digitized characters.

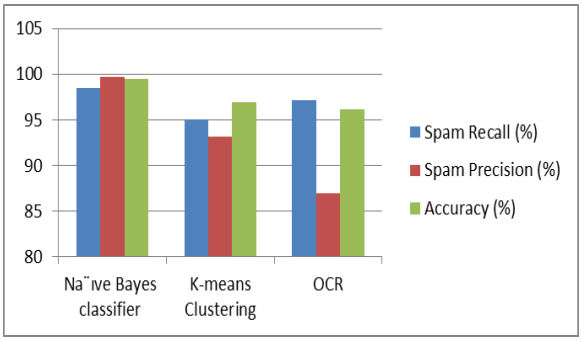
Feature extraction The objective of feature extraction is to capture the essential characteristics of the symbols, and it is generally accepted that this is one of the most difficult problems of pattern recognition. The most straight forward way of describing a character is by the actual raster image. Another approach is to extract certain features that still characterize the symbols, but leaves out the unimportant attributes.

Post processing They are two types of post processing, 1. Grouping 2. Error-detection and correction.

V. RESULT

In the below graph and table show that the sapm recall(%), spam precision(%) and accuracy(%) of Naïve Bayes Algorithm, K-means Clustering Algorithm and OCR. We can see that the accuracy of Naïve Bayes Algorithm is 99.46%, and for K-means Clustering is 96.9% and for OCR is 96.2%.

Algorithm	Spam Recall (%)	Spam Precision (%)	Accuracy (%)
Naïve Bayes classifier	98.46	99.66	99.46
K-means Clustering	95	93.12	96.9
OCR	97.14	87	96.2



VI. CONCLUSION

This paper has identified that 100% accuracy in spam classification of email system is still an unmet need. Project has drawn upon the work of the existing email classification systems known as ‘context-based email classification system’ and ‘Linger’ to address the unmet need. Main steps of the context-based email classification system begins with preprocessing email using POS Tagger then it extracts several email features to transform emails into graphs and then graphs are matched to representative graph so that emails are classified to the folder which the representative graph with highest match represent. Linger implements information gain classifier for filtering spam and use neural network to classify emails into homogenous clusters. The proposed system adopts spam filter from Linger to reinforce the accuracy needed to separate spam emails without any mistake. Proposed solution provides 100%

accuracy at filtering spam from a set of mixed emails. As far as the experiment shows, processing time between using spam filter and not using spam filter differ insignificantly. It is important to however to stress the need to reduce the processing time of the spam classification because processing time of 0.1 second is an unmet need in this solution.

VI. FUTURE SCOPE

Though, thesis has made efforts towards solving the problem of Spam E-mail using legislative, behavioral and technological measures, the solution proposed are not complete solutions. The problem of Spam E-mail and Anti-Spam solution is game of cat and mouse since, every day Spammer will come up with new techniques of sending Spam E-mails. This work has given the potential direction for classification of the Spam E-mails.

The future efforts would be extended towards:

- Achieving accurate classification, with zero percent (0%) misclassification of Ham E-mail as Spam and Spam E-mail as Ham.
- The efforts would be applied to block Phishing E-mails, which carries the phishing attacks and now-days which is more matter of concern.
- Also, the work can be extended to keep away the Denial of Service attack (DoS) which has now, emerged in Distributed fashion called as Distributed Denial of Service Attack (DDoS).

REFERENCES

[1] J. Clark, I. Koprinska and J. Poon, "Linger - A Smart Personal Assistant for E-Mail Classification", in International Conference on Artificial Neural Networks, 2003, pp. 274–277.

[2] S. Wasi, S. Jami and Z. Shaikh, "Context-based email classification model", Expert Systems, vol. 33, no. 2, pp. 129-144, 2015.

[3] I. Alsmadi and I. Alhami, "Clustering and classification of email contents", Journal of King Saud University - Computer and Information Sciences, vol. 27, no. 1, pp. 46-57, 2015.

[4] J. Rennie, "ifile: An Application of Machine Learning to E-Mail Filtering", in Proceedings of the KDD (Knowledge Discovery in Databases) Workshop on Text Mining, 2000.

[5] S. Sayed, "Three-Phase Tournament-Based Method for Better Email Classification", International Journal of Artificial Intelligence & Applications, vol. 3, no. 6, pp. 49-56, 2012.

[6] M. Fuad, D. Deb and M. Hossain, "A trainable fuzzy spam detection system", in 7th International Conference on Computer and Information Technology, 2004.

[7] S. Youn and D. McLeod, "Spam Email Classification using an Adaptive Ontology", JSW, vol. 2, no. 3, 2007.

[8] M. Aery and S. Chakravarthy, “eMailSift: Email Classification Based on Structure and Content,” Data Mining, Fifth IEEE Int. Conf., pp. 18–25, 2005.

[9] S. Chakravarthy, A. Venkatachalam, and A. Telang, “A graph-based approach for multi-folder email classification,” Proc. - IEEE Int. Conf. Data Mining, ICDM, pp. 78–87, 2010.

[10] T. Ayodele, S. Zhou, and R. Khusainov, “Email Classification Using Back Propagation Technique,” Int. J., vol. 1, no. 1, pp. 3–9, 2010.

[11] D. Patil and Y. Dongre, “A Clustering Technique for Email Content Mining,” Int. J. Comput. Sci. Inf. Technol., vol. 7, no. 3, pp. 73–79, 2015.

[12] K. Taghva, J. Borsack, J. Coombs, A. Condit, S. Lumos, and T. Nartker, “Ontology-based classification of email,” Proc. ITCC 2003. Int. Conf. Inf. Technol. Coding Comput., pp. 194–198, 2003.