International Journal of Web-Based Learning and Teaching Technologies
Volume 15 • Issue 2 • April-June 2020

ML-EC^2: An Algorithm for Multi-Label Email Classification Using Clustering

Aakanksha Sharaff, National Institute of Technology Raipur, Raipur, India
Naresh Kumar Nagwani, National Institute of Technology Raipur, Raipur, India

ABSTRACT

A multi-label variant of email classification named ML-EC^2 (multi-label email classification using clustering) has been proposed in this work. ML-EC^2 is a hybrid algorithm based on text clustering, text classification, frequent-term calculation (based on latent dirichlet allocation), and taxonomic term-mapping technique. It is an example of classification using text clustering technique. It studies the problem where each email cluster represents a single class label while it is associated with set of cluster labels. It is multi-label text-clustering-based classification algorithm in which an email cluster can be mapped to more than one email category when cluster label matches with more than one category term. The algorithm will be helpful when there is a vague idea of topic. The performance parameters Entropy and Davies-Bouldin Index are used to evaluate the designed algorithm.

KEYWORDS

Classification Using Clustering, Email Clustering, Latent Dirichlet Allocation, Multi Label Classification, Non-Negative Matrix Factorization, Taxonomic Terms

INTRODUCTION

As the number of incoming email messages increases, it becomes very difficult for the users to handle these emails. There are different tools for facilitating the management of incoming emails. e.g. use of threads and use of folders or labels for classifying incoming emails. Email categorization (classification) is a process of classifying emails to discrete set of predefined categories. Categorization of emails becomes difficult due to the enormous volume of emails (sent/received) as well as different topics may be discussed in an email. Hence, categorizing emails manually becomes a heavy burden for users. Categorizing emails by identifying categorical terms is an important issue. It adds semantics to email management. Multi label email classification is not explored in detail in literature.

The objective of this paper is to detect similar emails and categorize them in multi label classes as well as to identify (discover) categorical terms in a different way by adapting latent Dirichlet allocation (LDA) as topic modelling approach. Hence, to accomplish the objectives; an algorithm Multi Label Email Classification using Clustering (ML-EC^2) is proposed in this paper. It is a type of multiple classification of emails. Classifying emails into classes can be topic oriented or group oriented. Topic

DOI: 10.4018/IJWLTT.2020040102

oriented classification includes the emails belonging to such as "job opportunities", "entertainment" etc., whereas group oriented classification can be "place specific", "people specific", "course or project specific". In multi label classification of emails, each email file may belong to one or more number of categories. The algorithm ML-EC^2 has been designed for creating the categorized groups of similar emails using textual similarity of email attribute. It is a multi-label text clustering based classification algorithm, where one email cluster can be mapped to more than one email category. If in a single label there are 1500 emails on the same topic suppose on entertainment then it becomes very difficult and time consuming to find a desired email. Hence to overcome this problem a multi class categorization of email has been designed and implemented. A hierarchy is formed with a single label/class. For example, in entertainment class a hierarchy of music, videos, movies etc. can be formed and the email associated with concerned label (class) can be placed on these sub-hierarchies. The proposed technique of email categorization can reduce the problem of email overload.

LITERATURE REVIEW

Managing huge amount of emails received from users is a very challenging problem which needs to be solved in an effective and efficient way. Various researches have been done in the field of email mining. Some of the surveys done are as follows.

Park & An (2010) proposed an Email multicategory classification approach using semantic features and a dynamic category hierarchy reconstruction method in which the user reorganizes all e-mail messages into categories. Guan & Yuan (2013) reviews the existing work on mislabeled data detection techniques for pattern classification and classifies them into three types: Local learning-based, ensemble learning-based and single learning-based methods. The author Armentano & Amandi (2014) presented an approach to label the incoming emails based on user preference; a set of experiments using Google's webmail system, Gmail is performed to obtain a good rate of acceptance of the agent interactions. Alsmadi & Alhami (2015) introduced an algorithm for performing clustering and classification of email text corpus. They have proposed a model for classification of emails based on subject and folder using N-grams. Islam et al. (2009) proposed a new technique of e-mail classification based on the analysis of grey list (GL), which uses multi-classifier classification ensembles of statistical learning algorithms.

Sakurai & Suyama (2005) proposed a method to extract key concepts from e-mails and presents their statistical information which has been applied to three kinds of analysis tasks: a product analysis task, a contents analysis task, and an address analysis task in which acquired concept relation dictionaries gave high precision ratios in the classification. Koprinska et al. (2007) investigated the use of random forest for automatic e-mail filing into folders and spam e-mail filtering. Sappelli et al. (2005) presented an approach of categorizing emails that can alleviate the common problem of email overload. Sun et al. (2010) developed a clustering based algorithm for detecting duplicate emails by using hash function. Gomez et al. (2012) classified emails into spam and ham by reducing the dimensionality of email using Principal Component Analysis (PCA) and compare several feature selection methods with novel content-based statistical feature extraction techniques. Recently a Singular Value Decomposition (SVD) method has been proposed by Zareapoor et al. (2015) to classify email in order to compress sparse email data but retaining the most informative and discriminate features of email. Aloui & Neji (2010) developed a multi-agents system EQASTO (E-mails Question Answering System using Text-mining and Ontological techniques) to relieve the burden of e-mails processing by using a combination of text-mining and ontological techniques to classify semantically e-mails, fetch, generate, and send answers automatically to learners. Bekkarman et al. (2004) presented an email foldering scheme by using two large corpora i.e. Enron and SRI and point out the challenges arises by using email foldering scheme instead of traditional document classification. The author Beseiso et al. (2012) proposed an ontology based email knowledge extraction process which reduces the users time and resources to handle unstructured Email messages.

Carmona-Cejudo et al. (2011) presented GNUsmail, an open-source extensible framework which incorporates feature extraction, feature selection, learning and evaluation methods in the domain of email classification. Bermejo et al. (2011) proposed a method for Email folder classification based on learning and sampling probability distributions. Manco et al. (2008) proposed a unified frame work for handling huge amount of emails received from users and cluster these emails based on similar feature to a user-defined folder. Pattern discovery and clustering approach has been applied for email classification. Zhang et al. (2016) proposed a Label Compression (LC) method termed as robust label compression (RLC) to deal with outliers present in feature space. This method reduces the time cost and also improves classification performance for multi label classification. Dehghani et al. (2016) developed a model Alecsa an attentive learning approach for automatic email categorization. Alecsa uses structural aspects of email as distinguishing feature to identify the behavior of users while they attempt to categorize a new email. Schmid et al. (2015) categorized texts to address authorship attribution problem. Sharaff & Nagwani (2019) incorporates the textual similarity between email attributes using Latent Dirichlet Allocation in identifying categorical terms. Clustering of emails i.e. forming a group of similar emails is a key area of email mining. Various algorithms and approaches has been used to form cluster of emails such as pattern matching, quantitative profiles (Špitalský, & Grendár, 2013), based on semantics (He, B., Li, Z., & Yang, N. 2014), unsupervised clustering using labeling of similar contexts (Kulkarni, & Pedersen, 2005). Clustering of emails has various applications; automatic answering systems (Li, et al. 2006), managing email overload (Xiang, 2009), email forensic analysis, hierarchical user feedback (Huang, & Mitchell, 2008).

METHODOLOGY

Email classification is a challenging area as email contains large numbers of attributes (features) (Wang, Liu, Feng, & Zhu, 2015). Attribute selection is the main deciding factor in email classification problem. Attributes in email can be related to headers section or can be related to the content section. Header section includes "to mail_id (receiver)", "from mail_id (sender)" addresses, "date" and "time" of email which can show the trend of email, etc. whereas content section includes "subject", "body", "words", "sequence of words" etc. Content section of emails i.e. subject and body part of email messages are considered in this work to provide multi label categorization of email clusters. The methodology of the proposed work is presented in Figure 1. The overall research work is divided in two phase. The first phase is to generate taxonomic (categorical) terms and second phase is to provide labeling of email concerning with its associated category.

Firstly, emails are retrieved and then preprocessing activities such as parsing, tokenization, removal of stop words, stemming are used to eliminate the irrelevant words which do not have any significance in the process (Berry, 2004). Once the preprocessing activities have been applied, email attribute similarity has been computed by forming clusters of emails. K-means clustering, agglomerative clustering and NMF clustering algorithm has been used to form email clusters. Once the clusters are formed, then frequent terms are identified and clusters are labeled using these unique frequent terms (features). LDA (Wei, & Croft, 2006) a topic modeling has been used to generate categorical terms. LDA is the topic modeling based clustering algorithm where clusters are decided on the basis of topics generated and content similarity (Sharaff, & Nagwani, 2016). The labels generated are then mapped to categorical terms (belonging to pre-defined category) and then email clusters with its category are identified. The final step is to evaluate the results obtained by using performance parameter measure.

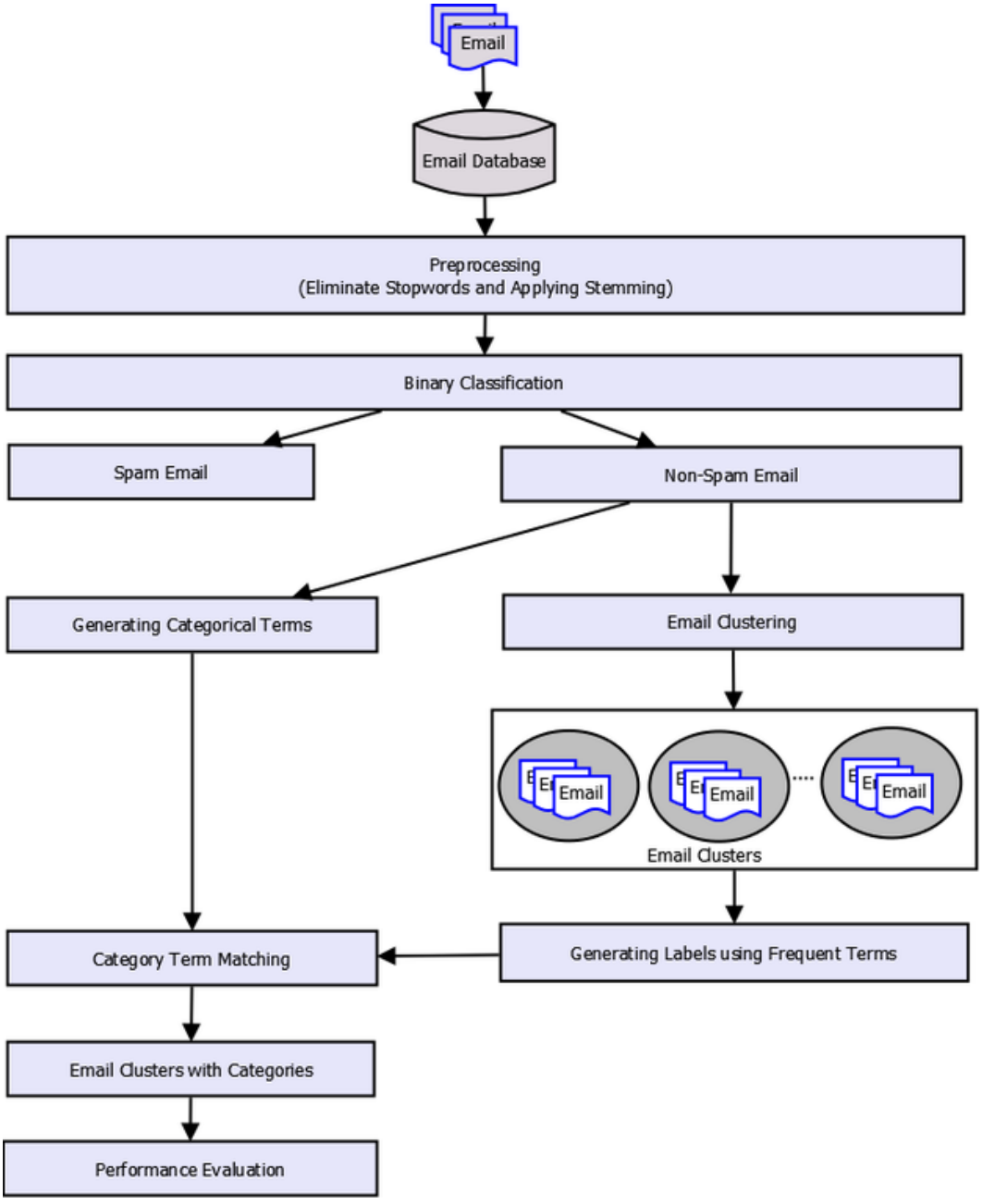
The ML-EC^2 Algorithm

Let C denote the category taxonomic terms instance space and L denote the class/cluster label space.

The task of supervised learning based ML-EC^2 is to learn a function:

$$f: C \rightarrow L$$

Figure 1. Major steps in ML-EC^2 algorithm



from the training set:

$$\{(c_i, l_i) \mid 1 \leq i \leq m\}$$

Here, $c_i \in C$ is an instance of category characterizing the properties (features) of a category and $l_i \in L$ is the corresponding class/cluster label characterizing its semantics associated with c_i.

Algorithm ML-EC^2

Returns: a) K-Clusters consisting of similar emails
b) Categories of each cluster
Arguments: τ- Category Threshold
N- Number of frequent terms in cluster labels
K- Number of cluster to be formed
W_s- Similarity weight for attribute Subject
W_c- Similarity weight for attribute Content

Step 1 Preprocessing of data

For each email message

1a) Parse and extract email attributes from each email file.

1b) Perform **Stopping and Stemming** to email attributes (subject and content).

Step 2 Email Binary Classification

Classify emails as either spam or non-spam (ham) messages using classification algorithm.

Step 3 Email Clustering

Apply **K-means** clustering algorithm to create email clusters

3a) Randomly choose any K emails as a centroid (C) of each cluster.

3b) For each email E_i and the centroid email (C_j), calculate the textual similarity between attributes subject and content, using similarity weights W_s and W_c such that the similarity value is normalized to 1. i.e. $W_s + W_c = 1$.

3c) $Sim(E_i, C_j) = W_s \times Sim(E_{i\text{-subject}}, C_{j\text{-subject}}) + W_c \times Sim(E_{i\text{-content}}, C_{j\text{- content}})$

3d) Calculate the relative distance (d) between emails and all the centroids by using cosine similarity (distance = 1 – cosine similarity).

3e) Put each email into the cluster of their nearest (d will be minimum) centroid.

3f) Calculate new centroids (C_{new}) by taking the mean of the distance in each cluster.

3g) Repeat step 3b to 3f until the value of old centroids equals to the new centroids.

OR

Apply **Agglomerative** clustering algorithm to create email clusters

3a) Find the email messages or email cluster which have highest similarity (or minimum dissimilarity).

3b) Now successively merge these email clusters to form a cluster hierarchy based on their similarity.

3c) Repeat step 3b until a single cluster remains.

OR

Apply **NMF** clustering algorithm to create email clusters

3a) Construct Term Email Matrix (TEM) from email corpus

3b) Determine the two non-negative matrices W and H from TEM.

3c) Normalize W and H obtained from step 2b.

3d) Use matrix H to obtain cluster label from each email.

Step4 Cluster Label Generation - Using Frequent Terms for a Cluster
For each cluster C_i, get the lists of emails belonging to this cluster.
4a) Extract the subject and content of these emails.
4b) Concatenate this textual data to form the cluster text data.
4c) Calculate the N frequent terms $\{T_{i1}, T_{i2} \cdots T_{iN}\}$ from each cluster text data, and assign them to these clusters as cluster labels.
4d) Label $(C_i) \leftarrow \{T_{i1}, T_{i2} \cdots T_{iN}\}$
Step5 Mapping Clusters to Classes
For each cluster C_i, get each term T_{iK} in the Label (C_i) (cluster label) and match it with the email taxonomic terms. The match indicates the belongingness of cluster in that email category. If number of matching term is more than the category threshold (τ) then put that email to that category. If the matching term with taxonomic terms is more than the category threshold (τ) of two or more number of categories, then the cluster will belong to all of these categories.
Step6 Performance Evaluation and Output Representation
Calculate performance parameters Entropy and Davies-Bouldin Index over Sample size, Number of Clusters.

The classification using a clustering algorithm for multi-label classification of emails has been presented in this section. The algorithm performs the task of multi label classification in five steps. The input to this algorithm is set of emails, the number of frequent terms obtained through LDA, number of clusters to be formed, category threshold, similarity weight for subject Ws and content Wc. The similarity weight of Ws and Wc should be such that the sum of these two will become 1. The similarity weight has been chosen more for Ws as more emphasis is given on subject rather than content. For experimental analysis Ws has been taken as 0.6 and Wc as 0.4. Category threshold represents the value which determines the number of frequent terms above which the email cluster will belong to one category or more than one category. The output of the algorithm will be the number of similar emails belonging to a particular cluster and the category of each cluster. One cluster can belong to more than one category is the idea behind this research. The overall approach of proposed work has discussed in detail below:

- Preprocessing

The first step is to parse the email data retrieved and extract the required features. Extraction of features is done by applying preprocessing activities over email data. Preprocessing of email data involves two major preprocessing activities namely Stopping and Stemming. Stopping is used to remove irrelevant words such as “is”, “to”, “am”, “for”, “are” etc. Stemming is used to form root words such as “covers”, “covered” and “covering” will be converted to word “cover” so that all the words should be uniformly considered as unique word.

- Email Binary Classification

Once the preprocessed email has been obtained, emails are classified into classes spam or non-spam. This classification is done by using any classification technique. Non-spam email messages are considered for performing the research.

- Email Clustering: Create Email Clusters using K-means or Agglomerative or NMF

Clustering algorithm has been used to form the emails clusters. Each cluster contains the emails which are similar in their content to derive relevant information from a huge corpus of emails. Clusters are formed by using one of the clustering technique among K-means or agglomerative or NMF.

- Cluster Label Generation

After forming the clusters, each cluster has to be labeled. The cluster label is generated by selecting and concatenating the N-most frequent words that are present in the cluster. Selection of value of N depends on the extent of details that the user wants to represent for each cluster. If N is too high, then the results will not be generalized and would over fit the data, whereas if N is too low, useful information may get lost. If the terms obtained in cluster label matches with the categorical terms with a threshold value (category threshold) then that cluster will belong to that category. If the matching term (frequent terms obtained through cluster label) with taxonomic terms is more than a category threshold of two or more number of categories then the cluster will belong to all of these categories.

- Mapping Clusters to Classes

The various classes to which each cluster is to be assigned have been generated using LDA. Each class consists of a set of words which describes the gist of the classes. Hence the clusters are mapped to classes based on their labels. The words on the label of a cluster are matched with the words in each class. The most suitable class based on this matching is selected for that cluster. If none of the classes mapped to the cluster, then the cluster is deemed as uncategorized. Multiple clusters can be assigned to a single class, which is a many-to-one form of mapping.

- Performance Evaluation and Output Representation

The performance parameters Sample size, Number of Clusters, Entropy and Davies-Bouldin index are calculated.

Email Clustering

Several text clustering approaches are used for forming clusters of email messages. In this paper, K-means, Agglomerative and Non-negative Matrix Factorization (NMF) clustering approaches are discussed.

- K–means

K–means is an iterative clustering approach. The main idea behind k–means clustering approach is to select initially k seeds (or messages) from the original data and assign the email messages to email messages to one of these k seeds based on their closest similarity. In the next step, the centroid of assigned messages to each seed is computed in order to define a new seed for that cluster. The process continues till it converges. The main disavantage with k-means is the initial selection of seeds which affects the quality of cluster formed. Hence agglomerative clustering approach is used to decide the initial k number of seeds (Alsmadi & Alhami 2015).

- Agglomerative Clustering

Agglomerative clustering is a hierarchical clustering technique based on bottom up approach. It creates a tree like a hierarchy and improves the searching process. The concept behind agglomerative clustering approach is to successively merge the messages into clusters by finding the best pairwise similarity between the messages and groups. This clustering technique forms a dendogram or cluster hierarchy in which each leaf node represents an individual message, internal nodes represents merged clusters (group of messages). When two groups of clusters are merged, they form a new node (large merged group) in tree. This process of forming a chain of nested clusters continues until a single cluster is formed which consists of all the messages in a corpus.

- Non-negative Matrix Factorization (NMF)

NMF is a feature transformation method based on analysis of term document matrix. NMF can be used to determine word clusters instead of document clusters and is particularly suitable for clustering. Suppose a non-negative data matrix V is given, the objective of NMF is to find an approximate factorization V≈WH into non-negative factors W and H. Two non-negative matrices W and H are determine from Term Email Matrix (TEM) such that it should minimize the objective function of error function J described in Equation (1):

$$J = \frac{1}{2}V - WH \tag{1}$$

EXPERIMENTS

The experiments are performed using Java programming language, and implementation of LDA, is carried using Java-based API namely, Mallet (McCallum, 2002). Experiments are performed on the popular and freely available Enron email corpus dataset (Klimt, & Yang, 2004). The Enron corpus consists of the Enron Corporation emails, with 200,399 messages from 158 unique users. Enron is considered to be the largest publicly available email dataset. All the experiments were performed on the sent mail folder of Enron dataset.

Generating Taxonomic Terms

A large number of emails and the large number of unique terms are used as inputs to the clustering and classification process; hence discovering categorical terms (unique features) is a major challenge. Categorical terms are used to define the category based on topic terms. The categorical terms are generated using topic modelling approach LDA, frequent term analysis. The categories based on the categorical terms is identified from email corpus is presented in Table 1.

Mapping of Categorical Terms to Category

The categorical terms are generated for Enron email dataset using LDA. Table 1 consists of eight major categories of email with their corresponding preprocessed (after stopping and stemming) categorical terms. The terms which are not covered in the above mentioned categories are considered as uncategorized terms (or others).

Classifier Performance Evaluation

Various performance parameters exist to evaluate the performance of results obtained. Some of the parameters used in experiments are described below.

Table 1. Categorical terms

Category Name	Categorical Terms
Resources	gas, natural, oil, plant, tree, power, internet, web, electricity
Information Technology	information, message, e mail, news, communication, contact, research, data, transmission, project, images, question, request, call, to/from, review, list, spam, services, click, program, system, link, server, online, internet, web, e trade, file
Interface	color, changes, font, size, center, left, class, align, font-size, image, width, height, position, list, table, click, intended, updated, attached, link
Time and date	time, hour, rate, night, day, future, schedule, sat, fri, meeting, sun, week, travel, october, november, year, june, date, daily, over, morning, behold, set, wait, tomorrow
Business	report, work, schedule, meeting, agreement, credit, management, conference, market, company, business, trading, deal, stock, services, contract, corporate, manage, project, order, product, sell, agreement, program, employee, member, team, people, group, launch
Location	london, texas, california, houston, address, position, market, company, class, travel, miles, east, north, left, migration, states
Finance	credit, rate, market, business, trading, deal, stock, order, seal, price, change, bill, financial, buy, fare, transaction, tax, billion, total, e trade, credit
Uncategorized	Terms that are not included in above eight categories

- Sample Size

The number of emails represents the sample size. The samples are selected from a huge database of emails randomly. The sample size should be large for better and useful results.

- Number of Clusters

The number of clusters affects the information extracted substantially and should be carefully selected along with evaluation measures like entropy and DB Index.

- Entropy

It represents the average information content of the clusters formed. Less the entropy, the more is the information obtained from clusters. Hence, it gives better results when entropy is less and thus the classification of emails will be better. Entropy of a cluster k can be calculated as given in Equation (2):

$$E_k = -\sum_j p_{jk} \log(p_{jk}) \tag{2}$$

where p_{jk} is the probability that a member of cluster k belongs to class j.

- Davies-Bouldin Index

It is an internal evaluation metric which gives a measure of inter-cluster and intra-cluster density. It measures the average distance between each cluster and finds the most similar one. Lower the DB Index better is the result as it implies high inter-cluster distance and low intra-cluster distance between emails. The DB-index is calculated using Equation (3):

$$DB_k = \frac{1}{k}\sum_{i=1}^{k} \cdot max_{j=1,\ldots,k}\left\{\frac{distance(c_i)+distance(c_j)}{c_i - c_j}\right\} \quad (3)$$

When i≠j, distance (c_i) is the sum of distances (cosine similarity) of all emails of cluster i to its centroid and distance (c_j) is the sum of distances (cosine similarity) of all emails of cluster j to its centroid.

The results obtained from the experiments are analyzed by considering four parameters, namely sample size, number of clusters, entropy and davies-bouldin index. The observations made from each experiment has been discussed below.

Effect of Number of Clusters Over Entropy

As entropy measures the disorder in clustering. It represents the information content of the clusters of emails, hence the less the entropy, the better is the quality of cluster (Kovács, Legány, & Babos, 2005). It can be observed from Figure 2 that when the sample size is 400 and number of cluster is low, the entropy is too low for all clustering techniques; whereas when number of clusters increases entropy also increases which signify that a good cluster can be formed with lower number of clusters. It is found that K-means clustering algorithm gives minimum entropy when compared with agglomerative and NMF clustering when two numbers of clusters are generated. As the cluster size increases agglomerative and K-means both performs well but NMF comes out to be the least performer.

Effect of Number of Clusters Over DB Index

DB Index is used to signify the quality of the clusters formed (Kovács, Legány, & Babos, 2005). Taking a sample of 400 emails, and varying the number of clusters, a low value of DB Index signifies a good performance by the clustering algorithm. Same as that with entropy, the general trend is that increasing the number of clusters can help in getting better performance, but only up to a limit after which it will taint the outcome. It can be observed from Figure 3 that NMF performs better than the other two algorithms whereas K-means and agglomerative gives approximately same result but when the cluster size is small K-means outperforms after NMF.

Effect of Sample Size Over Entropy

Sample size is the number of emails that were clustered and concurring with the idea that increasing sample size results in better outcome. It can be clearly observed from Figure 4 that the entropy increases steadily which can be interpreted as a constant increase in the information retrieved by

Figure 2. Effect of number of clusters over entropy for ML-EC²

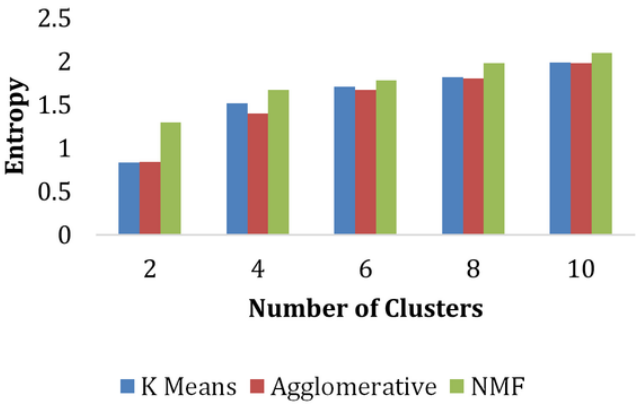


Figure 3. Effect of number of clusters over DB index for ML-EC²

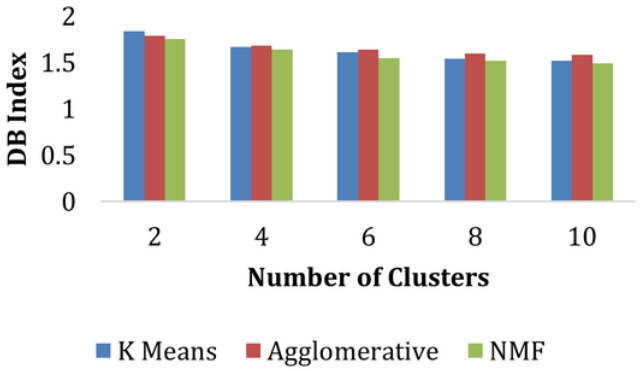
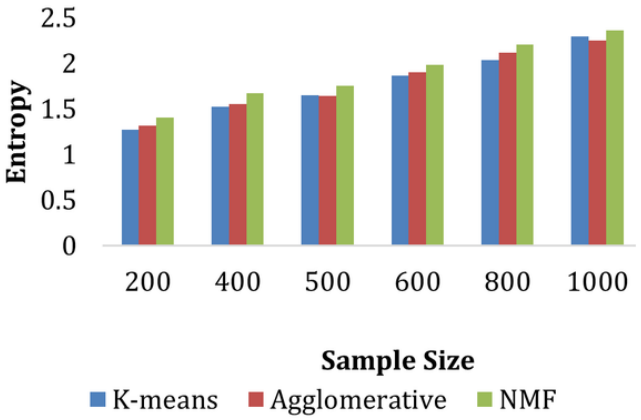


Figure 4. Effect of sample size over entropy for ML-EC²



proposed procedure. Among all clustering algorithm K-means performs better for different sample size. But it can also be predicted that as the number of emails increases entropy also increases which means the obtained cluster quality is not good. Hence smaller the number of email considered better entropy will be achieved.

Effect of Sample Size Over DB Index

When the number of emails keeps on increasing the DB Index decreases in almost a steady rate, which signifies a consistent and good performance. It is also easily observed from Figure 5 that the NMF clustering algorithm performs better in terms of DB index.

Email and Cluster Analysis Using Clustering Approach

The Email clusters generated using K-means, Agglomerative and NMF clustering approach from Enron email database is presented in the Table 2. In the Table E-id represents Email-id and C-id represents cluster Label id. C-id is a group of E-id which forms clusters of emails. Multiple C-ids represents the different cluster formation of emails. The clusters are formed by finding the similarity between emails. The Table shows the cluster label generated through frequent words and their corresponding email category.

Figure 5. Effect of sample size over DB index for ML-EC²

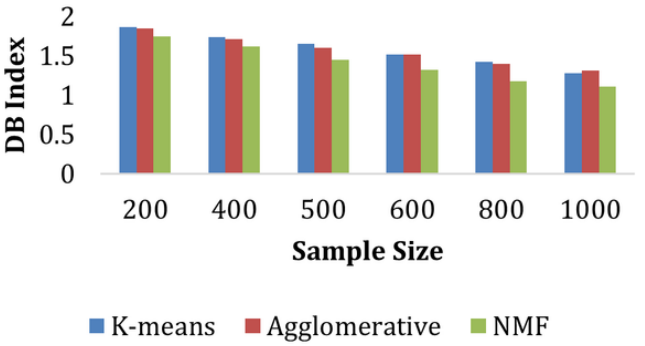


Table 2. Email and cluster label analysis using clustering approach

S. No.	E-id	Email Subject	Frequent Words	K-Means		Agglomerative		NMF	
				C-id	Category	C-id	Category	C-id	Category
1	10027	Future, meeting	from, forward, future, meeting	C-1	Time and Date	C-1	Time and Date	C-1	Time and Date
2	695	Buy, tomorrow, rate	tomorrow, from, frank, buy, launch, rate, tax	C-4	Business, Finance	C-3	Business, Finance	C-2, C-1	Business, Finance, Time and Date
3	15534	Coenergy trading company	energy, index, coenergy, trading, project, file, oil	C-3	Business, Resource	C-3	Business	C-2, C-4	Business, Finance, Resource
4	15206	Password, set, from	trade, set, from, password, forward, project, file, seal	C-2	Information Technology	C-2	Information Technology	C-3, C-2	Information Technology, Business, Finance
5	3410	Deal	report, fare, total, services, deal	C-4	Business, Finance	C-3	Business	C-2	Business, Finance
6	15872	Conference call	conference, over, credit, call, from, bill, sell	C-5	Finance	C-3	Business, Finance	C-2, C-1	Business, Finance, Time and Date
7	15516	Gas daily pricing	work, employee, gas, project, file, news, services	C-3	Business, Resource	C-3	Business	C-4, C-3	Resource, Information Technology
8	2409	Meeting, program	set, meeting, contact, email, program	C-2	Information Technology	C-2	Information Technology	C-1	Time and Date
9	18835	Message, agreement	etrade, daily, billion, forward, message, file, agreement	C-2	Information Technology	C-2	Information Technology	C-3, C-2	Information Technology, Business, Finance

In this work, multi labeling of emails using classification based on clustering approach has been proposed and the effect of entropy and davies-bouldin index parameters over the cluster quality has been studied. The effect of sample size as well as number of clusters over performance parameter is also explored in this work. The effect of entropy over sample size and number of cluster shows that K-means and agglomerative clustering approach performs approximately similar when the cluster size is small but NMF does not perform well compared to other two clustering techniques. Whereas the effect of DB-Index over sample size and number of cluster shows that it performs in a consistently decreasing manner which indicates a good quality of cluster formation in which NMF gives good results.

CONCLUSION

An algorithm ML-EC^2 has been designed for creating categorized similar emails using text clustering and classification approach. The purpose of this algorithm is to create clusters of emails belonging to various categories. This algorithm is designed for managing emails into pre-defined category using cluster labels. ML-EC^2 is based on text classification using clustering approach. The goal of this work is to manage the emails systematically when there is a vague idea of email topic. Proper categorical term identification is required for effective categorization of emails. Hence a methodology for identifying categorical terms and clustering emails based on LDA has presented in this proposed work. ML-EC^2 is presented with pseudo-code where a single email cluster can be mapped to more than one email category i.e. a form of many-to-one mapping. It uses K-means, Agglomerative and NMF based clustering algorithm to form email cluster. Through experiments the effect of clusters and sample size (number of emails) over the performance parameter entropy and DB-Index has been studied. While carrying out experiments, it has been observed that when increasing the cluster size and sample size DB index decreases almost at a steady rate and performs consistently good whereas entropy keeps on increasing. NMF gives better result in terms of DB-index but performs least when entropy is considered. Whereas K-means performs better with smaller size of cluster and sample size when entropy is considered.

OPEN RESEARCH

In future, the work presented can be utilized with bio-inspired algorithm for classifying emails. The categorical terms has been identified by using latent Dirichlet allocation in this paper. Further some re-estimation technique based on topic modeling approach can be explored to identify effective categorical terms.

REFERENCES

Aloui, A., & Neji, M. (2010). Automatic classification and response of E-mails. *Int. J. Digital Soc., 1*(1).

Alsmadi, I., & Alhami, I. (2015). Clustering and classification of email contents. *Journal of King Saud University-Computer and Information Sciences*, *27*(1), 46–57. doi:10.1016/j.jksuci.2014.03.014

Armentano, M. G., & Amandi, A. A. (2014). Enhancing the experience of users regarding the email classification task using labels. *Knowledge-Based Systems*, *71*, 227–237. doi:10.1016/j.knosys.2014.08.007

Bekkerman, R. (2004). *Automatic categorization of email into folders: Benchmark experiments on Enron and SRI corpora*. Academic Press.

Bermejo, P., Gámez, J. A., & Puerta, J. M. (2011). Improving the performance of Naive Bayes multinomial in e-mail foldering by introducing distribution-based balance of datasets. *Expert Systems with Applications*, *38*(3), 2072–2080. doi:10.1016/j.eswa.2010.07.146

Berry, M. W. (2004). Survey of text mining. *Computer Review*, *45*(9), 548.

Beseiso, M., Ahmad, A. R., & Ismail, R. (2012). A new architecture for email knowledge extraction. *International Journal of Web & Semantic Technology*, *3*(3), 1–10. doi:10.5121/ijwest.2012.3301

Carmona-Cejudo, J. M., Baena-García, M., Campo-Avila, J., Morales-Bueno, R., Gama, J., & Bifet, A. (2011, October). Using gnusmail to compare data stream mining methods for on-line email classification. In *Proceedings of the Second Workshop on Applications of Pattern Analysis* (pp. 12-18). Academic Press.

Dehghani, M., Shakery, A., & Mirian, M. S. (2016). Alecsa: Attentive Learning for Email Categorization using Structural Aspects. *Knowledge-Based Systems*, *98*, 44–54. doi:10.1016/j.knosys.2015.12.013

Gomez, J. C., Boiy, E., & Moens, M. F. (2012). Highly discriminative statistical features for email classification. *Knowledge and Information Systems*, *31*(1), 23–53. doi:10.1007/s10115-011-0403-7

Guan, D., & Yuan, W. (2013). A survey of mislabeled training data detection techniques for pattern classification. *IETE Technical Review*, *30*(6), 524–530. doi:10.4103/0256-4602.125689

He, B., Li, Z., & Yang, N. (2014, September). A novel approach for email clustering based on semantics. In *Web Information System and Application Conference (WISA), 2014 11th* (pp. 269-272). IEEE. doi:10.1109/WISA.2014.56

Huang, Y., & Mitchell, T. M. (2008). Exploring hierarchical user feedback in email clustering. *Email '08: Proceedings of the Workshop on Enhanced Messaging-AAAI*, (pp. 36-41). Academic Press.

Islam, M. R., Zhou, W., Guo, M., & Xiang, Y. (2009). An innovative analyser for multi-classifier e-mail classification based on grey list analysis. *Journal of Network and Computer Applications*, *32*(2), 357–366. doi:10.1016/j.jnca.2008.02.023

Klimt, B., & Yang, Y. (2004, September). The enron corpus: A new dataset for email classification research. In *European Conference on Machine Learning* (pp. 217-226). Springer. doi:10.1007/978-3-540-30115-8_22

Koprinska, I., Poon, J., Clark, J., & Chan, J. (2007). Learning to classify e-mail. *Information Sciences*, *177*(10), 2167–2187. doi:10.1016/j.ins.2006.12.005

Kovács, F., Legány, C., & Babos, A. (2005, November). Cluster validity measurement techniques. *6th International symposium of hungarian researchers on computational intelligence*.

Kulkarni, A., & Pedersen, T. (2005, December). *Name Discrimination and Email Clustering using Unsupervised Clustering and Labeling of Similar Contexts*. IICAI.

Li, H., Shen, D., Zhang, B., Chen, Z., & Yang, Q. (2006, December). Adding semantics to email clustering. In *Data Mining, 2006. ICDM'06. Sixth International Conference on* (pp. 938-942). IEEE. doi:10.1109/ICDM.2006.16

Manco, G., Masciari, E., & Tagarelli, A. (2008). Mining categories for emails via clustering and pattern discovery. *Journal of Intelligent Information Systems*, *30*(2), 153–181. doi:10.1007/s10844-006-0024-x

McCallum, A. K. (2002). *Mallet: A machine learning for language toolkit*. Retrieved from http://mallet.cs.umass.edu/

Park, S., & An, D. U. (2010). Automatic e-mail classification using dynamic category hierarchy and semantic features. *IETE Technical Review*, *27*(6), 478–492. doi:10.4103/0256-4602.67153

Sakurai, S., & Suyama, A. (2005). An e-mail analysis method based on text mining techniques. *Applied Soft Computing*, *6*(1), 62–71. doi:10.1016/j.asoc.2004.10.007

Sappelli, M., Pasi, G., Verberne, S., de Boer, M., & Kraaij, W. (2016). Assessing e-mail intent and tasks in e-mail messages. *Information Sciences*, *358*, 1–17. doi:10.1016/j.ins.2016.03.002

Schmid, M. R., Iqbal, F., & Fung, B. C. (2015). E-mail authorship attribution using customized associative classification. *Digital Investigation*, *14*, S116–S126. doi:10.1016/j.diin.2015.05.012

Sharaff, A., & Nagwani, N. K. (2016). Email thread identification using latent Dirichlet allocation and non-negative matrix factorization based clustering techniques. *Journal of Information Science*, *42*(2), 200–212. doi:10.1177/0165551515587854

Sharaff, A., & Nagwani, N. K. (2019). Identifying Categorical Terms Based on Latent Dirichlet Allocation for Email Categorization. In *Emerging Technologies in Data Mining and Information Security* (pp. 431–437). Singapore: Springer. doi:10.1007/978-981-13-1498-8_38

Shazmeen, S. F., & Gyani, J. (2011, April). A novel approach for clustering e-mail users using pattern matching. In *Electronics Computer Technology (ICECT), 2011 3rd International Conference on* (Vol. 6, pp. 205-209). IEEE. doi:10.1109/ICECTECH.2011.5942082

Špitalský, V., & Grendár, M. (2013). OPTICS-based clustering of emails represented by quantitative profiles. In *Distributed Computing and Artificial Intelligence* (pp. 53–60). Cham: Springer. doi:10.1007/978-3-319-00551-5_7

Sun, L., Liu, B. Q., Wang, B. X., & Wang, X. L. (2010, July). A clustering based fast detection algorithm for large scale duplicate emails. In *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on* (Vol. 6, pp. 3270-3274). IEEE. doi:10.1109/ICMLC.2010.5580695

Wang, Y., Liu, Y., Feng, L., & Zhu, X. (2015). Novel feature selection method based on harmony search for email classification. *Knowledge-Based Systems*, *73*, 311–323. doi:10.1016/j.knosys.2014.10.013

Wei, X., & Croft, W. B. (2006, August). LDA-based document models for ad-hoc retrieval. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 178-185). ACM.

Xiang, Y. (2009). Managing email overload with an automatic nonparametric clustering system. *The Journal of Supercomputing*, *48*(3), 227–242. doi:10.1007/s11227-008-0216-y

Zareapoor, M., Shamsolmoali, P., & Alam, M. A. (2015). Highly Discriminative Features for Phishing Email Classification by SVD. In *Information Systems Design and Intelligent Applications* (pp. 649–656). New Delhi: Springer.

Zhang, J. J., Fang, M., Wu, J. Q., & Li, X. (2016). Robust label compression for multi-label classification. *Knowledge-Based Systems*, *107*, 32–42. doi:10.1016/j.knosys.2016.05.051

Aakanksha Sharaff has completed her graduation in Computer Science and Engineering in 2010 from Government Engineering College, Bilaspur (C.G.). She has completed her post graduation Master of Technology in 2012 in Computer Science & Engineering (Specialization- Software Engineering) from National Institute of Technology, Rourkela and completed Ph.D. degree in Computer Science & Engineering in 2017 from National Institute of Technology Raipur, India. Her area of interest is Software Engineering, Data Mining, Text Mining and Information Retrieval. She is currently working as an Assistant Professor at NIT Raipur, India.

Naresh Kumar Nagwani has completed his graduation in Computer Science and Engineering in 2001 from G. G. Central University, Bilaspur. He completed his post-graduation Master of Technology in Information Technology from ABV-Indian Institute of Information Technology, Gwalior in 2005 and completed the Ph.D. in Computer Science and Engineering in 2013 from National Institute of Technology Raipur, India. His employment experience includes Software Developer and Team Lead at Persistent Systems Limited and presently Associate Professor at National Institute of Technology Raipur, India. He has published more than 40 research papers in various journals and conferences in the field of is data mining, text mining, mining software repositories, and information retrieval.