Web spam detection using fuzzy clustering

J.Shyam Jegadeesh
Bethlahem Institute of Engineering, Karungal, Tamil Nadu, India
shyamjegadeesh@gmail.com

P.Libin Jacob
Bethlahem Institute of Engineering, Karungal, Tamil Nadu, India
plibinjacob@gmail.com

J.John Spencer
Bethlahem Institute of Engineering, Karungal, Tamil Nadu, India
john.spencer@hotmail.com

C.Stanly DevaKumar
Bethlahem Institute of Engineering, Karungal, TamilNadu, India
stanlydevakumar@yahoo.co.in

Abstract— Internet is the most widespread medium to express our views and ideas and a lucrative platform for delivering the products. For this intention, search engine plays a key role. The information or data about the web pages are stored in an index database of the search engine for use in later queries. Web spam refers to a host of techniques to challenge the ranking algorithms of web search engines and cause them to rank their web pages higher or for some other beneficial purpose. Usually, the web spam is irritating the web surfers and makes disruption. It ruins the quality of the web search engine.
So, in this paper, we presented an efficient clustering method to detect the spam web pages effectively and accurately. Also, we employed various validation measures to validate our research work by using the clustering methods. The comparisons between the obtained charts and the validation results clearly explain that the research work we presented produces the better result.

Keywords— *Web Spam, Clustering, Spam, Search Engine*

*****

I. INTRODUCTION

Internet is the most widespread medium to express our views and ideas. We can communicate our notions with the customers, visitors and other businessmen for marketing and promoting their products. Really, after the invention of the Internet, the lives of all people of the world has been changed a lot [19]. Now everyone is depending on the internet to browse or seek anything they want. It has become a very important part of our essential daily life.

The internet browsers have been increasing every day, due to their personal reasons and interests. All people are using this online world as their business platform which is one of the most popular medium recognized among the internet visitors and users. So, it becomes necessary for every business to market itself in this online world [14]. Hence, business can get benefit by using internet as a medium to market the products and services.

Therefore with the help of internet, all businesses are making huge initiatives to build a large customer base. For getting huge number of visitors daily, it is essential for the web page to be indexed in the world known and popular search engines like Google, MSN, Bing, and Yahoo etc. Therefore, the internet surfers can search any information they need through these search engines. So, for getting indexed and catalogued in search engines, it is necessary to make initiatives like SEO that stands for Search Engine Optimization. Search Engine Optimization is a method to improve the online ranking of their web pages and their visibility of business by making its website to be appeared in the top results of search engines.

The business people can hire an SEO [19] firm to make their website user friendly. If their website is user friendly, then it will increase the probabilities of being indexed and catalogued in search engines. Therefore SEO must be included in a website. Search engines were known as the brightest stars in the Internet investing frenzy that occurred in the late 1990s. Several firms entered the market amazingly, receiving record gains during their initial public offerings. Some have used their public search engine, but many were caught up in dot-com bubble, a speculation-driven market explosion that ranked high in 1999 and ended in 2001.

Around 2000, search engine developed by Google rose to fame. The company achieved better results by inventing the new algorithm such as PageRank [19]. It is an iterative

algorithm which ranks the web pages based on their number and PageRank of other web sites and their corresponding pages that link there, on the evidence that good or needed pages are linked to more than others. Nowadays, all the web pages are following the concepts based upon the concepts of Google's PageRank algorithm.

Usually, the popular search engine operates in the following order

1) Web crawling.
2) Indexing.
3) Searching.

Usually, the search engines store the information about the web pages [14], which they retrieve from the page's HTML title tag, meta tags with attributes such as keyword, description and page contents. These pages are retrieved by a Web crawler (sometimes also known as a spider) — an automated Web browser which can't be seen, that follows each and every link on the site. But the robots.txt must be excluded. Analysis are done separately to determine how the contents of each page be indexed. The information or data about the web pages are stored in an index database for use in later queries. A query from a user can be a single phrase or sentence. The index helps to find information relating to the query as quickly as possible.

Spamming is the use of electronic messaging systems to send unlimited bulk messages (**spam**) [26] [27], especially advertising, extensively. The most recognized one is e-mail spam. Some of the other spams are instant messaging spam, wiki spam, web search engine spam, mobile phone messaging spam etc. Spamming remains economically feasible because the advertisers have no operating costs beyond the management of their mailing lists and it is very difficult to find out the senders.

In this paper, search engine spam is one of the vulnerable spam which employs a number of methods such as repeating unrelated words or phrases, to manipulate the relevance or prominence of resources indexed in a manner inconsistent with the purpose of the indexing system. Usually, search engines use various algorithms to determine the relevancy ranking. Search engines employ these techniques to determine whether the search term appears in the body or URL of the web page. It checks for the instances of the search engine spam and remove the mistrusted pages from the indexes [14]. Based upon the user complaints or false search term matches, people from the search engine optimization team can quickly block the results listing web pages. Using the unethical methods, they tried to make the website to list higher in the page rank and to originate top in the search engine listing. Usually, it is classified into two categories: content spam and link spam.

Content spam [27] is the technique which alters the logical views which the search engine has control over the web pages. They all aim at variants of the vector space model for information retrieval on text collections. Link spam is the technique that can be defined as links between pages that are presented for general reasons other than specific purpose. Link spam employs the link-based ranking algorithm which gives higher ranking for the website therefore it reaches top which surfing in the search engine.

Really, the effective spam detection method is a very tedious job. However, while detecting the search engine spam, we have to detect only the spam page and we do not mistakenly consider the legitimate page as spam. It is very useful if we detect the spam page as early as possible before the query processing. Therefore, we can effectively utilize our resources by properly crawling, pre-processing and indexing the original pages which is not a spam.

This paper clearly defines two goals: first, it explains a clear cut idea about the various types of content based spam, their ideas and principles. Secondly, it draws the algorithm for content based spam detection. Usually, search engine spam is annoying the web surfers and it creates a major problem to search engine. Hence, mostly search engines try to combat these types of spam. Combating search engine spam consists of identifying the search engine spam with high probability downgrade the spam during the page ranking, no crawling was performed on it, and no indexing was done on that page.

Our research paper is organized in the following manner. In Section II, we discussed about the Taxonomy of Spam Techniques. In Section III, we described about the goal of our research work. In Section IV, we explained our experimental results and discussions. Finally, we ended our paper with conclusion in Section V.

II. TAXONOMY OF SPAM

A. Content Spam

The web spam is a very critical problem for search engine optimization. The most important and widespread form of the web spam is the content spam. It is so widespread because of the fact that search engines use information retrieval models based on a page content to rank web pages, such as a vector space model [28], or statistical language models [27]. Thus, spammers analyze the weaknesses of these models and exploit them. The various techniques of content spamming are

Title Spamming: To retrieve the data or information from the internet, the title field is very important. But to achieve

high rank for their web page while searching, spammers have a clear motivation to overstuff it.

Body Spamming: The web page is constructed using HTML tags. Usually, the content is displayed in the body tag. So, in this type of spam, the body of a webpage is modified. This is the most common and cheapest form of content spam because the spammers apply various strategies to achieve high rank. For instance, if a spammer wants to achieve a high rank for their webpage with only limited set of queries, they can use the redundant strategies by over-stuffing the body of a page with phrases or keywords that appear in the set of queries. At the same time, the site owners use a lot of random keywords at once, if their aim is to cover as many queries as possible. Also, they color the content in the same manner that of background color which could be recognized only by search engine. Hence, they get high rank while searching.

Meta-Tags Spamming: Usually, the meta-tags were placed under the head tag. This meta-tag [19] plays an important role in the document description. All the search engine looks for the meta-tags and analyzes them carefully, if found. Hence, the spammer places the spam content in this field. Because of the immense spamming, nowadays the software engineers who design search engines give a less priority to this tag or even ignore it fully.

Anchor Text Spamming: The anchor tag is used to create link to other web page or other document. Hence, in 1994, the anchor text is used for web ranking, and but nowadays, spammers added this valuable strategy for their usage. Spammers create hyperlinks with the desired anchor text which is unrelated to linking page content in order to get high rank.

URL Spamming: Some search engines consider the tiny or smallest part of URL of a webpage as a zone. From the set of queries they type, spammers create a URL for webpage from the keywords. For instance, if the query is like "livecricketscore", then the spammers can create a URL "live-score.com/live-cricket/score-cricket.html" for high rank.

B. Link Spam

Link spam is defined as links between pages that are present for reasons other than merit. Link spam is working based upon the link-based ranking algorithms, which gives websites higher rankings than the linked highly ranked websites. Some of them are described below

Link Farm: A link farm is any group of web sites in the internet that links to every other site in the group. In general terms of graph theory, a link farm is a cluster. Although some link farms can be created manually, most farms are created automatically with the help of programs and services. A link farm is a form of spamming the search engine's index (sometimes called spamdexing) [19]. Other link exchange systems are designed in such a way that it allows individual websites to exchange links with other websites and it won't be considered a form of spamdexing. Search engines usually require methods to verify the relevancy of webpage. A known method is to examine the one-way links that belongs to the relevant websites. The new links should not be bewildered with being mentioned links in the cluster, as the latter requires reciprocal return links, which often makes the back link in vain. This is due to oscillation, causing confusions over the vendor site and the promoting site.

*Cookie Stuffing***:** In World Wide Web, Cookie stuffing is a marketing technique used to generate unlawful associate sales. Here, the user browses the product website. Without being aware of the user, the spamdexing from the target website introduces the third party cookie which belongs to different website. When the user made purchases by following the links through the target website, the commission is paid to the target website.

Page hijacking: Page hijacking is accomplished by creating a copy of a popular website which is similar to the original web site. They pretend that they are the real one to a web crawler, but it redirects web browsers to separate or malicious websites. Usually for high ranking purpose, spammers can use this technique. But, some web crawlers detect duplicate page while indexing web pages. A single URL will be indexed, if the web pages contain same content. But in certain cases, legitimate web pages can also be edited by external advertisers via XSS and redirected to promoting web site.

III. THE RESEARCH WORK

The aim of our research presented in this paper is to develop an algorithm and efficiently classify the web page into spam and legitimate web page and to find how efficiently the web pages are classified into their respective categories. The web-page can be represented in terms of its features. The features in the domain considered in this paper will be words. Words are represented with discrete values based on statistics of the presence or absence of words.

In this paper, the main purpose of the classifier is used to classify a web page to be either spam or non-spam accurately. Each cluster [5][23] is abstracted using one or more representatives. These representative points are the results of efficient clustering algorithms like K-Means [6] and Fuzzy C-means[4].

Nearest neighbor classifier and K-nearest neighbor classifier [25] are the two classifiers which are used and assigns each unlabeled web page to its nearest labeled cluster.

Based upon the working process, the function of the

developed algorithm is as follows,

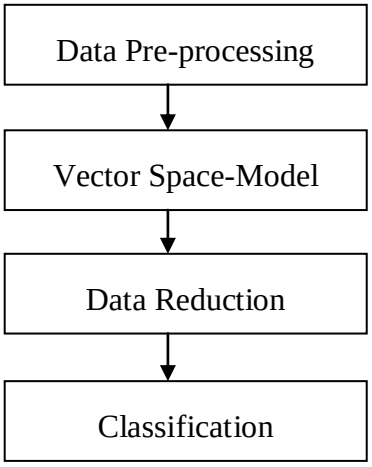

Fig 1 Steps used for spam filtering

- To remove of words which is having the length<3.
- To remove of stop words.
- A Single word must be maintained, if multiple words occurred.
- Convert the webpage into vector form.
- Cluster similar training webpages for data reduction.
- To classify the test pattern.

This paper deals with the possible improvements gained from differing classifiers used for a specific task. Basic classification algorithms as well as clustering are introduced. Common evaluation measures are used. The methodology used in the research work considered in this paper for the spam filtering is summarized under the following 4 steps shown in the Fig. 1.

C. Data Preprocessing

The basic step that has to done in data pre-processing is truncating and trimming. Truncating is the process of removal of words that are lesser in length (i.e., words with length less than specified value), frequently occurring words and special symbols. For grammatical reasons, web pages are going to use different forms of a word, such as function, functions, and functional [9]. Trimming reduces derivation related forms of a word to a common base form. The Trimmer is based on the idea that the suffixes in the English language (approximately 1200) are mostly made up of a combination of smaller and simpler suffixes. This Trimmer is a linear step process.

Usually, it works on the principle by removing the common morphological endings from words in English. This process is known as the term normalization process that must be done before setting up information retrieval systems. It works based on number of vowels followed by a consonant character in the trim (measure), must be greater than one for the rule to be applied [3]. Based upon these principles, the web pages can be easily classified into spam & non-spam webpage more efficiently**.** Hence, truncating and trimming are done to reduce the vocabulary size which helps information retrieval and classification purposes.

D. Building the Vocabulary

This section explains how to build a vocabulary using a space model. To start with, assign to each term in the webpage, a weight for that term [2]. The simplest approach is to assign the weight to be equal to the number of occurrences of the term *s* in webpage *w*. This weighting scheme is referred to as 'term frequency' and is denoted $sf_{s,w}$ with the subscripts denoting the term and the webpage in order [10].

For the webpage *w*, the set of weights (determined by the sf weighting function above or indeed any weighting function that maps the number of occurrences of *s* in *w* to a positive real value) may be viewed as a vector, with one component for each distinct term. The vector view only retains information on the number of occurrences [11]. Usually, raw term frequency suffers from a critical problem, i.e., all terms are considered equally important when it comes to assessing relevancy on a query. Certain terms have little or no distinguishing power in determining relevance [2].

Therefore, scale down the term weights of terms with high collection frequency, which is defined to be the total number of occurrences of a term in the collection.

The idea would be to reduce the sf weight of a term by a factor that grows with its collection frequency. Instead, use the document frequency wf_s defined to be the number of webpages in the collection that contain a term *s*. Denoting the total number of webpages in a collection by *N*, the inverse webpage frequency (iwf), inverse webpage frequency of a term *s* is given by Eq. (1) as [12]

$$iwf_s = log\left[\frac{n}{wf_s}\right] \quad (1)$$

Now, combine the above expressions for term frequency and inverse webpage frequency, to produce a composite weight for each term in each webpage. The $\mathrm{sf-iwf}$ weighting scheme assigns to term *s* a weight in webpage *w* given by Eq. (2) as

$$\mathrm{sf-iwf} = \mathrm{sf}_{s,w} \times \mathrm{iwf}_s \quad (2)$$

The model thus built shows the representation of each webpage described by their attributes [13]. Each tuple is assumed to belong to a prior-defined class. Thus, the webpages with their weighted terms can be represented in the form of a table given in Table 1.

E. Data Reduction and clustering

Data reduction includes data clustering that concerns how to group a set of objects based on their similarity of attributes in the vector space. Clustering methods are applied on the tuples which produce 2 different training sets, one belonging to spam and the other belonging to non-spam.

TABLE 1 Web pages with their weighted terms

	Term1	Term2	…	Term N
Web Page1	W_1	W_2		W_N
Web Page 2	W_1	W_2		W_N
Web Page3	W_1	W_2		W_N
:	:	:		:
Web Page N	W_1	W_2		W_N

The cluster representatives will now belong to 2 different classes. The class that each tuple (webpage) belongs which is given by one of the attributes of the tuple often called the class label attributes [14].

i) Data Clustering

Clustering [4] is the process of partitioning or dividing a set of patterns (data) into clusters. Clustering is a type of classification imposed on finite set of data patterns. The relationship between patterns is represented in a proximity matrix in which the tuples represent 'n' webpages and fields correspond to the terms given as dimensions. If objects are categorized as patterns, the proximity measure can be Euclidean distance [6][7] between the data items. Unless a meaningful measure of distance, between a pair of data items are established, no meaningful cluster analysis is possible. Clustering can be applied to a lot of research areas like decision making, data mining, text mining, etc. It also helps in detecting outliers and to examine small size clusters.

The proximity matrix serves as a useful input to the clustering algorithm. It represents a cluster of *p* patterns by *c* points. Typically, $c < p$ leading to data compression, can use centroids. This would help in prototype selection for efficient classification. In this paper, the pattern clustering activity involves the following steps such as

- Representation of Pattern (optionally including feature extraction and/or selection),
- Definition of a pattern proximity measure,
- Clustering,
- Data abstraction (if needed), and
- Evaluation of output (if needed).

The Fig. 2 shows the first three of the above mentioned 5 steps, including a feedback path where the output of the grouping step could affect feature extraction and similarity computations [16]. The individual scalar components x_i of a pattern x are called features. Feature selection is the process of identifying the most effective subset of the original features.

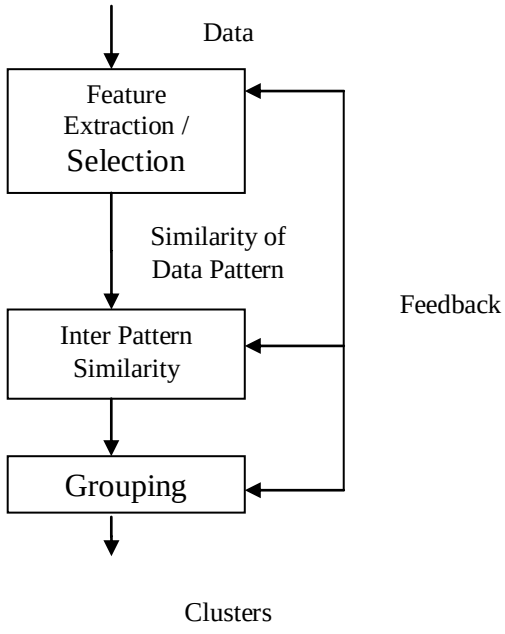

Fig 2 Initial Phases of Clustering Process

Feature extraction is the use of one or more transformations of the input features to produce new significant features. Either or both of these techniques can be used to obtain an appropriate set of features. Pattern proximity is usually calculated by a distance function defined on pairs of patterns, such as the Euclidean distance between patterns.

The clustering algorithms are broadly classified into hard (a partition of the data into groups) or soft (where each pattern has a variable degree of membership in each of the output clusters). In the work considered in this paper, both methods have been used [18].Some of the clustering algorithms popularly used are:

ii) K-Means

The K-means algorithm takes the input cluster parameter *c*, and partitions a set of *n* patterns into *c* clusters so that the resulting intra cluster similarity is high but the inter-cluster similarity is low. The K-means algorithm is classified as a combined method, because it requires that all the data should be previously known. However, there are variants of the k-means clustering process, which gets around this limitation. The most important step in K-means algorithm is to choose the proper initial. The time complexity of the k-means algorithm is O(*ncl*), where *n* is the number of patterns, *c* is the number of clusters, and *l* is the number of iterations [19]. The other

variant of K-means algorithm is the single pass K-means algorithm. Generally K-means algorithm takes more number of iterations to converge. For handling large data sets with K-means algorithm needs buffering strategy which takes single pass over the data set.

Buff is the size of buffer and *C* is the cluster representatives which are representing the means of cluster. Initially, data of size *Buff* is placed into the buffer. K-means algorithm is applied on the buffered data. The cluster representatives are stored into the memory. The remaining data is discarded from the memory. Again data is loaded into memory from disk. But, the K-means algorithm [4] is performed on this new data with previous cluster representatives. This process repeats until the whole data is clustered. It takes less computational effort as compared to the normal K-means algorithm. But, the K-means algorithm suffers from initial guess of centroids, value of *C*, lack of scalability, capacity to handle numerical attributes and resulting clusters can be unbalanced. The K-means algorithm is explained below:

Step 1 : Select *k* initial centers.

Step 2 : repeat {

- ❖ assign every data point to the nearest cluster based on the distance measure between the data point and the center of the cluster.
- ❖ calculate the new centers of the k clusters

} until(the convergence criterion is met).

Generally, the k-means algorithm has the following important properties such as

- ❖ It is efficient in processing large data sets.
- ❖ It often terminates at a local optimum.
- ❖ The clusters have spherical shapes.
- ❖ It is sensitive to noise.

iii) Fuzzy C-means

The fuzzy clustering [6][9] allows overlapping of clusters. Fuzzy cluster analysis takes into account memberships of data points to cluster in [0,1]. The membership degrees provides finer of detail of data model. Beside from this, membership degree also shows how definitely a data point should belong to a cluster. Fuzzy clustering provides solution spaces in the form of fuzzy partitions for a set of samples $X = \{x_1, x_2, x_3 \ldots., x_n\}$.The fuzzy partition matrix or membership matrix is represented by a $c\ x\ n$ matrix U= $(U_{ki}) = \{U_1, U_2, U_3 \ldots., U_n\}$.The cluster assignment U_{ki} is the membership degree of a feature vector xk to cluster c such that $U_{ki} = C_k C_i \in [0,1]$, since memberships to clusters are fuzzy, fuzzy clustering methods associate a fuzzy label vector to each feature vector xk to states its memberships to *c* clusters:

$$U_k = (U_{1k}, U_{2k}, U_{3k}, \ldots., U_{ck})^T \quad (3)$$

Most fuzzy clustering algorithms [9] are objective function based: They determine an optimal classification by minimizing the proximity measure. Each cluster is represented by a cluster prototype which consists of a cluster center. The cluster center is an instantiation of the data points used to describe the domain which is used to divide the dataset. The centroid of the cluster is computed by using the clustering algorithm. It may or may not appear in the dataset. The degrees of membership to which a given data point belongs to the different clusters are computed from the distances of the data point to the cluster centers with respect to the size of the cluster as stated by the additional prototype information. The degree of membership to the cluster is higher if the data point lies close to the center of a cluster. Hence the problem to divide a dataset $X = \{x_1, x_2, x_3 \ldots., x_n\}$ into c clusters can be stated as the task to minimize the distances of the data points to the cluster center such that the degree of membership is maximized.

Several fuzzy clustering algorithms can be distinguished depending on the additional size and shape information contained in the cluster prototypes, the way in which the distances are determined, and the restrictions that are placed on the membership degrees.

Fuzzy C means is an algorithm using fuzzy clustering concept. It describes the fuzzy classification for the pixels by computing fuzzy membership value. FCM is defined to be optimal when it minimizes the objective function:

$$J(U, C) = \sum_{i=1}^{C} \sum_{j=1}^{N} \mathrm{U\ ki} \left\| X_j - \mu_i \right\|^2 \quad (4)$$

Where the parameter m is called the fuzzifier or weighting exponent and d_{ik} represents the distance measure in feature space using a feature vector x_j and a cluster center C_i in feature space using Euclidean norm between is the Euclidean distance. The necessary conditions for minimizing J(U,C) with respect to U and C and then equating it to zero:

$$U_{ki} = \frac{1}{\sum_{k=1}^{c} \left[\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right]^{\frac{2}{m-1}}} \quad (5)$$

$$C_k = \frac{\sum_{i=1}^{N} U_{ik} X_i}{\sum_{i=1}^{N} U_{ik}} \quad (6)$$

The FCM algorithm proceeds by iterating the above two conditions until the optimal value is reached. Each data point will be associated with a membership value for each cluster center of their clusters are assigned. High membership values and low membership values are assigned to data points far from the cluster center. The fuzzy C-means algorithm is explained below

Step 1) Choose the initial class centers c_i of the data patterns with $1 \leq i \leq c$, where c is the total number of classes.

Step 2) Find the pattern g_x from the observation.

Step 3) Compute the square Euclidean distance measure between the data pattern g_x and the class center c_i for all classes as follows:

$$d_{x,i}^2 = \|g_x - c_{i_i}\|^2, for\ 1 \leq x \leq n, 1 \leq i \leq c \quad (7)$$

Step 4) Calculate the membership matrix $U_{x,i}$ given as

$$u_{x,i} = \frac{[\frac{1}{(d_{x,i})^2}]^{1/(m-1)}}{\sum_{i=1}^{c}[\frac{1}{(d_{x,i})^2}]^{1/(m-1)}}, for g_x \neq c_x \quad (8)$$

and

$$u_{x,i} = \begin{cases} 1, & if x = i \\ 0, & otherwise \end{cases} \quad (9)$$

Step 5) Update the class centers as

$$c_i = \frac{1}{\sum_{x=1}^{n}(u_{x,i})^m g_x} \sum_{x=1}^{n}(u_{x,i})^m g_x \quad (10)$$

Step 6) Check $\Delta = \max[|U^{(t+1)} - U^{(t)}|]$.

If $\Delta > \epsilon$ then go to *Step 4)*. Otherwise, stop.

The initial value m, called the exponential weight is set to be two, and it will alleviate the noise effect when computing the class centers. The larger the value of m (m>1) is, the more the sensitivity of the noise will be.

F. Classification of Data Pattern

Classifiers are used to predict the class label of the new webpage which is unlabeled [7]. The classifiers are explained below

i) NNC

The NNC (Nearest NeighbourClassifier) assigns to a test pattern a class label of its closest neighbour. If there are n patterns $X_1, X_2, X_3, \ldots . X_N$ each of dimension d, and each pattern is associated with a class c, and if we have a test pattern P, then

$$if\ d(P, X_k) = \min\{(d(P, X_i)\}\ where\ i = 1,2, \ldots n$$

To compare the distances of a given test pattern with other patterns, the nearest neighbour classifier uses the Euclidean distance method which is given by [11]

$$\sqrt{(x_2 - x_1)^2 - (y_2 - y_1)^2} \quad (11)$$

Pattern P is assigned to the class associated with X_k. The algorithm for NNC is explained below

Step 1 :Calculate the centroids from the clustering module.

Step 2 : Calculate the distance between the data pattern and each centroid.

Step 3 :The data pattern is assigned to the class associated with the least distance from the distances calculated.

ii) K-NNC

The K-NNC (K-Nearest Neighbour Classifier) [24] is a variant of the nearest neighbour classifier in which k nearest neighbours are found but in the case of nearest neighbour classifier, only one nearest neighbour is calculated. The nearest neighbours are calculated by using the Euclidean distance.

The value chosen for *c* is crucial and with the right value of c, the classification accuracy will be better compared to that of nearest neighbour classifier. In order to reduce the error, the value of *k* can be larger for large data sets. Choosing *c* can be done experimentally, where a number of patterns taken out from the training set can be classified using the remaining training patterns for different values of *c* and *c* can be chosen the value which gives the least error in classification [2].

When the training patterns are noisy, this method will reduce the error in classification. The closest pattern of the test pattern may belong to another class. Patterns are more likely to be classified correctly when the numbers of neighbours are obtained and the majority class label is considered. The algorithm for the K-NNC is explained below

Step 1 : Based upon each centroid, the distance of test data is calculated.

Step 2 : Find out the "K" nearest neighbors from the above calculated distances.

Step 3 : Usually, the test data has majority of the minimum distances. Then, classify the test data corresponding to the class label.

G. Evaluation of Measures

This is done in 2 steps such as classifier accuracy and alternative to the measure of the accuracy.

i) Classifier Accuracy

Estimating classifier accuracy is important in that it allows one to evaluate how accurately a given classifier will label the test pattern. It can be calculated using the formula discussed below. The data set used for training and testing is the 'ling spam corpus'. Each of the 10 sub-directories contains spam and legitimate messages, one message in each file. The total number of spam messages is 481 and that of legitimate messages are 2412.

ii) Alternatives to accuracy measure

A classifier is trained to classify e-mails as non-spam and spam mails [6]. An accuracy of 87 % may make the classifier accurate, but what if only 15-20 % of the training samples are actually "spam"? Clearly an accuracy of 87 % may not be acceptable if the classifier could be correctly labeling only the "non-spam" samples. Instead, we would like to be able to access how well the classifier can recognize "spam" samples how well it can recognize "non-spam" samples. The sensitivity and specificity validation measures can be used for this purpose. In addition, we may use precision to access the percentage of samples labeled as "spam" that actually are "spam" samples. The evaluation measures which are used in approach for testing process in our research work could be defined as follows [4]:

True Positive (TP) : This states the number of spam web pages correctly classified as spam.

True Negative (TN) : This states the number of non-spam web pages correctly classified as non-spam.

False Positive (FP) : This states the number spam web pages classified as non-spam.

False Negative (FN) : This states the number of non-spam web pages classified as spam.

TABLE 2. The different measure used to classify spam and non-spam messages

Measure	Formula	Meaning
Precision	$\frac{TP}{TP+FP}$	The percentage of positive predictions that are correct
Recall / Sensitivity	$\frac{TP}{TP+FN}$	The percentage of positive labeled instances that were predicted as positive.
Specificity	$\frac{TN}{TN+FP}$	The percentage of negative labeled instances that were predicted as negative.
Accuracy	$\frac{TP+TN}{TP+TN+FN+FP}$	The percentage of predictions that are correct.

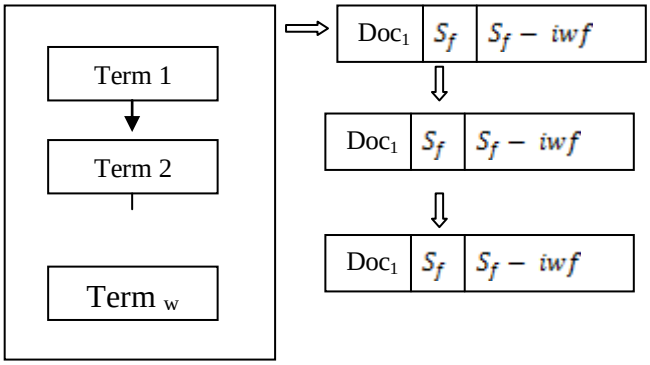

Fig 3 Vector Space model

Note that the evaluation is done based upon the above 4 parameters. It is summarized in the form of a table in table 2.

iii) Vector space model

Due to the large number of features (terms) in the training set, memory requirements will be more. Arrays cannot be used to store the features as this leads to memory problems so we use a linked list to implement the storage of features and the $S_f - iwf$ calculation [5]. As the training set contains large number of web pages, the web pages are also implemented in the linked list format as shown in Fig. 3.

FCM is used to cluster large number of data. Using Euclidean distance, it forms the clusters from the data pattern in the web document. The data flow diagram used for the design of the algorithm for efficient spam webpage classification is shown in the Fig. 4 along with the inputs & outputs. The general description of the inputs and the outputs shown in the Fig. 4 could be further explained as follows which involves a 5 step procedure [7].

The algorithm for the vector space model is explained below

Step 1 : Find whether the word is already present in the vocabulary list.

Step 2 : If not found, insert this word into a new node and

update the webpage number and frequency in the corresponding node.

Step 3 : If the word is already found, and if it is appearing for the first time in the webpage, then create a new node with the webpage number and it's corresponding frequency.

Step 4 : Otherwise if the word is appearing again in the same webpage then increment the frequency.

Step 5 : Calculate the inverse webpage frequency (iwf) for each term(word) by the formula $iwf = log\ (N/wf_s)$, where N is the total number of webpage and wf_s is the number of webpages that the term has occurred in.

Step 6 : Calculate the $S_f - iwf$ of each word in each webpage by the formula

$$S_f - iwf = \text{Frequency} * \text{iwf}$$

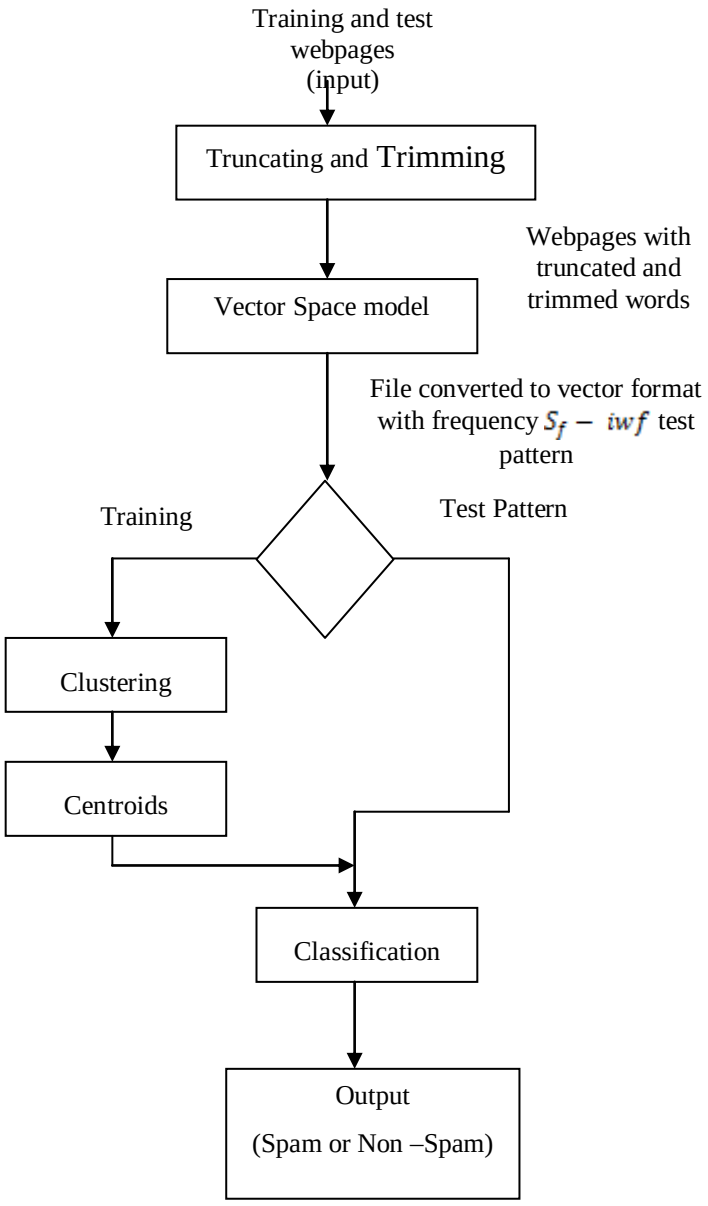

Fig. 4 : Data flow diagram (DFD) of the designed system or the proposed model

IV. RESULTS AND DISCUSSION

The coding was done in PHP; after the code was run, various performance measures such as the precision, recall, specificity & the accuracy, etc. were observed. The results are shown in the Figs. 5 to 8 respectively.

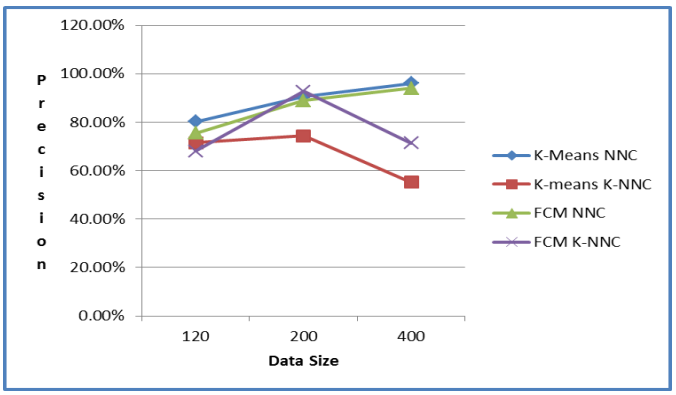

Fig 5 Chart for Precision Vs Data Size

i) Precision

The percentages of positive predictions that are correct are high for nearest neighbour classifiers. The precision table in table 3 and the following graph in Fig. 5 shows that for large data sets, FCM with NNC and K-means with NNC has an optimal value.

TABLE 3 Quantitative results of Precision

Data Size	K-Means NNC	K-means K-NNC	FCM NNC	FCM K-NNC
120	80.2	71.5	75.5	68
200	90.5	74.4	88.9	92.7
400	96.1	55.2	94	71.3

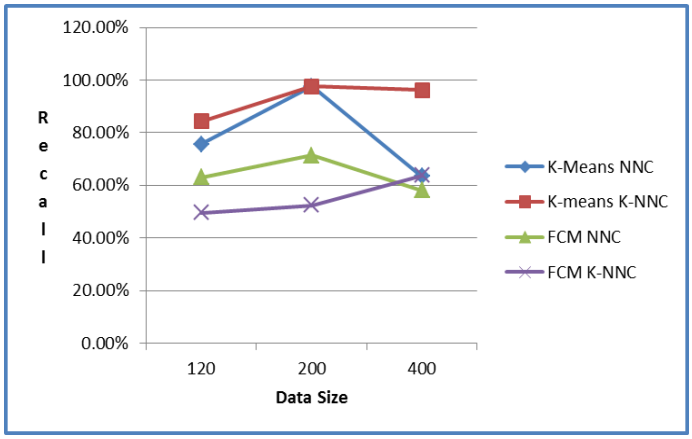

Fig 6 Chart for Recall Vs Data Size

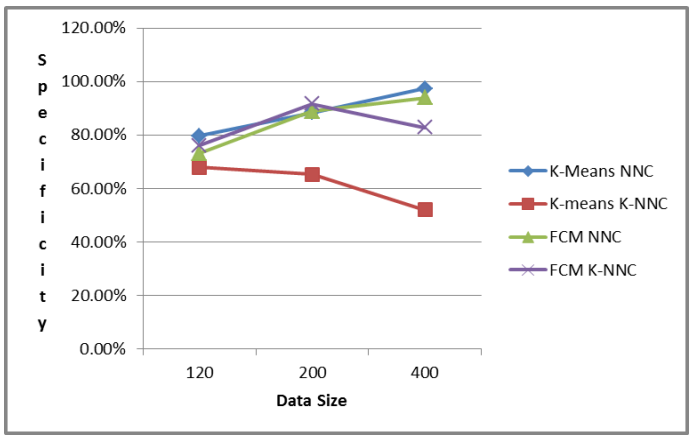

Fig 7 Plot of measure of specificity vs. data size

ii) Recall

The percentage of positive labeled instances that predicted positive are high for the combination of K-means algorithm with K-NNC as the classifier and the percentage increases as the data set size increases. FCM does not work well for smaller data sets. The recall values can be visualized from the following table in table 4 which indicates that for large data sets, FCM with K-NNC has a high value, which can also be observed from the Fig. 6.

TABLE 4 Quantitative results of Recall

Data Size	K-means	K-means	FCM NNC	FCM K-NNC
120	75.67%	84.30%	63.10%	49.60%
200	97.60%	97.60%	71.40%	52.30%
400	63.40%	96.20%	58.10%	63.70%

TABLE5 Quantitative results of Specificity

Data Size	K-Means NNC	K-means K-NNC	FCM NNC	FCM K-NNC
120	79.60%	67.85%	73.20%	76%
200	88.40%	65.20%	88.90%	91.70%
400	97.40%	52.00%	94%	82.80%

iii) Specificity

The percentages of negative labeled instances that are predicted as negative are high for the combination using NNC as the classifier. The specificity values for large data sets as seen from the following table in table 5 are optimal for FCM with K-NNC combination, which can also be observed from the Fig. 7.

iv) Accuracy

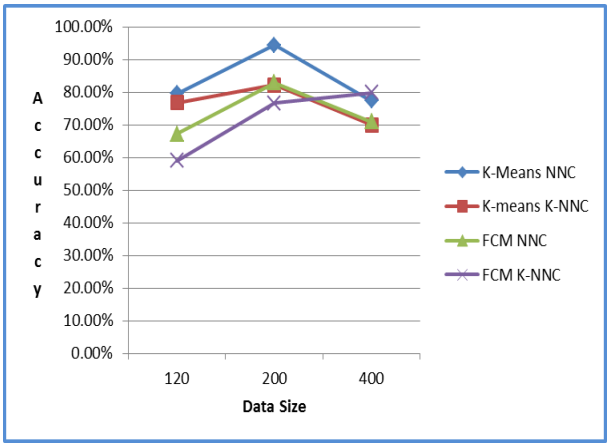

Fig. 8 : Plot of measure of accuracy vs. data size

Accuracy for FCM with K-NNC has an optimal value as the data set increases; also K-means works well for smaller data set. It can be visualized from the graph in Fig. 8 that conditions being checked hold good for large data and FCM with K-NNC is the best combination if the data set increases.

TABLE 6 Quantitative results of Accuracy

Data Size	K-Means NNC	K-means K-NNC	FCM NNC	FCM K-NNC
120	79.60%	76.85%	67.20%	59%
200	94.40%	82.20%	82.90%	76.70%
400	77.40%	70.00%	71%	79.80%

TABLE 7. Comparison of FCM and K-means with dataset

	FCM	**K-means**
Time	Faster	Slower

Sensitivity to input pattern of dataset	Yes	No
Cluster Quality (center location, number of data point in a cluster, radii of clusters)	More accurate	Less accurate
Demand for memory	Less	More

It can be seen that FCM with K-NNC is more accurate for large data, which can be observed from the quantitative results shown in the table 6.

V. CONCLUSION

In this paper, the proposed clustering method can be able to identify the spam webpage. The proposed technique includes the distance between all of the attributes of webpage. Different performance and validation measures such as the precision, recall, specificity and the accuracy were observed. K-means clustering algorithm works well for smaller data sets. FCM with K-NNC is the best combination as it works better with large data sets. In FCM clustering, decisions made without scanning the whole data and FCM utilizes local information (each clustering decision is made without scanning all data points). FCM is a better clustering algorithm requiring a single scan of the entire data set thus saving time. The work presented in this paper can be further extended and can be tested with different clustering algorithms and varying size of large data sets.

REFERENCES

1) SudiptoGuha, Adam Meyerson, Nina Mishra, Rajeev Motwani and Liadan O'Callaghan, "Clustering Data Streams," IEEE Trans.s on Knowledge & Data Engg., 2003.
2) D.L. Pham, J.L. Prince, An adaptive fuzzy C-means algorithm for image segmentation in the presence of intensity inhomogeneities, Pattern Recogni-tion Letters 20 (1) (1999) 57–68.
3) M.N. Ahmed, S.M. Yamany, N. Mohamed, A.A. Farag, A modified fuzzy C-mean algorithm for bias field estimation and segmentation of MRI data, IEEE Transactions on Medical Imaging 21 (3) (2002) 193–199.
4) Miin-Shen Yang, Hsu-Shen Tsai, A Gaussian kernel-based fuzzyc -means algorithm with a spatial bias correction, Pattern Recognition Letters 29 (2008) 1713–1725.
5) Al-Daoud, M. B., & Roberts, S. A. (1996). New methods for the initialization of clusters. Pattern Recognition Letters, 17, 451–455.
6) A.K. Jain, Data clustering: 50 years beyond K-means, Pattern Recognition Letters 31 (2010) 651–666.
7) Masulli F, Schenone A. A fuzzy clustering based segmenta-tion system as support to diagnosis in medical imaging. Artif Intell Med 1999;16(2):129 —147.
8) H. Frigui, R. Krishnapuram, A robust competitive clustering algorithm with applications in computer vision, IEEE Transactions on Pattern Analysis and Machine Intelligence21 (1999) 450–465.
9) Jianzhong Wang, Jun Kong, Yinghua Lu, Miao Qi, Baoxue Zhang, A modified FCM algorithm for MRI brain image segmentationusing both local and non-local spatial constraints, Computerized Medical Imaging and Graphics 32 (2008) 685–698
10) Enrico Blanzieri and Anton Bryl, "A Survey of Learning-Based Techniques of Email Spam Filtering," Conference on Email and Anti-Spam., 2008.
11) Jain A.K., M.N. Murthy and P.J. Flynn, "Data Clustering : A Review,"ACM Computing Surveys., 1999.
12) Tian Zhang, Raghu Ramakrishnan, MironLivny, "BIRCH: An Efficient Data Clustering Method For Very Large Databases," Technical Report, Computer Sciences Dept., Univ. of Wisconsin-Madison, 1996.
13) Porter. M, "An algorithm for suffix stripping", Proc. Automated library Information systems, pp. 130-137, 1980.
14) Manning C.D., P. Raghavan, H. Schütze, "Introduction to Information Retrieval", Cambridge University Press, 2008.
15) Richard O. Duda, Peter E. Hart, David G. Stork, "Pattern Classification", Wiley-IntersciencePubs., 2nd Edn., Oct. 26 2000.
16) http://www.informationretrieval.org/
17) http://www.aueb.gr/users/ion/publications.html
18) http://www.cl.cam.ac.uk/users/bwm23/
19) http://www.wikipedia.org
20) Jiawei Han and MichelineKamber, "Data Mining Concepts and Techniques", Second Edn.
21) Ajay Gupta and R. Sekar, "An Approach for Detecting Self-Propagating Email Using Anomaly Detection", Springer Berlin / Heidelberg, Vol. 2820/2003.
22) AnaghaKulkarni and Ted Pedersen, "Name Discrimination and Email Clustering using Unsupervised Clustering and Labeling of Similar Contexts", 2nd Indian International Conference on Artificial Intelligence (IICAI-05), pp. 703-722, 2005.
23) Bryan Klimt and Yiming Yang, "The Enron Corpus: A New Dataset for Email Classification Research", European Conference on Machine Learning, Pisa, Italy, 2004.
24) Sahami M., S. Dumais, D. Heckerman, E. Horvitz, "A Bayesian approach to filtering junk e-mail". AAAI'98

Workshop on Learning for Text Categorization, http://robotics.stanford.edu/users/sahami/papers-dir/spam.pdf, 1998.

25) Sculley D., Gordon V. Cormack, "Filtering Email Spam in the Presence of Noisy User Feedback", CEAS 2008: Proc. of the Fifth Conference on Email and Anti-Spam. Aug., 2008.
26) Dave DeBarr, Harry Wechsler, "Spam Detection using Clustering, Random Forests, and Active Learning", CEAS 2009 – Sixth Conference on Email and Anti-Spam, Mountain View, California, USA, July 16-17, 2009.
27) Manning, C.D., Raghavan, P., and Schutze, H., "Scoring, Term Weighting, and the Vector Space Model", Introduction to Information Retrieval, Cambridge University Press, Cambridge, England, pp. 109-133, 2008.
28) Naresh Kumar Nagwani and Ashok Bhansali, "An Object Oriented Email Clustering Model Using Weighted Similarities between Emails Attributes", International Journal of Research and Reviews in Computer Science (IJRRCS), Vol. 1, No. 2, pp. 1-6. Jun. 2010.