Improving Web Search Results by removing Outliers using Data Mining Techniques

Mennatollah M. Mahmoud
Information Systems department
Faculty of Commerce & Business Administration
Helwan University, Cairo, Egypt

Shaimaa Salama
Information Systems department
Faculty of Computers & Information
Helwan University, Cairo, Egypt
Faculty of Computers and Information Technology
King Abduaziz University, Jeddah, KSA

Doaa S. Elzanfaly
Information Systems department
Faculty of Computers & Information
Helwan University, Cairo, Egypt
Faculty of Informatics & Computer Science
British University in Egypt

ABSTRACT

Many users access the web seeking for information. They put their query or question in search engines that may returns irrelevant pages or results compared to users' needs. This research paper proposes a model to remove outliers from the search results. The proposed model is based on association rules, modified Naïve Bayes algorithm and clustering techniques. The Naïve Bayes algorithm is modified to help removing outliers from the search results. The proposed model has been evaluated using the Sum of Squared Errors (SSE), silhouette coefficient and entropy evaluation measures against the standard k-medoids algorithm. Experimental results show that the proposed model outperforms the standard k-medoids clustering algorithm in removing the search outliers.

Keywords

Information Retrieval (IR), Web mining, Association rules (AR), Classification, Clustering, Outlier detection.

1. INTRODUCTION

The internet is an important source of information; it produces every day a huge number of electronic documents and of all types such as web pages, research papers, e-mails, audio and video documents [1].

The presence of the plenty of information, in addition to the dynamic and heterogeneous nature of the Web, makes information retrieval a hard process for the user. Search engines and Meta-Search engines were developed to help users to quickly and easily find their information need [2, 3]. Sometimes the user writes ambiguous query, so that the search engine cannot determine the context or category of the user needs. For this reason the query results may contain irrelevant web pages compared to the users need as well as redundant web pages [2]. This problem occurs because the search engine performs exact matching between the query terms and the keywords that are contained in each web page and presents the results to the user [3]. The search results returned using any searching paradigm appear as long lists of URLs, which are very hard to filter or get the relevant pages [2, 3].

To avoid this problem, it is important to organize and filter the resulted documents according to user query. These have brought challenges for automatic organization of these electronic documents effectively and efficiently by using data mining [4].

Document clustering is a data mining technique that categorizes documents into different groups known as clusters. Documents within each cluster are similar to each other and share some common properties according to defined similarity measure and are dissimilar to other clusters. It is a type of data clustering [4].

Document clustering algorithms are categorized into two categories; hard and soft clustering categories. In hard clustering category, each point is assigned to exactly one cluster. Examples of hard clustering are k-means, k-medoids, etc. In soft clustering category, each point can belong to multiple clusters according to its membership degree. Fuzzy c-means is an example of the soft clustering category [4]. Although both k-medoids and k-means belong to the same, hard clustering category, yet they differ in many respects. K-means is based on Euclidean distance while k-medoids can be based on any distance measure. K-means is sensitive to outliers while k-medoids isn't sensitive to outliers; this is because k-means depends on the mean value of the data points in determining the cluster center which is called centroid. On the other hand k-medoids depends on the median value to determine the cluster center. The cluster center is called medoid. Depending on the median for determining the cluster center as well as using cosine distance as a similarity measure make k-medoids outperforms k-means in document clustering [5].

In this paper, a model that will save the users' time and effort for finding the relevant document is developed by removing the web document outliers and organizing the search results. The model uses Apriori algorithm to find and get the related documents. An algorithm based on Naïve Bayes is developed to remove the outliers from the resulted documents. The model also used K-medoids technique to categorize the relevant documents. A comparison is made between the proposed model and the traditional k-medoids algorithm. The experimental results show that the proposed model outperforms the traditional k-medoids clustering algorithm.

The organization of the paper is as follows. The concise review of related researches is presented in Section 2. The proposed model is described in Section 3. The extensive analysis of the proposed model using different parameters as well as the comparison with other clustering algorithm is given in section 4. Section 5 concludes the paper and proposes the future work

2. LITERATURE REVIEW

A lot of researches have targeted the topic of improving query results on the web.

2.1 Improving clustering algorithms

Some of the researches work on improving clustering algorithms to better categorizes the web search results**.**

Moe Moe Zaw and Ei Ei Mon in [6] presented a PSO-based Cuckoo Search Clustering Algorithm to combine the strengths of Cuckoo Search and Particle Swarm. The proposed method includes two phases: preprocessing phase and clustering phase.

The documents to be clustered are collected first then preprocessing is applied to them through tokenization, stop words removal and features representation in Vector Space Model. In the clustering phase, the distance from the centroids to the other documents is measured by Cosine distance measure. The documents to the nearest center will go to this cluster. For next center selection, the old center is moved to the new center by PSO based Cuckoo Solutions. The algorithm will finally produce the user-defined number of document clusters. A limitation of the proposed algorithm is that the value of k, the number of desired clusters, is still required to be given as an input.

K A Abdul Nazeer et al in [7] improved k-means algorithm in a novel heuristic method to determine the initial centroids of the cluster. It ensures that the centroids are chosen in accordance with the distribution of data. The proposed algorithm can deal with multidimensional data values. Unlike the original k-means algorithm in which the initial centroids are selected randomly, the proposed algorithm determines the initial centroids in a more meaningful way, in accordance with the distribution of data. Consequently, the algorithm converges much faster than the original k-means algorithm. A limitation of the proposed algorithm is that the value of k, the number of desired clusters, is still required to be given as an input.

A.S.N.Chakravarthy et al in [8] suggested fast greedy k-means algorithm that enables users to find the relevant documents more easily, it overcomes the drawbacks of k-means algorithm, that is it works very slow and it is not applicable for large databases, and it is very much accurate and efficient. The fast greedy k-means algorithm has a limitation when the algorithm is used for large number of data points.

2.2 Enhancing similarity measures

Other researches depended on enhancing the document similarity measures which produce better clusters.

M. Yasodha and P. Ponmuthuramalingam in [9] proposed concept-based mining model which is used to improve the text clustering quality. It exploits the semantic structure of the sentences in documents. The proposed model analyzes terms on the sentence, document, and corpus levels. It can effectively discriminate between non important terms with respect to sentence semantics and terms which hold the concepts that represent the sentence meaning. The similarity between documents is calculated based on a new concept-based similarity measure. A limitation of this model is that it doesn’t consider the importance of the word to the document. It doesn’t give weights for each word.

P. Vigneshvaran et al in [10] proposed an empirical method to estimate the semantic similarity using HBase. In the proposed method, the key document is compared with the source documents in the document corpus and the similarity measure has been calculated. Then process pre-processing work, notation removal and stop word removal. Now the document contains only the keywords in it. These keywords are tokenized. Repeated keywords are forms a cluster. Those important keywords are transferred to HBase database. There exists checking of semantic similarity by means of listing synonym of keywords. Calculate numbers of words that match with the source document keyword. Identify the document that contains matched keywords. Based on the importance level calculate MSE (Mean Square Error). Then calculate MSD (Mean Square Deviation) for key document where constants values taken from MSE. If MSD <= MSE then the document is similar otherwise it is not similar.

Hyungsul Kim et al in [11] proposed to use frames to represent documents, which can capture the semantics of documents and represent documents in a comprehensive and concise way. They use an information network approach that treats the corpus as a gigantic semantic information network. Then, a link-based similarity function called SynRank is proposed to capture the similarity between frames in an iterative way. A limitation of the proposed approach is that it doesn’t name each type and argument role [12].

2.3 Detecting outliers

Other researches proposed models to detect the outlier web pages which help the users to get relevant documents.

S. Sathya Bama et al in [13] proposed a mathematical model based on correlation method. Their model depends on calculating the correlation between the document pairs. In their model, they apply preprocessing on the documents that result from the search. Preprocessing step includes stop words removal, stemming and tokenization. Then term frequency is calculated for all the terms in the documents. Then they score each term according to their frequency in the document. The term having highest frequency should be ranked 1. Then the correlation coefficient is used to compare documents. If the correlation value equals to 1, then the document is redundant and should be removed. This method improves the reliability of the search engine for removing the redundant pages. This model has the limitation of handling only the redundant documents. It couldn’t achieve the required accuracy of search engines by removing irrelevant pages to the query.

W.R. Wan Zulkifeli et al in [14] depended on a traditional weighting technique TF-IDF from Information Retrieval in the construction of their algorithm. The proposed algorithm used full word matching and an organized domain dictionary which is indexed based on the word length. They assume the presence of a dictionary for a specific category. The full word frequency profile for the web page is created. The web pages are weighted based on their frequencies. A penalty is given to the word that is found in the document but not in the domain dictionary. This word tends to be dissimilar to the document. While those found in the dictionary increase the possibility of the similarity between the document and the dictionary. This model has a limitation that exact word matching doesn’t consider the context or the semantic of the documents

3. PROPOSED MODEL

The proposed model is developed to filter the search results generated from any search engine by removing outliers and focusing only on the relevant documents and results which save users’ time and effort.

As shown in Figure 1, the model consists of four main stages: Document preprocessing, frequent keywords and related documents extraction, outlier filtration, and document clustering.

3.1 Document preprocessing

Document preprocessing step is critical because it extracts meaningful keywords. The next steps depend on this critical step. Document preprocessing includes stop word removal, stemming and tokenization. Stop word removal is the process of removing the common unwanted and meaningless words such as a, an, the, is, was, etc. that affect the performance of mining the document. Stemming is the process of removing the word derivatives and return the word to its root. Tokenization is the process of converting the whole document text into separate words called tokens [15].

3.2 Frequent keywords and related documents extraction

This is the initial pruning step where frequent keywords are generated from the whole dataset. All the documents that share the same set of the extracted frequent keywords are related to each other. First, the most frequent key words are extracted from the whole dataset using minimum support threshold. Then the documents that contain these frequent keywords are identified. Documents that match are considered to be related to each other, because they share common keywords. Apriori algorithm is used to generate the frequent key words.

There is no standard value for the minimum support. According to the chosen minimum support value the accuracy level of the resulted documents will be achieved. So it is important to choose the minimum support threshold. The output of this step is the documents that match the frequent keyword.

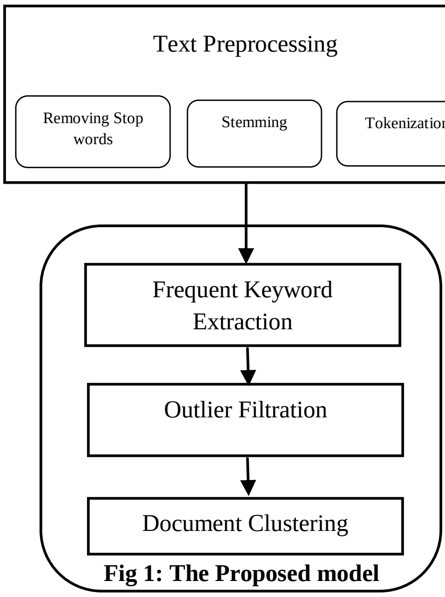

Fig 1: The Proposed model

3.3 Outlier filtration

There are other documents in the dataset that are related to the extracted documents, in the previous step, by context. These documents aren't extracted because they don't contain exactly the same frequent keywords. In this step they will be extracted. An outlier filtration algorithm based on Naïve Bayes is developed to extract the related documents from the remaining documents. Figure 2 shows the algorithm.

First, the related documents as well as the remaining documents are represented as binary vectors. The related documents are put into a corpus. If the remaining document contains 70% or more of the words of the related documents, then consider this document to be relevant and add it to the relevant documents corpus. Else, create a new class called "Pending class" and put the remaining documents in it. Repeat the steps on the pending class documents until no changes happened.

3.4 Document clustering

In this step, clustering technique is applied on the related documents. In other words, the documents that result from the previous step only clustering technique is applied to them. The aim of this step is to form categories or groups of the resulted documents. The documents that share the same topic are put in the same cluster. This helps the users to save their time and effort to find and get the needed documents.

The clustering method used in this step is K-Medoids with cosine distance as the distance matrix.

```
Input:
  • ID: Input Documents
  • PD: Pending Documents
Output: Relevant documents and filtered outliers

Method:
    Corpus of Related Documents CRD = ID.
    Repeat
      N= total number of docs in CRD
      Loop (ID=1, ID<=N, ID++)
          Split ID to feature sets (words), where the
          attributes are unique words W.
          Represent documents ID using the vector
          space model binary representation.
          Assign values to these attributes to be
          either 0 or 1, where 0 is the absence of the
          word in a given document and 1 is the
          presence of the word in the document.
      End loop
      If (W in PD>= 70%)
            CRD=CRD+PD
      Else
          Pending Class PC =0
          PC = PC +PD
      End if
  Until no changes occur.
```

Fig 2: The Outlier Filtration Algorithm

4. EXPERIMENTL RESULTS AND EVALUATION

The proposed model is performed on a 396 documents dataset with approximately 5000 keywords from UCI Machine learning repository [16]. It is also applied on another dataset that contains 100 documents from Google search engine for the topic "data mining in education" with nearly 2000 keywords.

Regarding to the above mentioned datasets, the number of the generated clusters is determined using the Elbow method as it is a very straightforward method used to generate the most suitable number of clusters.

The idea of the elbow method is to run k-medoids clustering on the dataset for specified values of k and for each value of k calculates the sum of squared errors (SSE).Then plot a line chart of the SSE for each value of k. Choose the value of k that is in the location of the elbow (knee) in the plot [17]. The number of the generated clusters for the first dataset (396 documents) is 3 clusters. The number of the generated clusters for the second dataset (100 documents) is 2 clusters.

The used tools are: KNIME for frequent keyword generation and clustering. Python program for the outlier filtration algorithm with Spyder tool.

The experimental results as well as the evaluation measures will be discussed in this section.

4.1 Evaluation measures

The Sum of Squared Errors (SSE) measure, the Silhouette coefficient measure and the Entropy measure are used to evaluate the final clusters after applying the proposed model. The SSE is chosen to be one of the evaluation measures, because it measures the variation within the cluster. It measures the distance between each data point and the cluster center or medoid. The SSE is described in equation 1 [18].

$$E = \sum_{i=0}^{k} \sum_{p \in c} dist(p, c\, i)^2 \qquad (1)$$

Where E is the sum of the squared error for all objects in the data set; p is the point in space representing a given object; and ci is the centroid of cluster Ci. The smaller the SSE the better the performance or the cohesion between the data points in the cluster is high.

The silhouette coefficient is used to measure the internal cluster cohesion and the external cluster separation. The silhouette coefficient is described in equation 2 [19]

$$Si = (bi - ai)/max\,(ai, bi) \qquad (2)$$

Where Si is the silhouette coefficient for a given data point, bi is the minimum average distances between the data point and the points in other clusters not containing the data point and ai is the distance between the data point and other points in the same cluster. An overall measure of the goodness of the clustering process can be obtained by calculating all the average silhouette coefficient of all data points. The larger the silhouette coefficient, the better the performance.

The entropy is an external clustering evaluation measure. It depends on comparing clusters using its labels. Each cluster acts as information source or the base to measure the quality of the other cluster. The entropy is described in equation 3 [20].

$$Entropy = \sum_{r=1}^{k} \frac{n_r}{n}\left(-\frac{1}{log\, q}\sum_{i=1}^{q}\frac{n_r^i}{n_r} log \frac{n_r^i}{n_r}\right) \quad (3)$$

Where k is the number of clusters, n_r is the size of cluster r and n is the total number of data points, q is the number of classes. The smaller the Entropy value means better clustering.

4.2 Experimental results

To evaluate the proposed model, its performance has been compared to the standard k-medoids clustering algorithm in terms of accuracy and the ability to get relevant results using the SSE, Silhouette coefficient and the Entropy.

The organization of this section is as follows. Tables 1-7 prove the choice of 70% as the outlier filtration percentage (for both datasets) as well as a preliminary comparison between the proposed model parameters combination and the traditional k-medoids. Table 8 describes a comparison between the proposed model and k-medoids in terms of the evaluation measures as well as figures that demonstrate such evaluation. Tables 9&10 describe the percentage of the related and outlier documents in the filtered documents regarding each cluster in both datasets.

Tables 1-3 evaluate the proposed model against the traditional k-medoids with different parameters. They describe the number of outlier documents, number of related documents, time in terms of the iteration numbers and the SSE with different minimum supports and combination of different percentages of the outlier filtering algorithm. As it can be noticed in the three tables, the most suitable parameters for further analysis, in terms of the iteration number, SSE, the number of the related documents and the number of outlier documents, are 3 as a minimum support and an outlier 70% as a filtration percentage.

Table 1. Minimum support = 2

%	Related	outliers	Iteration #	SSE
60%	364	32	3	4021
70%	335	61	3	3710
80%	304	92	2	3368
90%	257	139	1	3896
K-medoids				**4388**

Table 2. Minimum support = 3

%	Related	outliers	Iteration #	SSE
60%	359	37	5	3955
70%	300	96	5	3290
80%	201	195	5	2156
90%	137	259	1	1650
K-medoids				**4388**

Table 3. Minimum support = 4

%	Related	outliers	Iteration #	SSE
60%	355	41	7	3912
70%	267	129	17	2881
K-medoids				**4388**

Tables 4-7 describe the confusion matrix of the proposed model with the four outcomes of the related documents (positive class) and the outlier documents (negative class) as well as the algorithm accuracy. The accuracy of the classifier has been measured using the following equation [21, 22]:

$$Accuracy = \frac{(TP+TN)}{P+N} \qquad (4)$$

Where TP is the true positive documents, TN is the true negative documents, P is the total number of positive documents and N is the total number of negative documents.

In these tables, the chosen outlier filtration algorithm percentage, for further analysis, was 70%. Despite the percentages 70% and 80% have the same accuracy values (85%), 70% is more preferable. That is because only 11% of the related documents are considered as outliers and 84% of the real outliers are filtered.

Table 4. Outlier filtration percentage = 60%

True labels	Predicted labels		Total
	Related	Outlier	
Related	TP = 67	FP = 8	75
Outlier	FN = 9	TN = 16	25
Total	76	24	100
Algorithm accuracy = 83%			

Table 5. Outlier filtration percentage =70%

True labels	Predicted labels		Total
	Related	Outlier	
Related	TP=64	FP=11	75
Outlier	FN=4	TN=21	25
Total	68	32	100
Algorithm accuracy = 85%			

Table 6. Outlier filtration percentage =80%

True labels	Predicted labels		Total
	Related	Outlier	
Related	TP = 61	FP = 14	75
Outlier	FN = 1	TN = 24	25
Total	62	38	100
Algorithm accuracy = 85%			

Table 7. Outlier filtration percentage =90%

True labels	Predicted labels		Total
	Related	**Outlier**	
Related	TP = 58	FP = 17	75
Outlier	FN = 0	TN = 25	25
	58	42	100
Algorithm accuracy = 83%			

In conclusion, the most suitable percentage of the outlier filtration algorithm, described in figure 2, was 70% for any dataset. The minimum support value is given by the user according to the required level of the resulted documents accuracy.

Table 8 summarizes the comparison between the proposed model and k-medoids in terms of the evaluation measures.

Table 8. Comparison between the proposed model and k-medoids

396 documents			
Method	**SSE**	**Silhouette coefficient**	**Entropy**
K-medoids	4388	0.008	1.928
Proposed Model	3290	0.015	1.444
100 documents			
Method	**SSE**	**Silhouette coefficient**	**Entropy**
K-medoids	989	-0.07	1.581
Proposed Model	608	-0.07	0.967

Figure 3 shows the SSE measure for the 2 data sets of the k-medoids and the proposed model. It can be noticed that the cohesion of the proposed model for both data sets is better than that of the traditional k-medoids. That is because in each cluster of the proposed model the distance between each point and the medoid is small compared with that of the traditional k-medoids. So in each cluster of the proposed model the documents are more similar to each other.

In Figure 4, the Silhouette Coefficient for the proposed technique and the traditional k-medoids is measured for 396 and 100 documents. Both the cohesion and the separation of the proposed model are better than the traditional k-medoids. Higher silhouette coefficient means that the distance within each cluster is small and the distance between clusters is large. This means that the documents within one cluster are more similar to each other and dissimilar to other documents.

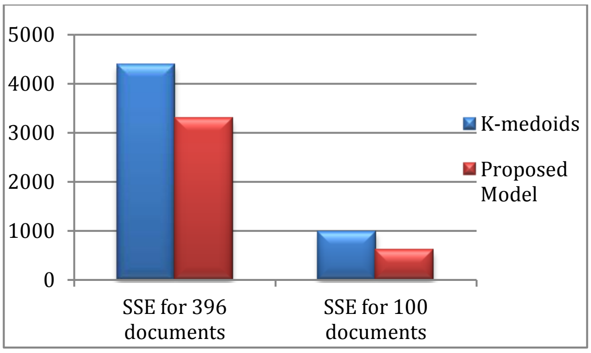

Fig 3: SSE Measurement

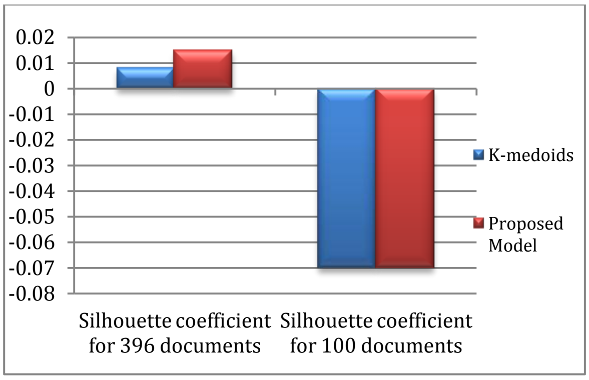

Fig 4: Silhouette Coefficient Measurement

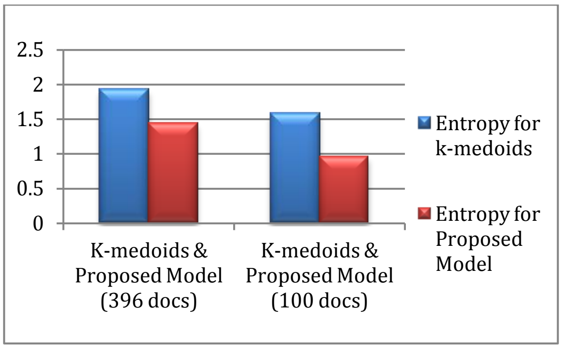

Fig 5: Entropy Measurement

Figure 5 shows the Entropy measure for the 2 data sets of the k-medoids and the proposed model. The proposed model for both datasets is better than the traditional k-medoids. Entropy compares the performance of both methods. Using the k-medoids clusters as a reference class shows that the performance of the proposed model outperforms that of the traditional k-medoids and vice versa.

Tables 9&10 describe the analysis of the filtered outliers regarding each cluster for both datasets. It determines how many related and outlier documents are filtered.

The analysis is made using the cosine distance between each outlier document and each cluster center (Medoid).

Table 9. The percentage of related and outlier documents filtered for 396-document.

Cluster	Outliers Documents	Related Documents	
	Distance = 1	**Distance ≤ 0.5**	**0.5 ≤ Distance ≥ 0.7**
Cluster 1	82%	1%	17%
Cluster 2	72%	28%	-
Cluster 3	70%	3%	27%

Table 10. The percentage of related and outlier documents filtered for 100-document.

Cluster	Outliers Documents	Related Documents
Cluster 1	66%	34%
Cluster 2	66%	34%

5. CONCLUSION AND FUTURE WORK

In this paper, an extensive analysis of the proposed model was conducted. This was based on filtering out the web document outliers and clustering the related documents. The methodology used was the frequent keywords extraction, outlier filtration algorithm based on modified Naiive Bayes algorithm and k-medoids clustering algorithm. The 396 documents from UCI machine learning as well as 100 documents from Google search were chosen to examine the efficiency of the proposed model. The experimental results show that at different parameters of the proposed model, it gives more accuracy than the traditional K-medoids clustering algorithm.

In the future, modifications to the proposed model will be developed to get the highest quality for the resulted documents; an improvement will be made to prevent the related documents from being filtered or treated as outliers, an algorithm will be developed to better extract the most important keywords from the web pages instead of the preprocessing step. Also an algorithm will be developed to rank the document categories based on the relevancy to the search query. An algorithm will be developed to give a brief description for each category.

6. REFERENCES

[1] D. S. Rajput, R. S. Thakur, and G. S. Thakur, "An integrated approach and framework for document clustering using graph based association rule mining", Second International Conference on Soft Computing for Problem Solving, India, 2012, pp. 1421-1437.

[2] R. K. Roul, O. R. Devanand, and S. K. Sahay, "Web document clustering and ranking using tf-idf based apriori approach," International Conference on Advances in Computer Engineering and Applications ICACEA, 2014, pp. 74-78.

[3] N. Negm, M. Amin, P. Elkafrawy, and A. B. M. Salem, "Investigate the performance of document clustering approach based on association rules mining," (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 4, pp. 142-151, 2013.

[4] N. Shah and S. Mahajan, "Document clustering: a detailed review," International Journal of Applied Information Systems (IJAIS), vol. 4, pp. 30-38, 2012.

[5] T. Velmurugan, "Efficiency of k-means and k-medoids algorithms for clustering arbitrary data points, Int. Journal of Computer Technology & Applications, vol. 3, pp. 1758-1764, 2012.

[6] M. M. Zaw and E. E. Mon, "Web document clustering using cuckoo search clustering algorithm based on levy flight", International Journal of Innovation and Applied Studies vol. 4, pp. 182-188, 2013.

[7] K. A. A. Nazeer, S. D. M. Kumar, and M. P. Sebastian, "Enhancing the k-means clustering algorithm by using a O(n logn) heuristic method for finding better initial centroids" , Second International Conference on Emerging Applications of Information Technology (EAIT), Kolkata, India, 2011.

[8] A.S.N.Chakravarthy, Deepthi.S, K.Satyatej, Sk.Nizmi, and S.Sindhura, "Document clustering in web search engine", International Journal of Computer Trends and Technology, vol. 3, pp. 290-293, 2012.

[9] M. Yasodha and P. Ponmuthuramalingam, "An advanced concept-based mining model to enrich text clustering", IJCSI International Journal of Computer Science Issues, vol. 9, pp. 417-422, 2012.

[10] P. Vigneshvaran, E. Jayabalan, and K. Vijaya, "A predominant statistical approach to identify semantic similarity of textual documents", in Informatics and Mobile Engineering (PRIME) International Conference on Pattern Recognition, Salem, India, 2013, pp. 496-499.

[11] H. Kim, X. Ren, Y. Sun, C. Wang, and J. Han, "Semantic frame-based document representation for comparable corpora", IEEE 13th International Conference on Data Mining (ICDM), Dallas, TX, USA, 2013.

[12] S. S. Bama, M. S. I. Ahmed, and A. Saravanan, "A mathematical approach for mining web content outliers using term frequency ranking", Journal of Science and Technology, vol. 8, pp. 1-5, 2015.

[13] L. Huang, T. Cassidy, X. Feng, H. Ji, C. R. Voss, J. Han, and A. Sil, "Liberal event extraction and event schema induction", 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 2016, pp. 258-268.

[14] W. R. W. Zulkifeli, N. Mustapha, and A. Mustapha, "Classic term weighting technique for mining web content outliers", International Conference on Computational Techniques and Artificial Intelligence (ICCTAI'2012), Penang, Malaysia, 2012.

[15] V. Gurusamy and S. Kannan, "Preprocessing techniques for text mining," 2014.

[16] UCI Machine Learning Repository: AAAI 2014 Accepted Papers Data Set. https://archive.ics.uci.edu/ml/datasets/AAAI+2014+Accepted+Papers.

[17] T. M. Kodinariya and P. R. Makwana, "Review on determining number of cluster in k-means clustering", International Journal of Advance Research in Computer Science and Management Studies, vol. 1, pp. 90-95, 2013.

[18] J. Han, M. Kamber, and J. Pei, Cluster analysis: basic concepts and methods in Data mining concepts and techniques, Third Ed. New York, USA: Elsevier Inc.

[19] P.-N. Tan, M. Steinbach, and V. Kumar, Cluster analysis: basic concepts and algorithms in Introduction to data mining. Boston Pearson Addison Wesley, 2006.

[20] A. Rosenberg and J. Hirschberg, "V-Measure: A conditional entropy-based external cluster evaluation measure", Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Prague, 2007, pp. 410–420.

[21] J. Han, M. Kamber, and J. Pei, Classification: basic concepts in Data mining concepts and techniques. New York, USA: Elsevier Inc.

[22] T. R. Patil and S. S. Sherekar, "Performance analysis of naive bayes and j48 classification algorithm for data classification" International Journal of Computer Science and Applications, vol. 6, pp. 256-261, 2013.