Webology (ISSN: 1735-188X)
Volume 18, Number 5, 2021

Boosting Projection Neural Features For Semantic Similar Clustered Documents In Cloud

Mrs B. BalaVinothini[1] , Dr. N.Gnanambigai[2] , Dr. P.Dinadayalan[3]

[1]Research Scholar, Bharathiar University, Coimbatore, India.

[2]Indira Gandhi Arts and Science College Puducherry, India.

[3]Kanchi mamunivar Centre for Postgraduate Studies Puducherry, India.

Abstract— Cloud computing has emerged as real world technology over the Internet. Due to the development of big data with high dimensionality, data storage possibility over cloud has created large scope in recent times. Document clustering is the fundamental topic that turned into an indispensable component in many areas like cloud computing. Document clustering partitions the document into significant classes or groups for retrieving the relevant document. Many researchers used the factorization methods and ontologies for internal and external knowledge based document clustering. However, existing methods failed to provide the semantic feature construction and leads to the information loss while covering all the ideas in documents. In order to address these problems, different document clustering techniques in cloud has been reviewed in this paper. In addition to that Document clustering by Entropy-based Boosting with Projection Neural Feature (EB-PNF) method is presented. The proposed method involves two stages. They are, similar document identification based on semantic similarity score, feature extraction which includes the extraction of both single and multi-label features based on the precision, recall and computational complexity to prove that EB-PNF method produces high-quality clusters comparable to the state-of-the-art methods.

Keywords— Document clustering, Cloud Computing, Entropy, Boosting, Projection Neural Feature, dimensionality, document clustering, semantic feature, ontologies, factorization.

I. INTRODUCTION

Cloud computing has emerged as one of the real world technology that is hugely in use in over internet. With the advent of big data with huge size and dimensionality of data, storage possibility over the cloud has created a large scope of cloud data storage and cloud data management features. Clustering is the process of browsing the document collection or arranging the results returned by the search engine based on user query. Document clustering is used for text mining and information retrieval process in cloud. It is an effective way of identifying the nearest neighbors of the document. Document clustering is carried out to form the documents cluster in hierarchical order automatically. Document clustering involves a text mining model that is widely used in grouping documents that are similar in nature into a single cluster. The document clustering is carried out for

increasing the precision rate or recall rate during the information retrieval Process. With objective of improving the clustering performance, Semi Supervised Concept Factorization (SSCF) was investigated in [1] based on reward and penalty terms. SSCF also ensured that the data points concerning a cluster in the original space still belongs to the same cluster, therefore performing betterment in terms of accuracy and mutual information. However, it lacked in semantic relationship evaluation. To overcome the issue in conventional model, clustering was performed based on the internal and external knowledge. Factorization techniques were used to cluster based on the internal knowledge and construction of ontology [2] for clustering depends on external knowledge. However, factorization techniques lack in semantic feature construction. On the other hand, by applying ontology, certain amount of information loss was said to take place. Automatic clustering of class label is considered to be different from labels specified by humans. Besides, the automatic classification is said to be of lower quality than manual classification. In [4], a big text document clustering model was investigated using term label and semantic feature with the objective of improving the quality of clustering. However, it was not found to be suitable for real life applications. To address this issue, Latent Dirichlet Allocation (LDA) over Map Reduce framework was designed in [5]. With this, modular implementation of multiple documents summarization was said to be ensured.

Yet another document to vector model was designed in [6] to obtain clusters of document with similar contents using Graph Theory and Natural Language Processing. Data clustering for cloud computing was designed in [7] called as, Efficient Stud Krill Herd Clustering (ESKH-C) technique using bacterial foraging algorithm, therefore ensuring optimal solution for real world applications. Co-clustering method provides several advantages over conventional clustering methods. For example, they minimize the initial matrix into a precise representation with an elementary structure and necessitate minimum computation when compared with separate processing of the initial data set and then performing transpose of it. Due to this, these methods are of profound interest to the data mining persons. In [8], two algorithms called, Hard Diagonal Double K Means (DDKM) and Fuzzy Diagonal Double K Means (F-DDKM) were designed with the objective of minimizing the computational complexity involved. However, the method lack term-document corpus-based semantic. To address this issue, Discrimination Information Maximization [9] was used for document clustering. With this, high quality clusters were said to be produced. Yet another semantic approach was designed in [10] by applying lexical chains for extracting semantically related words. As a result, clustering performance was said to be improved. Spectral clustering was applied in [11] by sentence level matrix representation to ensure higher retrieval rate. The rest of the paper is planned as follows. The rest of paper is organized as follows. Section II explains the study and analysis of the existing document clustering techniques in cloud In Section III, document clustering for cloud database storage and managements are introduced for related works. In Section IV, the proposed Entropy-based Boosting with Projection Neural Feature (EB-PNF) method for cloud storage is described in detail with the help of diagram and algorithm. Section V presents the experimental settings with performance evaluation.

II. BACKGROUND WORK

Document clustering is the process of textual documents cluster analysis for document arrangement, extraction and speedy information retrieval. Document clustering was examined for improving the precision or recall in the information retrieval systems through allocating the documents into unseen classes. The main objective is to browse the collection of documents or to organize the search results for displaying frequently in structured or hierarchical way.

Semi-Supervised Concept Factorization (SSCF) was introduced in [1] to enhance the clustering performance with supervisory information. SSCF integrated pairwise limitations into CF as reward and penalty terms. But, SSCF lacked in the semantic relationship assessment. A fuzzy document clustering approach was introduced in [2] for domain-specific ontology with vocabulary explaining the hazards depending on dairy products. However, factorization techniques not provide better semantic feature construction. A big text document clustering model was introduced in [3] with class label and semantic feature based Hadoop. But, big text document clustering model was not appropriate for the real life applications. Latent Dirichlet Allocation (LDA) over Map Reduce framework was introduced in [4] for summarizing the large text collection. But, MapReduce framework was not given importance to facilitate the summary generation from text document collections in many languages. Vector model in [5] combined vector embedding from Natural Language Processing in Graph Theory with dynamics to partition across scales. But, the clustering accuracy was not enhanced using the vector model. Krill herd Efficient Stud Krill Herd—Clustering (ESKH-C) technique was introduced in [6] for solving data clustering problem where swarm of krill converge to the particular position through minimizing the fitness function. But, ESKH-C technique failed to determine the data cluster groups for web applications.

A hard and fuzzy diagonal co-clustering algorithm was constructed in [7] on double K-means for solving document-term co-clustering problems. The designed algorithm comprised better convergence guarantee for improving the co-clustering quality and computational speed on sparse data. However, number of co-clusters knowledge was essential and initiative was not performed for parameter access. An algorithmic framework called CDIM was introduced in [8] to enhance the discrimination information sum provided by the documents. But, CDIM failed to perform document clustering in exact manner as clustering in spaces by corpus-based discrimination not hold large potential. Semantic approach was introduced in [9] through combining the WordNet with lexical chains and allotting the description for generated clusters. But, the lexical chains failed to improve the text clustering performance and not discovered the lexical chains feasibility. The spectral clustering depending on the sentence level matrix representation was carried out in [10] through spectral relaxation for making separable space. K-means algorithm generated k-centroids from Gaussian with the mean and variance. But, clustering time was not minimized. Term frequency based Maximum Resemblance Document Clustering (TMARDC), Correlated Concept based Maximum Resemblance Document Clustering (CCMARDC) and Correlated Concept based Fast Incremental Clustering Algorithm (CCFICA) were introduced in [11] for increasing clustering performance based on the system scalability. But, designed algorithm not performed the concept extraction depending on important phrases and incorporated the semantic relations. A scalable and open source for K-means clustering of correlated multidimensional data termed Centaurus was introduced in [12]. But, clustering accuracy was not improved using k-means clustering technique. In order to resolve limitations of traditional document clustering methods, this paper proposes a novel document clustering method that exploits relative semantic similarity score to identify similarity among documents and single and multi-label document classification by applying Projection-oriented Neural Feature Extraction. In the proposed method, first, semantic similar documents are identified using Semantic Similarity Score based on distance factor and vectorized document.

III. PROPOSED MODEL

The semantic similar document represents the similarity patterns by means of semantic features. Second, single and multi-label document classification is carried out using Projection-oriented neural model. Next, projective nonnegative matrix factorization is applied to the multi-label features to cluster the documents. With the efficient clustering of documents, documents are uploaded into the cloud database. Finally, with entropy-based

retrieval, document retrieval in a user friendly manner is said to take place. The main contribution of paper can be summarized as.

A novel Relative Semantic Cosine Document Similarity algorithm is proposed for obtaining semantic relationship between documents based on distance factor and vectorization, by reducing the time taken for obtaining similar documents. To fasten the algorithm by classifying single and multi-label document classification only considering the projection based on the threshold factor (i.e. labels in the documents), Projection Neural Feature Extraction algorithm is designed. A detailed analysis of the parameter settings (i.e. precision, recall and computational complexity) is given to show the impact they may have on the performance of the proposed Entropy-based Boosting with Projection Neural Feature (EB-PNF) method.

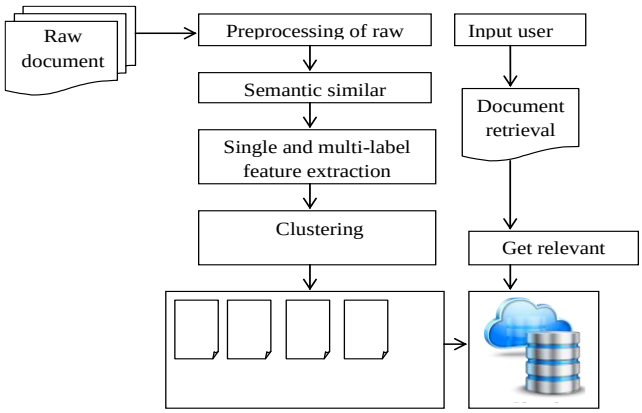

Figure 1 System design of EB-PNF method

This section proposes a document clustering by Entropy-based Boosting with Projection Neural Feature (EB-PNF) method to cluster the documents before uploading into the cloud database based on their respective categories. The EB-PNF method uses the TDT2 [3] dataset which contains nearly thousands of on-topic documents of different categories. The process is described in Figure 1.

As shown in figure 1, the proposed method consists of four major stages. They are preprocessing of raw documents, obtaining semantic similar documents, feature extraction which includes the extraction of both single and multi-label features, clustering the documents before uploading into the cloud database and finally document retrieval using the novel EB-PNF method.

System model

In this work, a directed graph represented as '$G = (V, E)$' is used. Here, 'V' refers to the set of nodes '$V = \{v_1, v_2, \ldots, v_n\}$'. Here each node 'V' represents a unique documents (i.e. image processing) in the entire document set (i.e. {image processing, data mining, cloud computing, etc.}). On the other hand, '$E = \{e_1, e_2, \ldots, e_m\}$' represents the set of edges designed in such a manner that edge 'e' is an ordered pair of documents '(v_i, v_j)'. The edge '(v_i, v_j)' is positioned from 'v_i' to 'v_j'. Besides, 'v_j' is adjacent to 'v_i'. Finally, a set of edges is said to be analogous to a sentence in a document if they link the documents analogous to the keywords in the same order the documents appeared in the entire document set.

Preprocessing of raw documents

Initially the documents are uploaded to the cloud storage where preprocessing is said to be performed. In the preprocessing stage, stop words removal, stemming, keywords extraction and indexing are performed. The first stage in the proposed method preprocesses the sample TDT2 corpus using Tokenization and Part-of-Speech

(PoS) tag. Tokenization executes the task of splitting the text into words. On the other hand, the PoS tags words based on the grammatical context of the word in the sentence. Accordingly it splits the words into nouns, verbs, adjectives, etc. After dropping the common stop-words, words that have been extracted are then stored as preprocessed documents.

Relative Semantic Cosine Document Similarity

With the preprocessed documents, similar documents have to be identified. In this work, a novel Semantic Similarity Score 'SSS' is determined based on two factors, distance and natural language processing techniques. To start with, in this work, similar documents are said to be identified in this work, where semantic similarity over a set of documents is performed. Here, the distance among them is measured on the basis of likeliness of the meaning when compared to similarity based on syntax [2].

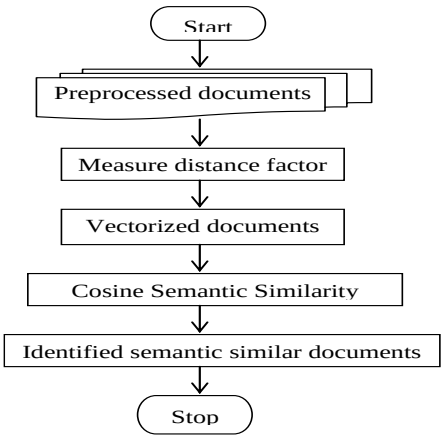

Figure 2 Flow diagram of semantic similar document identification

On the other hand, the natural language processing technique uses the concept of vectorization and vector space to identify similar documents. Figure 2 shows the flow diagram of semantic similar document identification. As shown in the figure, with the preprocessed documents given as input, the objective of the method remains in identifying semantic similar documents at minimum time interval. To start with, semantic similar document identification stage, distance factor is first measured with the assumption that 'R' represents the root. This is mathematically represented as given below.

$$\alpha\left(v_i, v_j\right) = \frac{2d}{L_i + L_j + 2d} \qquad (1)$$

From the above equation (1), 'v_i' and 'v_j', represents the nodes for which the semantic similarity has to be identified. This is evaluated by using the depth 'd', 'L_i' representing the path between 'L_i' and root, 'R', 'L_j' representing the path between 'L_j' and root 'R'. A semantic similar document model is shown in figure 3. For example, two methods are exploited for image processing. They are analog and digital image processing. Analogue image processing is used for obtaining hard copies (i.e. printouts and photographs). On the other hand, digital image processing assists in manipulating digital images with the aid of computers.

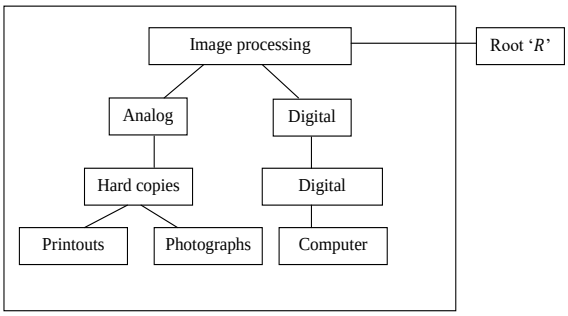

Figure 3 Sample semantic similar document model

Based on the distance factor, preprocessed documents are first vectorized. In this case, before application of vectorization, 'Data a' is referred to as 'Document a', 'Data b' is referred to as 'Document b' and so on. Once vectorization has been performed, 'Document a' refer to 'Vector a', 'Document b' refer to 'Vector b' and so on. In this work, the proposed method works in the assumption that the representation of a preprocessed document 'PD' is a matrix. First of all, we represent the domains of the document as a vector. Let 'D_j' be the vector representation of the 'jth' of the document, then the matrix representing the document will have 'D_j' as the 'jth' column. An illustration of this is represented as given below.

$$PDS_i = \begin{bmatrix} d_{11} & d_{21} & d_{m1} \\ d_{12} & d_{22} & d_{m2} \\ d_{13} & d_{23} & d_{m3} \\ \dots & \dots & \dots \\ d_{m1} & d_{m2} & d_{mn} \\ D_1 & D_2 & D_n \end{bmatrix} \qquad (2)$$

From the above equation (2), 'PDS_i' corresponds to the preprocessed document set for different domains '$D_1(\text{i. e.}, \text{image processing})$', '$D_2(\text{i. e. cloud computing})$', '$D_n$ $(\text{i. e. sensor networks})$' and so on. With this resultant value, it is simple to apply cosine semantic similarity to these vectors and identify how a document is related to another document. This is mathematically formulated as given below.

$$\text{Semantic Similarity Score} = \text{Cos}(\theta) = \frac{P_i Q_j}{|P_i||Q_j|} = \frac{\sum_{i,j=1}^{n} P_i Q_j}{\sqrt{\sum_{i=1}^{n} P_i^2}\sqrt{\sum_{j=1}^{n} Q_j^2}} \qquad (3)$$

From the above equation (3) 'P_i' and 'P_j' are elements of vector 'P' and 'Q' respectively, returning the similar documents 'SD' using the semantic similarity score (SSS). The pseudo code representation of Relative Semantic Cosine Document Similarity is given below.

Input: Preprocessed documents 'PD', set of nodes '$V = \{v_1, v_2, \dots, v_n\}$', set of edges '$E = \{e_1, e_2, \dots, e_m\}$'
Output: Similar Documents 'SD'
1: Begin 2: For each Preprocessed documents 'PD' with 'V' set of nodes and 'E' set of edges

3: Obtain the distance factor using (1)
4: Measure semantic similarity score using (3)
5: End for
6: End

Algorithm 2 Relative Semantic Cosine Document Similarity

As given in the above algorithm for each preprocessed documents in the form of nodes and edges, the objective remains in identifying similar documents with minimum time interval. This is performed in this work by using the vectorization concept. With the preprocessed documents as input, the algorithm first measures the distance factor. With the measured distance factor, next, the semantic similarity is measured for retrieving document similarity. With document similarity measured based on semantic cosine factor, similar documents are identified at minimum time interval.

Projection-oriented Neural Feature Extraction model

The hunt for a probable existence of certain irrelevant feature (i.e., words) in a huge size data in cloud can be difficult due to the curse of dimensionality issue. Due to this complication, consistently precise assessments for all smooth functions are not possible for huge size cloud data. In this work to minimize the information loss due to the difficulty in locating the comprehensive ontology [2] that covers all the concepts given in the documents, a Projection-oriented Neural Feature Extraction (PNFE) model is investigated. The unsupervised learning in a neural network using PNFE model therefore results in a single and multi-label features, or dimensionality reduction, of the similar documents.

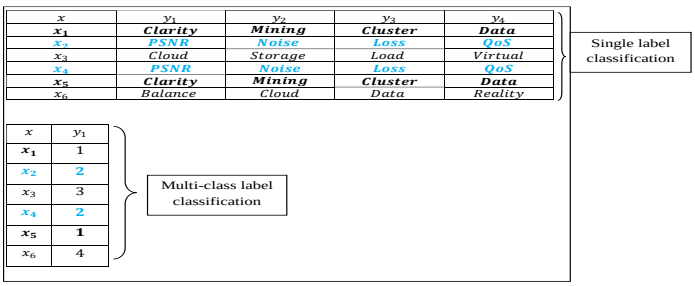

Figure 4 Sample single and multi-class label classification

In PNFE model, a feature is related with each projection direction where the classification function corresponds to a decision made by the neuron whether it covers all the concepts given in the documents or not for a given input (i.e. similar documents). Besides, with the inclusion of a threshold factor 'TF', we can say that an input possess a feature associated with that direction if its projection is larger than the threshold. Here, the threshold factor corresponds to the labels in the documents. Figure 4 shows sample single and multi-class label classification. From the above samples, we find that similar documents (i.e. 'x') 'x_1' and 'x_5' have the same labels, similarly, 'x_2' and 'x_4' have the same set of labels. Hence, single label classification and multi-class label classification are performed according to the domains and documents involved before uploading in the cloud database based on their domains. The PNFE model proceeds as follows. Given a compact set of similar documents '$SD = SD_1, SD_2, \dots, SD_n$', '$C = C_1, C_2, \dots, C_n$' be the number of classes, i.e. each class assigns the similar document 'SD' to the label 'l', if an only if the label is included in the similar document. Here, binary classifiers are used to perform single and multi-label document classification, i.e. the binary classifier '$BC =$

1', the result is multi-label document classifier or else the binary classifier '$BC = 0$', the result is single label document classifier. This is formulated as given below.

$$C(BC) = \bigcup_{i=1}^{2} C_i\,(BC) = l \qquad (4)$$

From the above equation (4), '2' binary classifiers are used where each similar document 'SD' is assigned to the label 'l', if the label is included in the similar document, signifying multi-label document classification. The mathematical representation of single classification is given below.

$$L = \bigcup_{p=1}^{P} l_p \qquad (5)$$

From the above equation (5), if 'P' represents the different sets of labels exist in the similar document, then each different set of labels is used as a new single label, signifying single classification. The pseudo code representation of Projection Neural Feature Extraction is given below.

Input: Similar Document 'SD', Threshold Factor 'TF', domains '$D = D_1, D_2, \ldots, D_n$'
Output: Dimensionality-reduced Single and multi-label feature extraction
1: Begin 2: For each Similar Document 'SD' with domains 'D' 3: Perform multi-label document classification using (4) 4: Perform single label document using (5) 5: End for 6: End

Algorithm 3 Projection Neural Feature Extraction algorithm

As given in the above Projection Neural Feature Extraction algorithm, for each Similar Document 'SD' obtained as input with domains 'D', the objective of the algorithm remains in extracting the features based on single and multi-label document classification. As a result, efficient feature extraction is said to take place with minimum complexity.

IV. EXPERIMENTAL EVALUATION

In this paper, we aim to reduce the computational complexity and time involved in clustering the documents with higher rate of accuracy in cloud environment. For this document clustering analysis, a subset of the original TDT2 corpus is used. The TDT2 corpus comprises data gathered from first half of 1998 from 6 difference sources. The sources here include 2 newswires, 2 radio programs and 2 television programs. The TDT2 corpus comprises 11201 on-topic documents that are classified into 97 semantic classes. In this subset, those documents occurring in more than one category were discarded and only largest 30 categories were retained, thus resulting in 9394 documents overall. The 9394 documents consist of data file containing variables 'fea' and 'gnd'. Here, 'fea' refers to the document-term matrix, where each row represents a document, whereas 'gnd' represents the label. Next, Doc ID refers to corresponding document name in original TDT2 corpus. Following, which comprise the terms present in the original TDT2 corpus. Entropy-based Boosting with Projection Neural Feature (EB-PNF) method for cloud storage is compared with the existing Semi Supervised Concept Factorization (SSCF) [1] and Fuzzy Clustering [2]. The proposal work plan to conduct experimental and analytical evaluation of clustering documents based on behavior in cloud computing environment with

data sets extracted from TDT2 corpus. The experimental evaluation of proposal work is conducted on various factors such as recall, precision, computational complexity, number of documents and document size.

Discussion

In this section, the performance evaluation is implemented in JAVA. The validation results of three different parameters, precision, recall and computational overhead with respect to documents is provided below. Detailed comparison analysis for the proposed EB-PNF method is made with the two existing methods, Semi Supervised Concept Factorization (SSCF) [1] and Fuzzy Clustering [2].

Recall rate

The first experiment considered is the recall rate. Recall refers to the ratio of relevant documents retrieved to the total number of relevant documents in TDT2 corpus. It is called as true positive rate. It is measured in terms of percentage (%).

$$R = \frac{\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}}{\{\text{retrieved documents}\}} * 100 \quad (11)$$

For example, let us consider the testing data that contains 80 documents on a specific topic. A search was performed on the topic and 60 documents were retrieved. Of the 60 documents retrieved, 45 documents were found to be relevant. Now, the precision and recall is measured as given below. Now, let, A (i.e. A = 45) represents the number of relevant documents retrieved, B (i.e. B = 35, [80-45]) represents the number of relevant documents not retrieved, and C (i.e. C = 15, [60-45]) represents the number of irrelevant documents retrieved. The sample calculations along with the graphical representation are given below for the EB-PNF method, SSCF [1] and Fuzzy Clustering [2] based on their documents respectively.

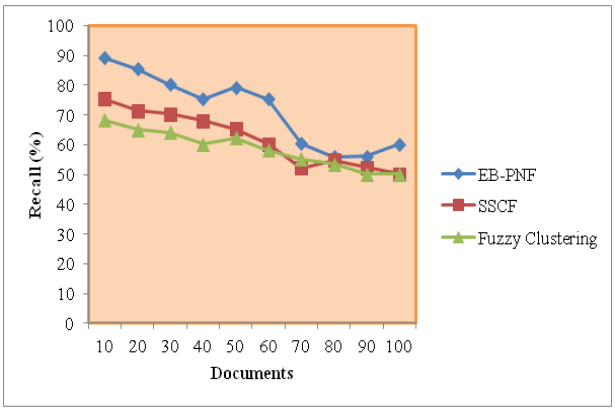

Figure 5 comparison of recall rate

Figure 5 given above shows the recall rate of all the three methods. This experiment is set to verify the high efficiency of the recall rate with respect to document size for clustering documents in cloud computing environment. In this experiment environment, we combine two different distance and natural language processing techniques to achieve the high efficiency recall rate. The recall rate is compared with SSCF [1] and Fuzzy Clustering [2] by capturing the same amount of document size. As illustrated in the above figure, the experimental results of recall rate for different document with same size (e.g. 10, 200, …., 1000) in semantic similarity stage is considered. From the experiment, we can learn that: 1) the recall rate is proportional to the number of documents used, and 2) to handle the same number of documents for measuring recall rate, EB-PNF method involves higher recall rate than [1] and [2]. Hence, as shown in the figure, the recall rate decreases with the increase in the number of documents and has insignificant gaps when the number of document is greater

than 50. That is to say, the recall rate for EB-PNF method is higher than those in [1] and [2]. This is because of the application of Relative Semantic Cosine Document Similarity in the EB-PNF method. The reasons for that are twofold. First, the former two methods measures the recall rate based on factorization and ontology methods with the aid of singular value decomposition whereas in EB-PNF method, two different factors, distance and natural language processing were applied, therefore reducing the irrelevant documents being retrieved. Second, for measuring the recall rate, the existing [1] and [2] methods considered only the pair-wise clustering for clustering documents in cloud environment, which is said to be compromised in case of the multi-class labels, whereas in EB-PNF method, semantic factors or similarity is performed with different documents and then with the optimal number of documents, clustering were performed. This in turn improved the recall rate involved in retrieving relevant documents in cloud environment using EB-PNF method by 15% compared to [1] and 22% compared to [2].

Precision rate

Followed by the recall rate, the second experiment considered for clustering documents before being uploaded into the cloud database is precision rate. Precision refers to the ratio of relevant documents retrieved to the total number of relevant and irrelevant documents retrieved. It is measured in terms of percentage (%).

$$P = \frac{\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}}{\{\text{relevant documents}\}} * 100 \qquad (12)$$

The sample calculations along with the graphical representation are given below for the EB-PNF method, SSCF [1] and Fuzzy Clustering [2] based on their documents respectively.

Precision rate incurred while clustering the documents before being uploaded for cloud database is one of the challenges to be addressed in cloud computing environment. With the increase in the number of documents, minimization of computational overhead cannot be attained. However, optimization can be achieved. The comparison of computational overhead for MR-WMCC method is measured and compared with [1] and [2] and is plotted in figure 6. The results reported in the figure confirm that with the increase in the tweet size, the computational overhead also gets increased.

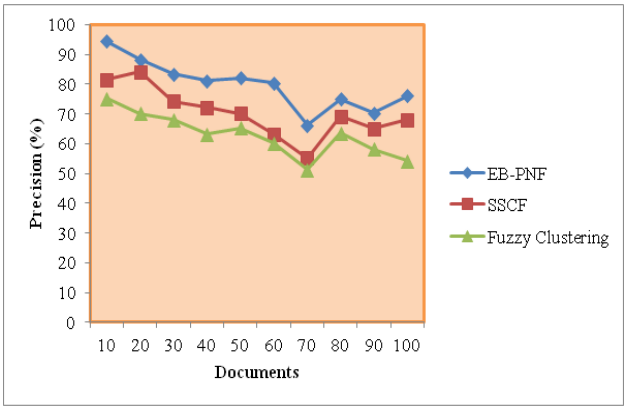

Figure 6 Measure of Precision rate

Figure 6 given above shows the comparison performance of precision rate for 10 different set of documents with document sizes in the range of 50KB to 500KB obtained at different time intervals. As a result, 100

different documents are observed in x axis and precision rate is observed in the y axis. With increase in the number of documents using the TDT2 corpus, precision rate for EB-PNF method also decreases. As a result, precision rate increases with the decrease in the number of documents. As a simulation, with '80' different number of documents considered for experimentation, the precision rate was found to be '75%' using EB-PNF method, '69.23%' using SSCF [1] and '63.52%' when applied with Fuzzy Clustering [2]. However, performance analysis on an average found EB-PNF method comparatively better than [1] and [2]. This is because of the Projection-oriented Neural Feature Extraction applied in the EB-PNF method that not only extracts the single document classification but also performs multi-label document classification using the threshold factor. By applying, above Projection Neural Feature Extraction algorithm, projection direction is used by the neuron to cover all the concepts given in the documents according to the threshold factor in an efficient manner. As the result, the fraction of retrieved documents that are relevant to the query is higher by applying the EB-PNF method. As a result, the precision rate for document retrieval in cloud computing is found to be comparatively higher using EB-PNF method by 14% compared to [1] and 27% compared to [2].

Computational complexity

While clustering documents before uploading the documents into cloud database, the complexity involved should be analyzed. With this objective, the third experiment is conducted for measuring the computational complexity using JAVA and comparison is made with two other methods, namely SSCF [1] and Fuzzy Clustering [2]. Computational complexity refers to the time involved in extracting the dimensionality reduced single and multi-label features with respect to the overall document size considered for experimentation.

$$CC = D_{size} * \text{Time (DRF)} \qquad (13)$$

From the above equation (13), the computational complexity 'CC' is measured according to the document size 'D_{size}' extracted from TDT2 corpus dataset and the time consumed for obtaining the dimensionality reduced features 'Time (DRT)'. It is measured in terms of milliseconds (ms). The sample calculations along with the graphical representation are given below for the EB-PNF method, SSCF [1] and Fuzzy Clustering [2] based on their documents respectively.

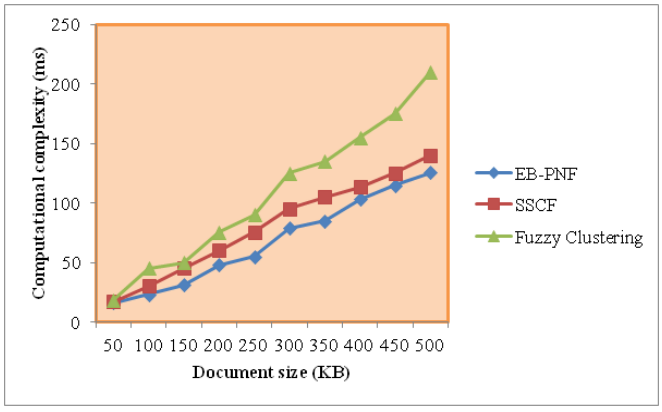

Figure 7 Measure of computational complexity versus document size

For all cases as shown in figure 7, the computational complexity is increasing with the document size considered from different cloud users with respect to different document retrieval. The targeting results of computational complexity using EB-PNF method is compared with the two state-of-the-art methods [1], [2] in figure 7 is presented for visual comparison. Our method differs from the SSCF [1] and Fuzzy Clustering [2] in that we have incorporated Projection Neural Feature Extraction algorithm between domains for obtaining

dimensionality reduced single and multi-label feature extraction. With this, the dimensionality reduction is said to be achieved. With the dimensionality reduced resultant features, feature extraction results are said to be efficient. Here, binary classifiers are used for multi-label document classifier, whereas, single label classification is carried out separately for different sets of labels occurring in the similar document, that in turn selects the domain frequent documents and therefore reduces the dimensionality factor. Besides, dominant domains within a document and presence of similar domains in other documents are extracted by applying the semantic relation. With the resultant values obtained through semantic distance factor and natural language processing, further minimizes the computational complexity during document clustering. Therefore the computational complexity for clustering similar documents using EB-PNF method is reduced by 17% compared to SSCF [1] and 36% compared to Fuzzy Clustering [2] respectively.

V. **CONCLUSION**

One of the major concerns for the users who access the cloud database is the relevancy problem and high document retrieval time. Document clustering is one of the most suitable solutions before uploading the document into the cloud database. However, with the explosive growth of huge size data in cloud there requires an urgent need to provide for high rate of accuracy for document retrieval stored in cloud. In this article Entropy-based Boosting with Projection Neural Feature (EB-PNF) method is designed that can be employed as a document clustering method before uploading into the cloud database. Initially, Relative Semantic Cosine Document Similarity algorithm is applied to the pre-processed document, to obtain similarity between documents. With similar documents retrieved, feature extraction is performed by applying Projection Neural Feature Extraction algorithm. With this, dimensionality reduced features are extracted, therefore minimizing the computational complexity involved. Through the experiments using real traces, we observed that our document clustering method for similar document retrieval in cloud reduced computational complexity and improved the precision and recall rate compared to the existing methods.

REFERENCES

[1] MeiLu, Xiang-Jun Zhaob, Li Zhanga, Fan-Zhang Lia, “Semi-supervised concept factorization for document clustering”, Information Sciences, Elsevier, Volume 331, February 2016, Pages 86-98

[2] Lin Yue, Wanli Zuo, Tao Peng, Ying Wang, Xuming Hand, “A fuzzy document clustering approach based on domain-specified ontology”, Data & Knowledge Engineering, Elsevier, Jun 2015, Pages 148-166

[3] N K Nagwani, “Summarizing large text collection using topic modeling and clustering based on MapReduce framework”, Journal of Big Data, Springer, Volume 2, Issue 6, January 2015, Pages 1-18

[4] Yong-Il Kim, Yoo-Kang Ji, Sun Park, “ Big Text Data Clustering using Class Labels and Semantic Feature Based on Hadoop of Cloud Computing”, International Journal of Software Engineering and Its Applications Volume 8, Issue 4 2014, Pages 1-10

[5] M. Tarik Altuncu, Sophia N. Yaliraki, Mauricio Barahona, “Content-driven, unsupervised clustering of news articles through multiscale graph partitioning”, KDD Data Science, Journalism and Media (DSJM2018), ACM, May 2018, Pages 1-8

[6] K. M. Baalamurugan, S. Vijay Bhanu, “An efficient clustering scheme for cloud computing problems using metaheuristic algorithms”, Cluster Computing, Springer, January 2018, Pages 1-8

[7] Charlotte Laclau, Mohamed Nadif, “Hard and fuzzy diagonal co-clustering for document-term partitioning”, Neurocomputing, Elsevier, Volume 193, June 2016, Pages 133-147

[8] Malik Tahir Hassan, Asim Karim, Jeong-Bae Kim, Moongu Jeon, "CDIM: Document Clustering by Discrimination Information Maximization", Information Sciences, Elsevier, Volume 316, 20 September 2015, Pages 87-106

[9] Tingting Wei, Yonghe Lu, Huiyou Chang, Qiang Zhou, Xianyu Bao, "A semantic approach for text clustering using WordNet and lexical Chains", Expert Systems with Applications, Elsevier, Volume 42, Issue 4, March 2015, Pages 2264-2275

[10] Võctor Mijangos, Gerardo Sierra, Azucena Montes, "Sentence level matrix representation for document spectral clustering", Pattern Recognition Letters, Elsevier, Volume 85, January 2017, Pages 29-34

[11] Jayaraj Jayabharathy and Selvadurai Kanmani, "Correlated concept based dynamic document clustering algorithms for newsgroups and scientific literature", Decision Analysis, Springer, Volume 1, Issue 3, December 2014, Pages 1-21

[12] Nevena Golubovic, Chandra Krintz, Rich Wolski, Balaji Sethuramasamyraja, Bo Liu, "A scalable system for executing and scoring K-means clustering techniques and its impact on applications in agriculture", International Journal of Big Data Intelligence (IJBDI), May 2016, Pages 1-13