OPEN ACCESS Asian Journal of Scientific Research

ISSN 1992-1454
DOI: 10.3923/ajsr.2018.

Research Article
Conceptual Relevance Based Document Clustering Using Concept Utility Scale

A. Kousar Nikhath and K. Subrahmanyam

Department of Computer Science and Engineering, Koneru Lakshamaiah Education Foundation, 522502 Guntur, Andhra Pradesh, India

Abstract

Background and Objective: Volume of documents that is generated, processed, stored and retrieved recently is very high and there is integral need of more robust solutions for document retrieval. In this context, clustering is one of the data-mining models to achieve document tracing and retrieval. Many of the document clustering models evinced in contemporary literature, which depends on individual terms of each document as bag of words and clustering documents based on the term frequency, which is a critical constraint of these models. In this paper, the emphasis of this manuscript is to develop a novel document clustering technique that performs clustering by using concept relations of the given documents to achieve more effective document clustering. **Methodology:** The proposed model tends to cluster the documents based on their concept relations. In order to this, the proposed model depicted a scale called concept utility scale (CUS), which will in use further to identify the concept scope in order to define the clusters from the given document corpora. The feature optimization and document clustering that performed by using t-score, which is a statistical scale for estimating whether the chosen two vectors are similar or diversified. Hereafter, the depicted model denotes as t-CUS. Proposed solution evaluated by renowned statistical metrics like sensitivity, accuracy, fall-out and specificity. Experiments carried out on datasets comprising specific kind of literature. **Results:** The experimental study evincing that the proposed model is effective in improving the accuracy of clustering and information retrieval. **Conclusion:** In addition, the computation complexity is low and linear with the proposed solution t-CUS.

Key words: Concept utility scale, t-score based document clustering, t-score, semi-supervised learning, frequency

Received: Accepted: Published:

Citation: A. Kousar Nikhath and K. Subrahmanyam, 2018. Conceptual relevance based document clustering using concept utility scale. Asian J. Sci. Res., CC: CC-CC.

Corre sponding Author: A. Kousar Nikhath, Department of Computer Science and Engineering, Koneru Lakshamaiah Education Foundation, 522502 Guntur, Andhra Pradesh, India

Competing Interest: The authors have declared that no competing interest exists.

Data Availability: All relevant data are within the paper and its supporting information files.

INTRODUCTION

Rapid development ICT has changed the dynamics of communication. Quantum of data generated, stored and processed every minute is very huge. Irrespective of the size of data and the type of storage, it is very essential that the data have to manage in structured and a classified manner for retrieval of the data an effective manner. It is very important to ensure the data managed systematically for effective navigation, summarization and organized structure, which can support in easy access to the requisite data in quick turn around time.

Document management systems have a vital role in the process of structured management of data. Fast and high-quality solutions with effective document clustering algorithms have vital role in the successful managing the data. Document clustering algorithms has the scope of intuitive navigation/browsing mechanism and by organizing huge volumes of information into small and related clusters. The process of retrieval can be more effective as certain factors like reduction of cluster-driven dimensionality, term weighing[1] and the process of query expansion[2] can support in improving the retrieval performance.

Majority of the search engines in the real-time environment relies on the string matching method and thus the documents retrieved may not be apt or might have some irrelevant document extractions too for a search query. If a robust document clustering approach is adapted and implemented, it will assist in organizing the document corpus in more structured manner. It will also support in addressing short comings in the existing methods of information retrieval.

In vivid range of text mining and information retrieval processes, the process of document clustering methods was tired in the earlier proposed solutions. However, earlier the document clustering focused for improving the precision, recall in the information retrieval systems and in identifying the close relativity of a document to the other documents, recently, the system targeted for many other factors too[3-5].

Clustering was adapted the process of browsing through a repository of documents and in organizing the results that are returned by a search engine for a search query[6,7]. Often, uses the document clustering as a process for automatically generating hierarchical clusters of documents[8]. For instance, in the case of a search engine, if the user poses a query, thousands of pages returned as results, by the search engine but the challenge is the relevance of those results. Effective clustering models play a major role in improving the accuracy of such factors.

Profoundly in many of the existing models of document clustering methods, vector space model (VSM) used for data representation in text classification and clustering process[9]. Feature vector uses for representation of documents in the VSM model. Every feature vector term comprises the term-weights of terms in the document. Term frequency-inverse document frequency (TF-IDF) is a statistical measure weight that uses for evaluating "importance of a word" for a document in the group of documents[10]. Significance for a word goes high with the repetition of a word in a document. Similarities amidst the documents are measured based using several similarity measures those chosen based on feature vector[11]. Some of the common measures that used are Jaccard measure and the Cosine measure.

Li and Zhu[12] proposed a contemporary method for document clustering in research literature. The proposed solution relies on negative matrix factorization (NMF), Topic discovery that depends on test or theory. This method clusters the research literature documents that comprise test or theory and NMF. The NMF method considered as an effective process for high dimensionality reduction in the text data and towards clustering them.

Test or theory is adapted for discovering topics for the documents those clustered by NMF method. The process performed is to construct learning matrix and comparison matrix for analysis. It is imperative from the some of the earlier experimental studies that the combination of NMF and Test or theory offers significant outcome. Many of the document clustering algorithms based on term frequency[13-15]. Varied researchers have focused on clustering based on synonyms and hypernyms[16-22].

It is imperative from the detailed analysis of the document clustering algorithms[23-36] that the profoundly the documents are represented based on the following factors:

- Firstly, it depends on pair wise concept or phrase, in which the similarity relationship amidst the sentences observed as per its usage[24,36]
- Secondarily, the tree representation is used and the similarity amidst two nodes or objects is identified and clustered[26-28]
- Component based clustering algorithm that uses the object-oriented software representation for document modelling, usage of Cosine and Euclid measure for clustering of document is adapted[25]
- The other model adapted is to observe semantic relations and representing the documents based on the related terms[33]
- The model depicted by Jing *et al.*[35] is using the concept and feature vector for representation

Majority of existing contributions (as detailed above) discussed about web page information representation tracking and retrieval process.

Our earlier contribution Nikhath and Subrahmanyam[37] proposed an optimal document clustering technique, which is using genetic algorithm to optimize the clusters. The objective of this model is to optimize the clusters that defined based on term frequency and document frequency[38] method. Some of the intrinsic aspects observed in the TF-IDF method are:

- It fails in differentiating degree of semantic importance for every term weights assigned without differentiating similarities amidst the semantically important and irrelevant words within the document. In addition, it does not consider polysemous and synonyms

To enhance the quality of cluster in the previously mentioned document sets (scientific and domain specific), the model presented by Jayabharathy and Kanmani[39] targeted the clustering of document based on terms and their technically related terms. The authors target domain specific dictionary for extracting related terms as concepts.

It is imperative from the review of literature that around 60% of the clustering techniques relies on term frequency models and around 30% of the clustering techniques adapt annotation tools, use synonyms and hypernyms for evaluating the concepts. Synonyms and hypernyms usually extracted using word net lexical database[40].

Application of synonym and hyponym oriented clustering solutions may not be very effective over the scientific literature and pure technical documents as, majorly the tracks of documents might comprise completely domain specific technical terms and synonym process may not be apt for identification of related terms.

The critical constraint observed among all of these existing models is the usage of term frequency, which is often fails to identify the conceptual relation. This is due to the circumstances like, the concept of the article may strongly reflect by infrequent terms those exists in meta text, sparsely in the description and the terms those are frequent may not be related to the concept or topic. Hence, in such cases the all of the existing models that includes the contemporary model called “correlated concept based maximum resemblance document clustering" (CCMARDC)[39] also fails to cluster documents with maximal accuracy.

In this context, this manuscript introducing a new term weighting strategy called concept utility scale. The concept utility scale defines the term weight rather by only its occurrence count.

MATERIALS AND METHODS

The proposed model is discovering the concept compatibility of the documents of the given corpus by the concept utility scale. In regard to this the proposed model discovers the concept utility scale further this scale will be used to form the class labels under concept relation and afterwards, clusters the documents based on these class labels.

Concept utility scale: This section explores the notations used in the proposal and the process of discovering their values. The proposed document clustering is centric to the concept utility scale, which is an extension to the model devised in our earlier contribution called concept based document clustering by corpus utility scale (COCUS)[41] that depends on the set of documents as concept-base that selected under supervised learning process. The input documents given to clustering process considered as corpus. The proposed process of defining the concept utility scale initially preprocess the given corpus (documents to be clustered) that discards stop words and noise, then stems the “ing” and “ed” forms. The resultant words from each document of the corpus represented further as vector in bag of words. Further, considers all unique words in the resultant bag of words as a set and measures the occurrence in concept-base of each term in respective set. Further assesses document level occurrence of each term. Afterwards, the document level utility of each term estimated by using these term occurrences of concept-base level and document level. Then the document level utility of a given term-set measured, which is the aggregate of document level utility of all terms exists in respective term set. In similar passion, the corpus level utility of the given term set will be assessed, which the aggregate of each document level utility of respective term set towards all documents in corpus. That followed by the assessment of document utility, which is cumulative of the respective document level utility for all the terms comprised in the document. In addition, the corpus utility evaluated as the cumulative of the document utility of all documents that constituted in the corpus. Further, the concept utility scale assessed by multiplying the corpus utility by the utility threshold that usually varies between 0 and 1.

Concept-base (CB): Set of documents strongly related to the concept, those selected from domain expert’s recommendation or prototype documents of the concept.

Concept-base level term occurrence (c(t, CB)): The number of times term t occurs in concept-base.

Document level term occurrence ($c(t, d_i)$): Represents the occurrence count of the term t in document d_i.

Document level term utility (u): Document d_i level term utility of term t is the product of concept-base level term occurrence c(t, CB) and the term occurrence at document level $c(t, d_i)$of the respective term t.

$$u(t, d_i) = c(t, CB) \times c(t, d_i) \tag{1}$$

The concept utility scale estimated as follows:

$$cu = \sum_{k=1}^{|Cor|} \left\{ \sum_{j=1}^{|TS|} \left\{ \sum_{i=1}^{|\{ts_j \exists ts_j \in TS|} \left\{ u(t_i, d_k) \exists t_i \in ts_j \wedge ts_j \in d_k \right\} \right\} \right\} \tag{2}$$

In the above equation.

The notation:

$$\sum_{i=1}^{|\{ts_j \exists ts_j \in TS|} \left\{ u(t_i, d_k) \exists t_i \in ts_j \wedge ts_j \in d_k \right\}$$

indicated the aggregation of the utility of the terms exist in term-set ts_j also exists in document d_k, which is denoted as document level term set utility $u(ts_j, d_k)$ of the term set ts_j in respective of document d_k.

The notation:

$$\sum_{j=1}^{|TS|} \left\{ \sum_{i=1}^{|\{ts_j \exists ts_j \in TS|} \left\{ u(t_i, d_k) \exists t_i \in ts_j \wedge ts_j \in d_k \right\} \right\}$$

indicated the aggregation of the document level utility of the all term sets exist in document d_k, which denoted as document utility $u(d_k)$of respective document d_k.

The notation:

$$\sum_{k=1}^{|Cor|} \left\{ \sum_{j=1}^{|TS|} \left\{ \sum_{i=1}^{|\{ts_j \exists ts_j \in TS|} \left\{ u(t_i, d_k) \exists t_i \in ts_j \wedge ts_j \in d_k \right\} \right\} \right\}$$

is indicating the aggregation of document utility of all documents exist in given corpus Cor, which is denoted as the corpus utility cu.

Further, concept utility scale (cus) is estimated as the product of corpus utility cu and the given corpus utility threshold $\{\tau \exists 0 < \tau < 1\}$, which is denoted by the following:

$$cus = cu(Cor) \times \{\tau \exists 0 < \tau < 1\} \tag{3}$$

Similarly, the corpus level term set utility measured as follows:

$$\forall_{i=1}^{|TS|} \{ts_i \exists ts_i \in TS\}$$
$$ctu(ts_i) = \sum_{k=1}^{|Cor|} \left\{ \sum_{j=1}^{|\{ts_i \exists ts_i \in TS|} \left\{ u(t_j, d_k) \exists t_j \in ts_i \wedge ts_i \in d_k \wedge d_k \in Cor \right\} \right\} \tag{4}$$

In the above equation, the aggregation of document level term set utility of respective term set ts_i is carried out, which is denoted as corpus level term set utility $ctu(ts_i)$ of respective term set ts_i.

T-Score[42]: This metric scale is significant to estimate the diversity of the values exists in two vectors, which indicates that both vectors are having similarity or diversified. Hence, the proposed model adapted t-score to reduce the features (term sets) count and to cluster the documents based on their similarity. The diversity of the values in two different vectors represent by t-score, which estimated as follows:

$$t\text{-}score = \frac{(M_{v1} - M_{v2})}{\sqrt{\dfrac{\sum_{i=1}^{|v1|} (x_i - M_{v1})^2}{|v1| - 1} + \dfrac{\sum_{j=1}^{|v2|} (x_i - M_{v2})^2}{|v2| - 1}}} \tag{5}$$

Here in the above equation:

- M_{v1}, M_{v2} represents the mean of the values observed in respective vectors
- The notations x_i, x_j represents each element of respective vectors v1, v2 of corresponding sizes |v1|, |v2|

The t-score is the ratio between the mean differences of respective vectors and the square root of sum of mean square distances of the respective vectors.

Then find the degree of probability (p-value)[43] in t-table[42] for the t-score obtained. The p-value that is less than the probability threshold indicates both vectors are distinct; hence, the feature representing respective vectors is optimal feature.

The t-score adapted here in this proposed model, since each document represented as a vector representing the document level utility of the features, which are the selected term sets. Hence it is obvious to use t-score to identify the given two documents are diversified or similar. This metric also feasible to notify the given two term sets are similar or divergent, if similarity found between any given pair of term sets, then cumulate those two term sets, which helps to reduce the number of features (term sets).

T-score based document clustering using concept utility scale: This section explores the process of document clustering, which is using t-score and concept utility scale initial step of the clustering process is to identify significant term sets as features using concept utility scale, further filters these features using t-score to reduce the number of features count. Further, clusters the given documents based on the optimal features and their similarity assessed by t-score in respective of given documents.

Features by concept utility scale: A term-set said to be the feature that used to notify the similarity or diversity between given two documents, which is having significance in semi-supervised learning if and only if the corpus level utility of that feature is greater than the utility scale, which are determined as follows.

The term sets will be find in the ascending order of the term set size from 1 to n. The one or more terms as a set considered as term set, if corpus level term set utility of that term set is greater than the concept utility scale. In order to define the term sets, initially obtains the term sets of size 1 that are having ctu≥cus. These term sets are moved to the set TS at level 1, which are denoted as (where i is 1 in this case) in further discussion.

The unique term sets of size 2 will be defined further, which are the resultant sets that are having ctu≥cus and obtained from union operation performed on each pair of term sets exists in TS at level i+1 (that are denoted as TS_1).

Further, the similar process applied recursively on term sets exists in TS at level i that results term sets having ctu≥cus, which placed further in TS at level i+1. If no term set exists in TS at level i+1 then this process ends. Upon completion of this process, the set TS contains term sets of size 1to n.

The algorithmic flow of the process to obtain optimal term sets follows.

Let T be the set contains all unique terms obtained from the all of the preprocessed documents in the given corpus.

```
Step 1:  Let TS be the set contains term sets of size 1 to n and having desired corpus level term set utility (ctu≥cus), which is initially empty
Step 2:  $\overset{|T|}{\underset{i=1}{\forall}}\{t_i \exists t_i \in T\}$ Begin
Step 3:    If (ctu($t_i$)≥cus) Begin // the Corpus level term set utility ctu will be measured as explored in Eq. 4 and concept utility scale is measured as explored in Eq. 2 and 3.
Step 4:      $TS_1 \leftarrow t_i$// $TS_1$ contains all term sets of size 1 with specified corpus level term set utility (ctu≥cus)
Step 5:    End // of step 4
Step 6:  End // of step 3
Step 7:  REPEAT {
Step 8:    i=|TS| // |TS| indicated the present last level of the optimal term sets
Step 9:    $\overset{|TS_i|}{\underset{j=1}{\forall}}\{ts_j \exists ts_j \in TS_i\}$ Begin // for each term set exists in TS at level i
Step 10:     $\overset{|TS_i|}{\underset{k=1}{\forall}}\{ts_k \exists ts_k \in TS_i \wedge j \neq k\}$ Begin // for each term set exists in TS at level  and term set $ts_j$ is not equal to $ts_k$
Step 11:       If(($ts_j \cup ts_k$)∉TS)^(ctu($ts_j \cup ts_k$)≥cus)) Begin // the condition indicates that the resultant term set from the union operation applied on term set $ts_j$ and term set $ts_k$ does not exist at any level of TS and corpus level term set utility of the resultant term set ctu($ts_j \cup ts_k$)is greater than or equal to concept utility scale
Step 12:         $Ts_{i+1} \leftarrow (ts_j \cup ts_k)$// move resultant term set ($Ts_j \cup ts_k$)to next level i+1 of TS
Step 13:       End // of step 12
Step 14:     End // of step 10
Step 15: UNTIL $TS_{i+1}$ is not empty // repeat (go to step 8) if the term sets exist at level i+1 of TS
```

Cluster formation: This section explores the proposed t-score based document clustering, which is using concept utility scale to select optimal term sets as features. The process of defining concept utility scale and identifying the optimal term sets as features using this concept utility scale.

A matrix M of size |TS| X |Cor| built, such that each column of the matrix contains the document level term set utility of an optimal term set in respective of a document in the corpus. The notation |TS| is the total number of optimal term sets, which considered as number of rows (horizontal vectors) in the matrix M. The notation |Cor| is the number of documents in the given corpus Cor that considered as number of columns (vertical vectors) in the matrix M. Each horizontal vector

Table 1: Matrix M of size |TS| X |Cor| that represents document level term set utilities discovered

↓ row header	column header →	d_1	d_2	.	.	.	$d_{\|Cor\|}$
ts_1		dtu ($ts_{1,}$ d_1)	dtu ($ts_{1,}$ d_2)	.	.	.	dtu ($ts_{1,}$ $d_{\|cor\|}$)
ts_1		dtu ($ts_{2,}$ d_1)	dtu ($ts_{2,}$ d_2)	.	.	.	dtu ($ts_{2,}$ $d_{\|cor\|}$)
.		.	.	.	.	.	.
.		.	.	.	.	.	.
.		.	.	.	.	.	.
$ts_{\|TS\|}$		dtu ($ts_{\|TS\|,}$ d_1)	dtu ($ts_{\|\|TS\|,}$ d_2)	.	.	.	dtu ($ts_{\|TS\|,}$ $d_{\|cor\|}$)

contains the document level term set utility of a term set in respective to all documents and each vertical vector contains the document level term set utility of all term sets in respective to a document. In a gist, each row of size |Cor| represents a term set and its document level utility scores and each column of size |TS| represents a document and that document level utility of all term sets. The mock representation of the matrix M depicted in Table 1. Further step of the process reduces the possible number of term sets that controls the clustering process complexity that follow.

Transpose the matrix M as M, arrange the vertical vectors (each vector contains the document level term set utility of a term set in respective to all documents) of M′ in descending order of the aggregate value of each vector. The vertical vector with highest value that depicted from the sum of all the values in corresponding vector will be the first vertical vector of the matrix M′, which followed by the other vertical vectors in descending order of their aggregate value.

Further, step cumulates all the vertical vectors that are not distinct based on t-score from each other. Concerning this, the document level term set utility found in respective columns of the vectors that are not distinct each other will aggregate. The algorithmic flow of the minimizing the vertical columns of the matrix M′ is follows.

Term set pruning:

```
Step 1: Let matrix M′ is having vertical vectors in descending order of their aggregate values
Step 2: $\overset{CS}{\underset{i=1}{\forall}}\{vv_i \exists vv_i \in M'\}$ Begin // each vertical vector of the matrix M′, CS is number of vertical vectors
Step 3:     VG← $vv_i$// is a set having all vertical vectors that are not distinct under t-score
Step 4:     $\overset{CS}{\underset{j=i+1}{\forall}}\{vv_j \exists vv_j \in M'\}$ Begin // each vertical vector from (i+1)th vertical vector of the matrix M′
Step 5:         vvg←VG// cloning the set VG as vvg
Step 6:         dist = false //Boolean variable initialized to true.
Step 7:         $\overset{|vvg|}{\underset{k=1}{\forall}}\{vv_k \exists vv_k \in vvg \wedge vv_k \neq vv_j\}$ Begin // each vector in vvg
Step 8:             If (p-value(t-score($vv_k$, $vv_j$))<pvt) Begin // the degree of probability p-value observed for t-score of v $v_j$, $vv_k$ is less than the degree of probability threshold pvt (usually 0.01,0.05 or 0.1) that indicates both vectors $vv_j$, $vv_k$ are distinct
Step 9:                 dist←true
Step 10:                Break // the loop in step 7 and control goes to step 12
Step 11:            End // of step 8
Step 12:            If (dist = false) Begin //indicates that the vertical vector $vv_j$ is similar to all vectors in $vv_g$
Step 13:                VG←$vv_j$
Step 14:                M′ \ $vv_j$// discarding vertical vector $vv_j$ from M′
Step 15:                CS-1 // reducing the number of vertical vectors by 1
Step 16:            End // of step 12
Step 17:        End // of step 7
Step 18:    End //of step 4
Step 19:    Find the union of all term sets representing the vertical vectors in VG, which is further referred as term set of vertical vectors $vv_i$ in matrix M′
Step 20:    Aggregate the values of the same column in respective vertical vectors of VG and move the resultant value to respective column of the vertical vector $vv_i$ of matrix M′
Step 21: End //of step 2
```

The resultant matrix M′ used further to cluster the documents as explored further.

Transpose the matrix M′ as M′ such that all vertical vectors represent a document and its document level term sets utility of all term sets (Table 1).

Then arrange the all-vertical vectors in descending order of their aggregate value.

Then generate clusters by cumulate the documents representing the vertical vectors, which are fo und to be not distinct by t-score. The process of cumulating the documents, as a cluster is identical to the steps (exclude the step 20) involved in term set pruning, for better understandability of process, the algorithmic flow of the cluster formation depicted in APPENDIX-A.

APPENDIX-A: Cluster formation

```
Step 1:  Let matrix M be the transpose of the matrix M′, such that all vertical vectors are ordered in descending order of the aggregate of the values exist in respective vertical vectors.
Step 2:  $\underset{i=1}{\overset{CS}{\forall}}\{vv_k \exists vv_i \in M\}$ Begin // each vertical vector of the matrix M′, CS is number of vertical vectors
    Step 3:  VG←$vv_i$ // is a set having all vertical vectors that are not  distinct under t-score
    Step 4:  $\underset{j=i+1}{\overset{CS}{\forall}}\{vv_j \exists vv_j \in M\}$ Begin // each vertical vectors from (i+1)th vertical vector of the matrix M′
        Step 5:  vvg←VG// cloning the set VG as vvg
        Step 6:  dist = false //Boolean variable initialized to true
        Step 7:  $\underset{k=1}{\overset{|vvg|}{\forall}}\{vv_k \exists vv_k \in vvg \wedge vv_k \neq vv_j\}$ Begin // each vector in vvg
            Step 8:  If (p-value(t-score($vv_k$, $vv_j$))<pvt) Begin // the degree of probability p-value observed for t-score of $vv_j$, $vv_k$ is less than the degree of probability threshold pvt(usually 0.01, 0.05 or 0.1) that indicates both vectors $vv_j$, $vv_k$ are distinct
                Step 9:  dist←true
                Step 10: Break // the loop in step 7 and control goes to step 12
            Step 11: End // of step 8
            Step 12: If (dist = false) Begin //indicates that the vertical vector $vv_j$ is similar to all vectors in vvg
                Step 13: VG←$vv_j$
                Step 14: M\$vv_j$// discarding vertical vector $vv^j$ from M′
                Step 15: CS = CS-1 // reducing the number of vertical vectors by 1
            Step 16: End // of step 12
        Step 17: End // of step 7
    Step 18: End //of step 4
    Step 19: Cumulate the all documents representing the vertical vectors in VG, which is further referred as cluster
Step 20: End //of step 2
```

RESULTS AND DISCUSSION

The performance of the proposed model assessed through set of statistical metrics[44] called accuracy (to form the clusters), specificity, sensitivity and fallout. In addition, the process complexity of the t-CUS estimated. In order to notify the significance of the proposal, the performance assessment done by comparing with other contemporary model CCMARDC[39], which clusters the scientific articles based on the concept relevance.

The set of documents representing divergent scientific topics from CORA dataset[45] considered to perform experimental study. The documents labeled based on the topic to which that topic relates. The total number of documents considered is 2080, which labeled with 16 divergent topics. The 25% of these documents considered as knowledge base, which strongly related to the corresponding topics. The document and topic relevance estimated according to the meta-text like keywords. Sparse number of documents represents the set of topics, which is to evince the robustness of the proposed model. The rest of 75% of the labeled documents are unlabeled and given to cluster by t-CUS and CCMARDC. The implementation of the proposal and the evaluation of the performance metrics carried using R language[46].

The number of documents considered for concept base is 513 and the rest of the documents of size 1567 are unlabeled and used as input to the both clustering techniques. The number of topics and the number documents representing each topic in concept base and input corpus depicted in Table 2.

The results obtained from t-CUS and CCMARDC depicted in Table 3 and 4, respectively. In order to assess the metrics accuracy, sensitivity and fallout at cluster level, the documents actually fit in to a respective cluster denoted as positives and the rest of all documents denoted as negatives. The true positives (TP) are the positive documents that are associated with resultant cluster. The false positives are the negative documents that are associated with resultant cluster. The false negatives are the positive documents not associated to corresponding resultant cluster and the rest of the documents are said to be true negatives, which are negative documents not associated with resultant cluster.

The metric accuracy indicates the ability of identifying the combination positive and negative documents in relate to respective cluster. The average of cluster accuracy found for t-CUS is 99%, which is 1.7% more that compared to CCMARDC. The significance of t-CUS that compared to CCMARDC evinced in true positive detection rate in respective of each cluster that referred as sensitivity. The sensitivity observed for t-CUS is 97% that around 9% more than the sensitivity observed for CCMARDC.

Similarly, detection of negative documents in respective to resultant cluster that denoted as specificity, which is approximately same in both t-CUS and CCMARDS. The fallout is a metric that depicts the scope of negative documents presence in respective clusters, which evinced as 0.2 and 0.6% in t-CUS and CCMARDC, respectively.

Table 2: Number of documents and topics used as input corpus and concept base

Topic ID	Total number of documents per each topic	Number of documents used as concept base	Number of documents per topic used as corpus
1	49	12	37
2	49	12	37
3	53	13	40
4	34	8	26
5	203	50	153
6	145	36	109
7	126	31	95
8	121	30	91
9	170	42	128
10	157	39	118
11	143	35	108
12	163	40	123
13	222	55	167
14	171	42	129
15	134	33	101
16	140	35	105

Table 3: Clusters and the results obtained for statistical metrics from t-CUS

TP	FP	TN	FN	Accuracy	Sensitivity	Fallout
36	0	1530	1	0.999	0.973	0
35	0	1530	2	0.999	0.946	0
39	2	1525	1	0.998	0.975	0.001
23	2	1539	3	0.997	0.885	0.001
149	0	1414	4	0.997	0.974	0
108	2	1456	1	0.998	0.991	0.001
93	3	1472	2	0.999	0.979	0.002
90	2	1474	1	0.998	0.989	0.001
127	4	1435	1	0.997	0.992	0.003
117	0	1449	1	0.999	0.992	0
107	7	1452	1	0.995	0.991	0.005
117	3	1441	6	0.994	0.951	0.002
160	4	1396	7	0.993	0.958	0.003
125	5	1433	4	0.994	0.969	0.003
97	2	1464	4	0.996	0.96	0.001
104	4	1458	1	0.997	0.99	0.003

True Positives (TP): The documents actually belong to the cluster and found in the resultant cluster. False Positives (FP): The documents actually not belong in to cluster but found in the resultant cluster. True Negatives (TN): The documents actually not belong in to cluster and found in other resultant clusters. False Negatives (FN): Documents actually belongs to the cluster but not fall in to that cluster

Table 4: Clusters and the results obtained for statistical metrics from ccmardc

TP	FP	TN	FN	Accuracy	Sensitivity	Fallout
31	2	1528	6	0.995	0.838	0.001
30	3	1527	7	0.994	0.811	0.002
39	10	1517	1	0.993	0.975	0.007
14	9	1532	12	0.987	0.538	0.006
137	9	1405	16	0.984	0.895	0.006
103	12	1446	6	0.989	0.945	0.008
84	3	1469	11	0.991	0.884	0.002
89	9	1467	2	0.993	0.978	0.006
111	24	1415	17	0.974	0.867	0.017
105	6	1443	13	0.988	0.89	0.004
98	17	1442	10	0.983	0.907	0.012
109	8	1436	14	0.986	0.886	0.006
156	3	1397	11	0.991	0.934	0.002
125	11	1427	4	0.99	0.969	0.008
90	6	1460	11	0.989	0.891	0.004
97	17	1445	8	0.984	0.924	0.012

CONCLUSION AND FUTURE RECOMMENDATIONS

The concept based document clustering is proposed here in this manuscript. The proposed model is defined a concept utility scale to discover the possible concept related term sets as features, which is from the influence of utility scale used in utility mining models. The covariance assessment method called t-score is used in two different contexts, one is to reduce the features and the other is to cluster the documents based on the similarity between features of the respective documents. The experimental study conducted on set scientific journal documents of 16 divergent topics from CORA dataset. The results obtained for both t-CUS and CCMARDC indicating that the cluster accuracy is nearly optimal in both cases but the sensitivity (identifying right documents of the cluster) observed for t-CUS is more robust than the CCMARDC. The computational complexity observed for t-CUS is linear and comparatively much lower than the CCMARDC. The observations learnt from the experiments leads our future work to determine the utility scale for other two factors of the text documents called context and semantic relevancy. In other direction, the concept utility scale can be used as fitness function in genetic algorithm to cluster the documents.

SIGNIFICATION STATEMENTS

This manuscript explores the scope of research to depict novel text clustering in regard to conceptual relevance of the text data. Profoundly the critical objective of the contribution is to depict a concept relevance based clustering model for text data that depends on the knowledge base to estimate the concept scope. The other significant theme of the manuscript is to use the statistical assessment strategy to select optimal features that are using in clustering process.

REFERENCES

1. Dixit, D.N. and R.K. Gupta, 2014. Study of recent advancement in document clustering. Int. J. Adv. Res. Comput. Sci., 5: 181-185.
2. Novak, P.K., N. Lavrac and G.I. Webb, 2011. Supervised Descriptive Rule Induction. In: Encyclopedia of Machine Learning, Sammut, C. and G.I. Webb (Eds.). Springer, USA., ISBN: 978-0-387-30768-8, pp: 938-941.
3. Everitt, B., 2012. Introduction to Optimization Methods and their Application in Statistics. Springer Science and Business Media, USA., ISBN: 9789400931534, Pages: 88.
4. Kowalski, G.J. and M.T. Maybury, 2006. Information Storage and Retrieval Systems: Theory and Implementation. 2nd Edn., Springer Science and Business Media, USA., ISBN: 9780306470318, Pages: 318.
5. Antoniou, D., Y. Plegas, A. Tsakalidis, G. Tzimas and E. Viennas, 2012. Dynamic refinement of search engines results utilizing the user intervention. J. Syst. Software, 85: 1577-1587.
6. Carpineto, C., M. D'Amico and G. Romano, 2012. Evaluating subtopic retrieval methods: Clustering versus diversification of search results. Inform. Process. Manage., 48: 358-373.
7. Sahami, M. and T.D. Heilman, 2010. Generating query suggestions using contextual information. U.S. Patent No. 7725485, U.S. Patent and Trademark Office, Washington, DC. https://www.google.com/patents/US8209347
8. Aggarwal, C.C. and C.X. Zhai, 2012. A Survey of Text Clustering Algorithms. In: Mining Text Data, Charu, C.A. and X.Z. Cheng (Eds.). Springer, New York, USA., ISBN:978-1-4614-3222-7, pp: 77-128.
9. Liu, D., X. Yang and M. Jiang, 2013. A novel text classifier based on quantum computation. Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, August 5-7, 2013, Sofia, Bulgaria, pp: 484-488.
10. Iezzi, D.F., 2012. Centrality measures for text clustering. Commun. Stat.-Theory Methods, 41: 3179-3197.
11. Meina, M. and H.S. Nguyen, 2015. Search result clustering based on query context. Fund. Inform., 137: 273-290.
12. Li, F. and Q. Zhu, 2011. Dcoument clustering in research literature based on NMF and testor theory. J. Software, 6: 78-82.
13. Kumar, N. and K. Srinathan, 2009. A new approach for clustering variable length documents. Proceedings of the IEEE International Advance Computing Conference, March 6-7, 2009, Patiala, India, pp: 982-987.
14. Luo, C., Y. Li and S.M. Chung, 2009. Text document clustering based on neighbors. Data Knowl. Eng., 68: 1271-1288.
15. Ni, X., X. Quan, Z. Lu, L. Wenyin and B. Hua, 2011. Short text clustering by finding core terms. Knowl. Inform. Syst., 27: 345-365.
16. Bharathi, G. and D. Vengatesan, 2012. Improving information retrieval using document clusters and semantic synonym extraction. J. Theor. Applied Inform. Technol., 36: 167-173.
17. Pessiot, J.F., Y.M. Kim, M.R. Amini and P. Gallinari, 2010. Improving document clustering in a learned concept space. Inform. Proc. Manage., 46: 180-192.
18. Li, Y., S.M. Chung and J.D. Holt, 2008. Text document clustering based on frequent word meaning sequences. Data Knowledge Eng., 64: 381-404.
19. Bollegala, D., Y. Matsuo and M. Ishizuka, 2011. A web search engine-based approach to measure semantic similarity between words. IEEE Trans. Knowl. Data Eng., 23: 977-990.
20. Kaiser, F., H. Schwarz and M. Jakob, 2009. Using Wikipedia-based conceptual contexts to calculate document similarity. Proceedings of the 3rd International Conference on Digital Society, February 1-7, 2009, Cancun, Mexico, pp: 322-327.
21. Shehata, S., F. Karray and M. Kamel, 2010. An efficient concept-based mining model for enhancing text clustering. IEEE Trans. Knowl. Data Eng., 22: 1360-1371.

22. Baghel, R. and D.R. Dhir, 2010. A frequent concepts based document clustering algorithm. Int. J. Comput. Appl., 4: 6-12.

23. Bhattacharjee, A.D., 2016. Feature Extraction. In: Intelligent Analysis of Multimedia Information, Bhattacharyya, S., S. Bhattacharyya, H. Bhaumik, S. De and G. Klepac (Eds.). IGI Global, USA., ISBN: 9781522504993, pp: 48-105.

24. Huang, R. and W. Lam, 2009. An active learning framework for semi-supervised document clustering with language modeling. Data Knowl. Eng., 68: 49-67.

25. Wang, F. and C. Zaniolo, 2008. Temporal queries and version management in XML-based document archives. Data Knowl. Eng., 65: 304-324.

26. Chehreghani, M.H., H. Abolhassani and M.H. Chehreghani, 2008. Improving density-based methods for hierarchical clustering of web pages. Data Knowl. Eng., 67: 30-50.

27. Algergawy, A., E. Schallehn and G. Saake, 2009. Improving XML schema matching performance using Prufer sequences. Data Knowl. Eng., 68: 728-747.

28. Delibasic, B., M. Vukicevic, M. Jovanovic, K. Kirchner, J. Ruhland and M. Suknovic, 2012. An architecture for component-based design of representative-based clustering algorithms. Data Knowl. Eng., 75: 78-98.

29. Zhang, T., Y.Y. Tang, B. Fang and Y. Xiang, 2012. Document clustering in correlation similarity measure space. IEEE Trans. Knowl. Data Eng., 24: 1002-1013.

30. Hu, X., X. Zhang, C. Lu and X. Zhou, 2009. Exploiting wikipedia as external knowledge for document clustering. Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, June 28-July 1, 2009, Paris, France, pp: 389-396.

31. Prathima, Y. and K.P. Supreethi, 2011. A survey paper on concept based text clustering. Int. J. Res. IT Manage., 1: 45-60.

32. Somprasertsri, G. and P. Lalitrojwong, 2010. Mining feature-opinion in online customer reviews for opinion summarization. J. Universal Comput. Sci., 16: 938-955.

33. Egozi, O., S. Markovitch and E. Gabrilovich, 2011. Concept-based information retrieval using explicit semantic analysis. ACM Trans. Inform. Syst., Vol. 29. 10.1145/1961209.1961211.

34. Skabar, A. and K. Abdalgader, 2013. Clustering sentence-level text using a novel fuzzy relational clustering algorithm. IEEE Trans. Knowl. Data Eng., 25: 62-75.

35. Jing, L., M.K. Ng and J.Z. Huang, 2010. Knowledge-based vector space model for text clustering. Knowl. Inform. Syst., 25: 35-55.

36. Margara, A., J. Urbani, F. van Harmelen and H. Bal, 2014. Streaming the web: Reasoning over dynamic data. Web Semantics: Sci. Serv. Agents World Wide Web, 25: 24-44.

37. Nikhath, A.K. and K. Subrahmanyam, 2016. Incremental evolutionary genetic algorithm based optimal document clustering (ODC). J. Theor. Applied Inform. Technol., 87: 479-488.

38. Kim, Y.Z., 2014. Credibility adjusted term frequency: A supervised term weighting scheme for sentiment analysis and text classification. Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, June 27, 2014, Baltimore, Maryland, USA., pp: 79-83.

39. Jayabharathy, J. and S. Kanmani, 2014. Correlated concept based dynamic document clustering algorithms for newsgroups and scientific literature. Decision Anal., Vol. 1. 10.1186/2193-8636-1-3.

40. Shehata, S., 2009. A WordNet-based semantic model for enhancing text clustering. Proceedings of the IEEE International Conference on Data Mining Workshops, December 6, 2009, Miami, FL., pp: 477-482.

41. Nikhath, A.K. and K. Subrahmanyam, 2017. COCUS: Concept based document clustering by corpus utility scale. Indian J. Sci. Technol., Vol. 10. 10.17485/ijst/2017/ v10i10/ 104236.

42. Larose, D.T., 2011. Discovering Statistics. Macmillan Higher Education, USA.

43. Washington, S.P., M.G. Karlaftis and F.L. Mannering, 2010. Statistical and Econometric Methods for Transportation Data Analysis. 2nd Edn., CRC Press, USA., ISBN: 9781420082852, Pages: 544.

44. David, M., M. Dzamba, D. Lister, L. Ilie and M. Brudno, 2011. SHRiMP2: Sensitive yet practical short read mapping. Bioinformatics, 27: 1011-1012.

45. Wang, Y., Y. Tong and M. Zeng, 2013. Ranking scientific articles by exploiting citations, authors, journals and time information. Proceedings of the 27th AAAI Conference on Artificial Intelligence, July 14-18, 2013, Bellevue, Washington, DC.

46. Ihaka, R. and R. Gentleman, 1996. R: A language for data analysis and graphics. J. Comput. Graphical Stat., 5: 299-314.