ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 10(10), DOI: 10.17485/ijst/2017/v10i10/104236, March 2017* ISSN (Online) : 0974-5645

COCUS: Concept Based Document Clustering by Corpus Utility Scale

A. K. Nikhath and K. Subrahmanyam

Department of Computer Science and Engineering, Koneru Lakshamaiah Education Foundation, Guntur– 522502, Andhra Pradesh, India; kousarnikhath@vnrvjiet.in, smdkodukula@kluniversity.in

Abstract

Objective: With the rising quantum of documents in corpuses, it is very important that data management and data assurance is with high interoperability towards retrieving the critical documents from vast range of services. By focusing on the semantic features, which could improve the level of accuracy in document tracing and retrieval, the issues and limitations in the present models could be addressed in an effective manner. **Methods/Statistical Analysis:** In this study, focus is on depicting the robustness of semantic features based clustering techniques and its efficacy, compared to the other kind of clustering techniques. This paper proposed a concept based document clustering by corpus utility scale (COCUS) proposed. The utility scale proposed in COCUS is derived with support of topic related selected document set as knowledge base that enables to cluster the documents by their concept relevancy. The proposed clustering model is assessed through the state of the art metrics called cluster purity, inverse of purity and cluster level harmonic mean. Experiments were carried out on datasets that comprise the containing specific kind of literature gathered from varied open access journals from publishers. The total 1509 number of documents was collected and among them 497 documents was used as knowledgebase and rest 1012 documents were used for clustering process. **Findings:** The experimental study evincing that the proposed model is scalable and robust. The purity and harmonic mean of the resultant clusters confirming that the COCUS clusters the documents by their concept relevancy with 94% accuracy (Average of the topic level harmonic mean of the clusters was found as 0.94). **Application/ Improvements:** The computational complexity of the COCUS is evinced as linear, where the majority of benchmarking models are found to be np-hard.

Keywords: Cluster, Corpus Utility, Harmonic Mean, Text Mining

1. Introduction

The increasing sources in terms of data available from vivid cradles like the online sources, physical documents, documents from the internal sources in the organizational systems network and other such factors are turning out to be potential stream that shall support the companies in effectively navigating and managing with the information essential for the specific tasks or information systems.

Fast and effective levels of high-quality clustering algorithms are gaining prominence in the case scenarios of intuitive navigation and also the browsing mechanism which is need for managing the high volumes of information in to meaningful clusters that shall support in managing the performance of retrieval and also in terms of developing small clusters. This could be feasible by dimensional reduction, term-weighting[1], and by query expansion[2].

In the case scenario of today's search engines, predominantly the string matching process is adapted which may not be effective for documents retrieval and may not even be apt to any kind of user queries. It is very essential to focus upon developing an effective cluster for the document corpus, which could be handled by teams for more efficient way of browsing and navigation. Also, the system should address the kind of deficiencies associated with the existing information retrieval methods.

There are many elements like the process of information processing, text mining and such related areas in which the document clustering could be envisaged. Though, information retrieval systems were catering as an effective medium for evaluating the neighbor of a document[3-5]. On

* *Author for correspondence*

the basis of the inputs gathered, the clustering is used for collection of documents and towards organizing the results that are envisaged in the search engine, whilst responding to the queries from the users[6-8].

Document clustering process has been vividly used by the companies even in the process of generating hierarchical clusters too[9], and generally the search engine offers wide list for broad queries as a result of the query processing that is carried out. , thus resulting in how the users browse or identify the kind of relevant information. Whereas, such issues can be easily handled by focusing on the clustering methods that can handle the process of developing meaningful categories in the process of Enterprise Search Engines[10]. But one of the significant issues that have to be considered in the process is about the scalability, as the rising number of documents demands the dynamically responsive formulated clusters.

In the process of document clustering in dynamic mode, the quantum of time and effort essential for document clustering shall reduce significantly, as the new algorithms handle the process more effectively support in document re-clustering process for the documents in the corpus. In the case scenario of some of the document clustering process like constituting the terms[11] or Synonyms and Hypernyms[12] may not be offering quality outcome in a dynamic environment, whilst addressing the technically related elements.

Considering such limitations, it is of paramount importance that there is need for more effective dynamic document clustering process that has more emphasis on the frequency and correlated terms adapted as significant factors. Such facets are considered in the proposed paper with the focus on scientific literature and newsgroup data sets which form the crux of data evaluated. Referred as Concept based Document Clustering by Corpus Utility Scale, which is a document clustering model, is the model proposed which could address numerous issues depicted in the case scenario?

Proposed solution is predominantly assessed using benchmarking metrics called cluster purity, inverse of purity and harmonic mean. The set of scientific literature documents are used as input to the proposed clustering algorithm and the resulting clusters has been depicted.

Among the varied range of document clustering methods that has been adapted in the VSM (Vector Space Model) is widely adapted for depicting the data and also in the instances of classifying the cluster solution[13]. In the case of VSM, some of the intrinsic factors that have to be taken in to account is the consideration of key feature vectors for the terms in each of the document. In the case scenario of some kind of weight used towards statistical measure –like Term Frequency-Inverse Document Frequency, it is important to focus on identifying the significance of keyword in document collection and corpus[14, 15].

With the emerging volume of keyword numbers in the search process, the relevance of a word is aware of; still the frequency of the word could be the offset in the corpus[16]. The similarity of the documents is analyzed using the feature vector model[17] and the prerogative whilst carrying out such process is Cosine and Jaccard Measure to be considered.

The model[18] is the method of document clustering by focusing on NMF (Negative Matrix Factorization) method for discovering which is reliant on project conditions test. In such methods, the key element that is targeted is about reducing the dimensionality for the text data and also the process of clustering them. In the NMF model, mainly the keywords are adapted in the search process. The data is compared using the learning matrix and the comparison matrix.

In accordance to the above depictions, evaluation of a case study pertaining to automatic classification represented in a Chinese conference proceedings were evaluated and the outcome from such process has been phenomenal as the proposed method of NMF has provided quality results.

Among the varied kind of document clustering algorithms that are adapted for term frequency[19-21] researchers are advocating on the clustering based synonyms and hypernyms[22-28].

Numerous document clustering algorithms from the recent literature was reviewed. It is imperative from the observation that the similarity relationship is profoundly on the sentences used[29,30], emphasis on the kind of tree representation and the similarity relationship that are usually identified and also on the basis of the clustered outcome[31-33]. Also, the scope of varied components based clustering which makes use of object-based representation is also considered, alongside the process of modeling the document and cosine, which is an integral part of document clustering[34].

The other certain factors that are considered in the process is about the adaptation of semantic relations and

the documents which could be critical on the relative levels of Terms and Related Terms[35, 36]. Predominantly majority of the works that are performed on the web page information has been carried out on the basis of representation, tracking and retrieval that could support in improving the services. The study[37] reflected upon the concept of clustering algorithm method and emphasized that use of TF-IFD techniques and solutions can lead to following impact

- Issues of consideration towards synonyms and also the polysemous are some of the issues that shall significantly impact the process.
- The need of differentiating the each term has not been successful in terms of finding the degree of semantic importance.
- Towards attaining the weight in a document internally, for words that is semantically important and not so important, without any kind of ascertaining is the other key factor.

Our earlier contribution called ODC is aimed to optimize the clusters using evolutionary computation approach called Genetic Algorithm. Though the model emerged as significant to optimize the clusters, the search space usage and computational complexity is evinced as complement to optimize large number of clusters.

From the critical review of literature, it is very important that many of the clustering techniques that depicted are on fundamentals of term frequencies, some of them are based on annotation tools that rely upon synonyms and hypernyms. WordNet lexical database in[38] is adapted in the process of extracting synonyms and hypernyms, in the instances of tracking any new set of documents, mainly the domain-specific technical terms shall be much effective. Also, the method of focusing on synonyms and hypernyms is not being result oriented, and needs considerable improvement. On other dimension of the constraints, all of these existing models that include our earlier contribtuion40 are compliment in search space usage and computational process.

With the objective of enhancing the quality of the clusters for topic specific document sets with linear complexity in search space utilization and computational process, this paper focus on model which clusters the documents using the utility scale which is discovered on the basis of given topic related knowledgebase. In the proposed model, using such knowledgebase the cluster labels are identified on the basis of concept relations.

2. Concept Based Document Clustering by Corpus Utility Scale

The proposed model is discovering the concept compatibility of the documents of the given corpus by the corpus utility scale. In regard to this the proposed model discovers the corpus utility scale (see sec 3.1), further this scale will be used to form the class labels under concept relation and afterwards, clusters the documents based on these class labels.

2.1 Corpus Level 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 corpus utility scale that depends on the set of documents as knowledge base that selected under supervised learning process. The documents to be clustered are considered as corpus. The proposed process of defining the corpus utility scale initially discovers the knowledgebase level occurrence of each term that exists in the documents found in respective knowledgebase. Further document level occurrence of each term will be assessed, which is the number of times the respective term appears in a document of respective corpus. Afterwards the document level utility of each term will be assessed, which is based on the knowledgebase level term occurrence and document level term occurrence. Then the document level utility of a given term-set is 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 the 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. Also, the Corpus Utility is also evaluated as the cumulative of the document utility of all documents that is constituted in the corpus. Under the domain contexts, the outcome of utility threshold could be varying between 0 and 1, and the corpus utility, results the corpus utility scale.

Knowledge Base (kb): Set of documents selected from domain expert's recommendation or prototype documents of the concept.

KB level term occurrence ($o_{kb}(t)$): The quantum of documents in kb comprise the respective term t

Document Level term Occurrence ($o_{d_i}(t)$): The number of times the term t appears in respective document of the corpus

Document Level Term Utility (tu): The product of term occurrence at knowledge base and respective corpus level document (knowledge-Base Level Occurrence*Document Level Occurrence).

$$u(t,d_i) = o_{kb}(t) \times o_{d_i}(t)$$

Document level utility of Term-set (dtu): the aggregate of the term utility of terms exists in given term set.

$$dtu(ts,d_k) = \sum_{i=1}^{|ts|} \{u(t_i,d_k) \exists t_i \in ts \wedge ts \in d_k\}$$

Corpus Level term-set Utility (ctu): the aggregate of document level term-set utility of each document.

$$ctu(ts) = \sum_{i=1}^{|Cor|} \{dtu(ts,d_i) \exists d_i \in Cor\}$$

Document Utility (du): The aggregate of the Term-set utilities of term-sets exists in that document

$$du(d_k) = \sum_{i=1}^{|TS|} \{dtu(ts_i,d_k) \exists ts_i \in d_k\}$$

Corpus Utility (cu): The aggregate of document utilities of documents exists in given corpus

$$cu(Cor) = \sum_{i=1}^{|Cor|} \{du(d_i) \exists d_i \in Cor\}$$

Utility scale (su): corpus utility * utility threshold // here the utility threshold given under the context which is greater than 0 and less than 1

$$su = cu(Cor) \times \{\upsilon \exists 0 < \upsilon < 1\}$$

2.2 Labels by Utility Scale

A term-set is said to be the label having significance in semi-supervised learning iff the corpus level utility of that term-set is greater than the utility scale, which are determined as follows:

The term sets of size 1 *to* n will be find in the respective order. The set of terms having corpus level utility greater than the utility scale are considered to be as a term set. In order to this initially term sets of size 1 will be obtained. Further, the union t_{pq} of each pair of term sets t_p and t_q found in $TS(i-1)$ will be placed in to $TS(i)$ such that iff the corpus level utility of the respective term set t_{pq} is greater than the utility score. Further the transactions list $LST(t_{pq})$ of respective term set will be prepared by intersecting the transaction lists $LST(t_p)$ and $LST(t_q)$ of the respective input term sets t_p and t_q. This continues till there is no new term sets discovered.

Term-Sets (TS) Discovery:

Find $TS(1)$:

$LST(t)$ // is a set contains all the transactions having term set t

For each term as a set t begin

Find corpus level utility of the term-set t is greater than the Utility Scale or not, if true move t to $TS(1)$ and create a list $LST(t)$ that contains all the documents with term set t.

End

$Let\ n = 1$

Repeat:

$n{+}{+}$;

$TS(n) = \phi$ // empty n^{th} term-sets

1. For each term-set t_p that exists in $TS(n-1)$ Begin

2. For each Term-set t_q that exists in $TS(n-1)$ and $t_p \neq t_q$ Begin

a. if

$$\left(|LST(t_p) \cap LST(t_q)| \neq 0 \;\&\&\; \left(\sum_{i=1}^{|LST(t_p) \cap LST(t_q)|} ctu(d_i \exists d_i \in LST(t_p) \wedge d_i \in LST(t_q) \right) \geq su \right) \text{begin}$$

$t_{pq} \leftarrow t_p \cup t_q$

$LST(t_{pq}) = LST(t_p) \cap LST(t_q)$

$TS(n) \leftarrow t_{pq}$

End // if in line a

End //for each in line 2

End //for each in line 1

UNTIL $TS(n)$ is not empty

n=n-1;

2.3 Cluster Formation

The resultant term sets and respective document lists of those term sets will be processed to discover the final clusters as follows. For each term set t_p, find the other term sets that are having t_p as subset, then prune the documents from $LST(t_p)$, which are existing in the any of the documents list of super sets discovered for t_p. Further, if $LST(t_p)$ is empty then discard t_p and respective documents list $LST(t_p)$. Finally the document lists of resultant term

sets are the clusters and respective term-sets are the labels of the respective clusters. The algorithmic representation of the cluster formation is as follows:

1. $\forall_{i=1}^{n}\{TS(i)\exists \mid TS(i)\mid > 0\}$ Begin

   a. $TS \leftarrow TS(i)$ // moving all term sets in to single list TS

2. End //of 1

3. $\forall_{p=1}^{|TS|}\{t_p \exists t_p \in TS\}$ Begin

4. $\overline{TS} = sortDescending(TS)$ // $\overline{TS}$ is a set contains all the term-sets of TS in descending order of their size.

5. $\forall_{q=1}^{|\overline{TS}|}\{t_q \exists t_q \in \overline{TS} \wedge t_p \neq t_q\}$ Begin

6. If $\left(t_p \subseteq t_q\right)$ begin

   a. $LST(t_p) = LST(t_p) \setminus \left(LST(t_p) \cap LST(t_q)\right)$ // pruning the documents list entries of the t_p, which are exists in documents list of t_q.

   b. If $LST(t_p)$ is empty begin

      i. $TS \leftarrow TS \setminus t_p$ // discard t_p from TS

      ii. continue to step 3

   c. End //of if b

7. End //of if 5

8. End // of line 4

9. End // of line 3

The respective documents lists of the term-sets retained in TS are confirmed to be the final clusters and the respective term sets as labels.

3. Experimental Study and Performance Analysis

3.1 The Dataset

The model devised in this manuscript is a semi supervised document clustering by corpus utility scale. The corpus utility scale proposed is the influence of utility scale often used in utility based frequent item set mining. The utility scope of the given input data is derived from the given support documents as corpus. The accuracy and robustness of the proposal is explored through the cluster evaluation metrics called cluster purity, inverse of purity, harmonic mean of the clusters, harmonic-mean of the topics, and computational complexity of the clustering process. In order to this set of recommended documents has been selected under the topic context as knowledge base, and the documents to be clustered as corpus, which have been grouped by topic to facilitate the cluster optimality evaluation by the said metrics.

3.2 Assessment Metrics and Strategy

The metrics purity, inverse of purity and harmonic mean are significant for cluster evaluation. The frequency of a category in all resultant clusters is referred as cluster purity[39] Let C be the set of resultant clusters of a given document clustering process, Let L be the set of labeled documents of reference distribution and N be the total documents used as input to the clustering process, then the cluster purity is computed as follows:

$$pu = \sum_{i=1}^{|C|} \frac{|c_i|}{N} \times \max\left(P_{c_i}\right)$$

Here pu is purity, C is the set of clusters formed, $|C|$ is total number of clusters, c_i is i^{th} cluster, P_{c_i} is the set contains the precision observed between c_i and all the categories in L. $\max(P_{c_i})$ is the max value of the precisions found in P_{c_i}.

The precision of a cluster c_i for a given category l_j is defined as:

$$pr(c_i, l_j) = \frac{|c_i \cap l_j|}{|c_i|}$$

The metric purity nullifies the noise in corresponding cluster, but not capable to discover the togetherness of the documents, that is if each document is considered as one cluster then it results clusters with maximal purity. Hence the metric called inverse of purity is significant to identify the scope of the documents of a cluster are of the same category. The metric inverse of purity is keen to discover the cluster with maximum recall for each category, which is estimated as follows:

$$pu_{inv} = \sum_{i=1}^{|L|} \frac{|l_i|}{N} \times \max(pr_{l_i})$$

Here pu_{inv} is the inverse of purity, L is the set of categories, pr_{l_i} is the set contains precision of category l_i with all clusters in C and $\max(P_{l_i})$ is the max value of the precisions exists in pr_{l_i}.

Defining a cluster that contains all input documents results maximum value for inverse of purity, since it this metric is unable to nullify the mix of documents from

different categories. Hence it is obvious to consider the harmonic mean of the clusters along the side of purity and inverse of purity. The harmonic mean of the cluster is the combination of purity and inverse of purity, which is estimated by comparing each category with the cluster that has a highest combined precision and recall[9, 40, and 41], which is also known as F-Measure:

$$F = \sum_{i=1}^{|L|} \frac{|l_i|}{N} \times \max(F_{l_i})$$

Here F is the F-Measure and F_{l_i} is a set contains the f-measure found between category l_i and each cluster found in C. The notation $\max(F_{l_i})$ is the max value found in the set F_{l_i}. The f-measure between a category l_i and cluster c_j can be measured as follows:

$$F(l_i, c_j) = \frac{2 \times pr(l_i, c_j) \times rc(l_i, c_j)}{pr(l_i, c_j) + rc(l_i, c_j)}$$

Here the notation $rc(l_i, c_j)$ represents the recall observed for category l_i and cluster c_j, which is identically equal to the precision $pr(c_j, l_i)$ of cluster c_j to category l_i.

The adopted model is a search strategy, which is often complexed towards process and resource utilization. Hence the time complexity and process complexity of the proposed algorithm also being assessed.

3.3 Experimental Study

The implementation of the proposed model is done using java 8 on a computer with i5 processor, 4GB ram and Nvidia 4GB graphics card[42] used. The assessment of the metrics on the resultant cluster is done by scripts defined in R programming language[43]. The statistics of the experimental study are explored in Table 1.

The corpus size (number of documents) is 1012, which is partitioned in to 24 topic groups. The knowledge base is the set of topic based selected documents of size 492. The number of clusters formed by the proposed model are 26 and among them 21 clusters highly optimal, which is depicted in Figure 1 - 2, since their harmonic mean is more than the average of harmonic means observed for all the clusters and 4 clusters are significant to consider, since their harmonic mean is approximately equal to the average of harmonic means observed for all clusters.

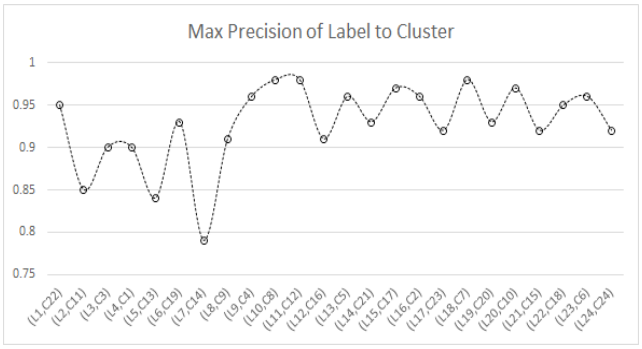

Figure 1. Max precision observed for labels to clusters.

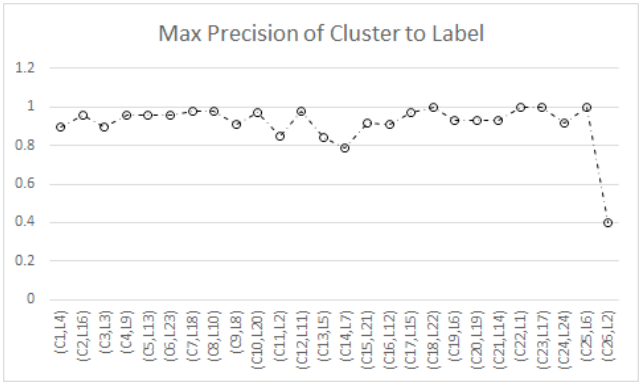

Figure 2. Max Precision observed for cluster to label.

Table 1. Input and observed metric values from the experiments

Total Number of Documents	**Knowledge Base: 492, Topiced Corpus: 1012**
Total Number of Clusters from the Documents	24
Total Number of clusters formed by COCUS	26
Purity of the Clusters	0.95
Purity of the Topics (Inverse of cluster level purity	0.94
The Number of Clusters with optimal harmonic mean (greater than the average of the cluster level harmonic means)	21
The Number of Topics with optimal harmonic mean (greater than the average of the topic level harmonic means)	15
The average of Cluster Level harmonic mean	0.92
The average of Topic Level harmonic mean	0.94
Clusters to Topic Correlation (correlation between cluster purity and inverse of purity)	0.999993
Clusters to Topic Covariance (Covariance between cluster purity and inverse of purity)	0.000175642

One cluster is found to be insignificant with fewer numbers of documents and the harmonic mean of that cluster is drastically low. The values obtained for metrics such as cluster purity that depicted in Figure 3, inverse of purity depicted in Figure 4, harmonic mean depicted in Figure 5 and Figure 6, and computational complexity that depicted in Figure 7 indicating that document clustering by COCUS is scalable and robust since the cluster purity (0.95) and inverse of purity (0.94) are high and the average of the harmonic means of the both clusters (0.92) and topics (0.94) at its best. The correlation and covariance between cluster purity and inverse purity also found at their best (correlation: 0.999993, covariance: 0.000175642).

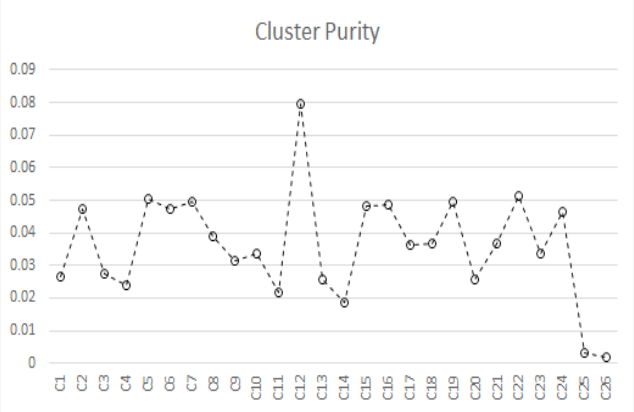

Figure 3. Each Clusters Purity obtained.

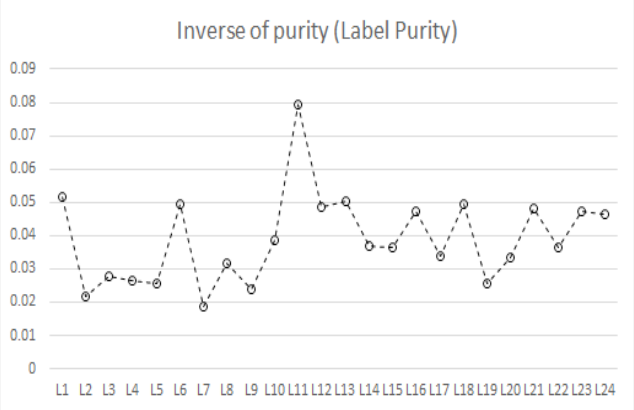

Figure 4. Inverse of Purity (purity obtained for each label).

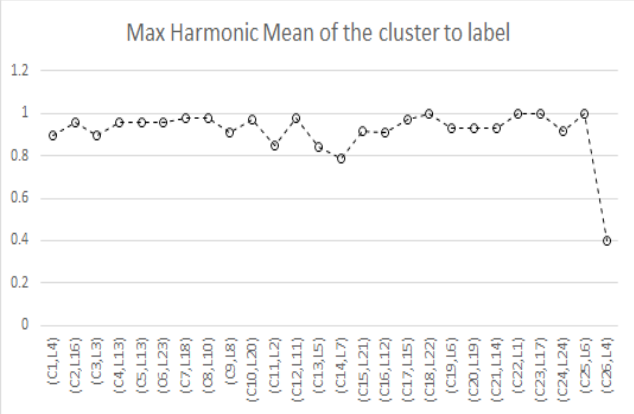

Figure 5. Cluster and label pairing by max harmonic mean.

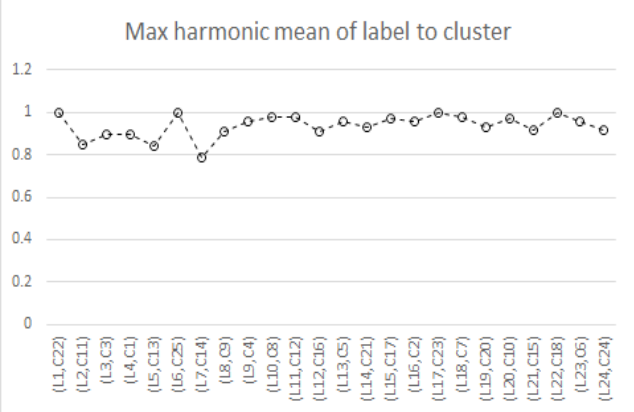

Figure 6. Label to cluster pairing by max harmonic mean.

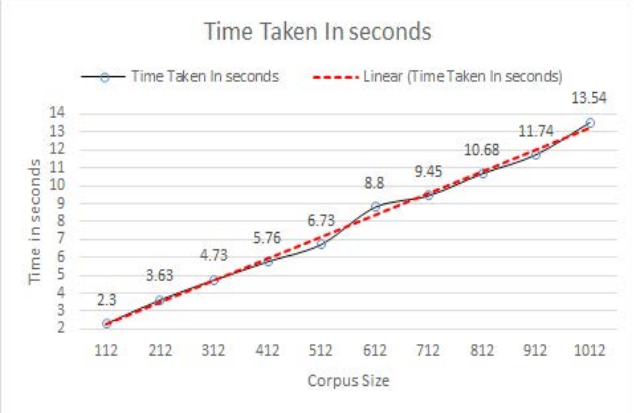

Figure 7. Process time observed for divergent count of documents as input corpus.

The process time that depicted in figure 7 has taken for divergent sizes of the corpus are evinced as linear. Since the increment in number of iterations occurred in uniform manner, which is observed as

$$itr_s = 2*(itr_{s-1}) - (itr_{s-2}) + \lambda + (s-3)*2$$

The above notation is addressing the correlation between the iterations found for the corpus size with the iterations found for corpus size $s-1$ *and* $s-2$. Here in theabove equation itr_{s-1} is the count of iterations found for corpus size $s-1$ and itr_{s-2} is count of iterations found for corpus size $s-2$, the notation λ is a constant that observed as 7 in our experiments.

4. Conclusion

This manuscript presented Concept based Document Clustering by Corpus Utility Scale (COCUS). The proposed model introduced the term weighing factor called corpus utility scale, which is estimated with the support of given concept referral documents as knowledge base. In order to estimate the corpus level utility, set of utility factors such as (i) knowledge base level term occurrence, (ii) Document level term occurrence (iii) document level term utility, (iv) document level utility of term set, (v) corpus level utility of term sets, (vi) document utility and (vii) corpus utility were introduced. The proposed model is semi supervised learning model that discovers optimal correlated class labels using corpus level utility scale and further clusters the documents based on these class labels. The process of class label discovery and cluster formation are performs together. The class label discovery is defined with influence of utility based frequent item set mining. The performance analysis of the COCUS is done using metrics such as (i) cluster purity, (ii) inverse of purity and (iii) harmonic mean (also known as f-measure) of the clusters. The results obtained from experimental study evincing that the COCUS is scalable and robust and results the optimal clusters from the given corpus, which is notified through the purity and harmonic mean of the clusters. The motivation gained from this proposal that is utility scale for concept relevancy, 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 evolutionary strategies such as genetic algorithm can be used to cluster the documents that use the corpus utility scale as fitness function.

5. References

1. Tang B, Shepherd M, Milios E, Heywood MI. Comparing and combining dimension reduction techniques for efficient text clustering. In Proceeding of SIAM International Workshop on Feature Selection for Data Mining 2005. 17–26 .
2. Sammut C, Webb GI, editors. Encyclopedia of machine learning. Springer Science and Business Media. 2011 Mar 28.
3. Everitt B. Introduction to optimization methods and their application in statistics. Springer Science and Business Media. 2012 Dec 6.
4. Kowalski GJ, Maybury MT. Information storage and retrieval systems: Theory and implementation. Springer Science and Business Media. 2006 Apr 11.
5. Buckley C, Lewit AF. Optimization of inverted vector searches. InProceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval. 1985 Jun 5; 97–110. Available from: Crossref
6. Cutting DR, Karger DR, Pedersen JO, Tukey JW. Scatter/gather: A cluster-based approach to browsing large document collections. In Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval. 1992 Jun 1; 318–29.
7. Zamir O, Etzioni O, Madani O, Karp RM. Fast and Intuitive Clustering of Web Documents. KDD. 1997 Aug 14; 97: 287–90.
8. Delafrooz N, Farzanfar E. Determining the customer lifetime value based on the benefit clustering in the insurance industry. Indian Journal of science and Technology. 2016 Jan 7;9(1):1–8.
9. Steinbach M, Karypis G, Kumar V. A comparison of document clustering techniques. In KDD workshop on text mining. 2000 Aug 20; 400(1): 525–6.
10. Andrews NO, Fox EA. Recent developments in document clustering. Technical report, Computer Science, Virginia Tech. 2007 Oct 16.
11. Wang X, Tang J, Liu H. Document clustering via matrix representation. In 2011 IEEE 11th International Conference on Data Mining. 2011 Dec 11; 804–13. IEEE. Available from: Crossref
12. Nadig R, Ramanand J, Bhattacharyya P. Automatic evaluation of word net synonyms and hyponyms. In Proceedings of ICON-2008: 6th International Conference on Natural Language Processing. 2008;831.
13. Aas K, Eikvil L. Text categorisation: A survey. Technical Report 941. Oslo Norway: Norwegian Computing Center. 1999.
14. Salton G, Buckley C. Term-weighting approaches in automatic text retrieval. Information processing and management. 1988 Dec 31; 24(5): 513–23.
15. Sajana T, Rani CS, Narayana KV. A Survey on Clustering Techniques for Big Data Mining. Indian Journal of Science and Technology. 2016 Feb 8; 9(3): 1–12.

16. Mishra SP, Mishra D, Patnaik S. An empirical analysis on effect of data expansion for clustering low dimensional data. Indian Journal of Science and Technology. 2016 Feb 9;9(3): 1–21.
17. Huang A. Similarity measures for text document clustering. In Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand. 2008 Apr 14; 49–56.
18. Li F, Zhu Q. Dcoument Clustering in Research Literature Based on NMF and Testor Theory. Journal of Software. 2011 Jan 1;6(1): 78–82.
19. Kumar N, Srinathan K. A New Approach for Clustering Variable Length Documents. In Advance Computing Conference, 2009. IACC 2009. IEEE International 2009 Mar 6; 982–7. Available from: Crossref
20. Luo C, Li Y, Chung SM. Text document clustering based on neighbors. Data and Knowledge Engineering. 2009 Nov 30; 68(11):1271–88. Available from: Crossref
21. Ni X, Quan X, Wenyin, L. Short text clustering by finding core terms. Journal of Knowledge and Information Systems, Springer Link. 2010; 27(3): 345–65
22. Bharathi G, Vengatesan D. Improving information retrieval using document clusters and semantic synonym extraction. Journal of Theoretical and Applied Information Technology. 2012 Feb; 36(2): 167–73.
23. Pessiot JF, Kim YM, Amini MR, Gallinari P. Improving document clustering in a learned concept space. Information processing and management. 2010 Mar 31; 46(2): 180–92.
24. Li Y, Chung SM, Holt JD. Text document clustering based on frequent word meaning sequences. Data and Knowledge Engineering. 2008 Jan 31; 64(1): 381–404.
25. Bollegala D, Matsuo Y, Ishizuka M. A web search engine-based approach to measure semantic similarity between words. IEEE Transactions on Knowledge and Data Engineering. 2011 Jul; 23(7): 977–90.
26. Kaiser F, Schwarz H, Jakob M. Using Wikipedia-based conceptual contexts to calculate document similarity. Third IEEE International Conference on Digital Society. 2009. ICDS'09. 2009 Feb 1; 322–7. Available from: Crossref
27. Shehata S, Karray F, Kamel M. An efficient concept-based mining model for enhancing text clustering. IEEE Transactions on Knowledge and Data Engineering. 2010 Oct; 22(10):1360–71.
28. Baghel R, Dhir R. A Frequent Concepts Based Document Clustering Algorithm. International Journal of Computer Applications. 2010 Jul; 4(5): 6–12.
29. Hammouda KM, Kamel MS. Efficient phrase-based document indexing for web document clustering. IEEE Transactions on knowledge and data engineering. 2004 Oct; 16(10): 1279–96.
30. Huang R, Lam W. An active learning framework for semi-supervised document clustering with language modeling. Data and Knowledge Engineering. 2009 Jan 31;68(1): 49–67.
31. Wang F, Zaniolo C. Temporal queries and version management in XML-based document archives. Data and Knowledge Engineering. 2008 May 31; 65(2): 304–24.
32. Chehreghani MH, Abolhassani H, Chehreghani MH. Improving density-based methods for hierarchical clustering of web pages. Data and Knowledge Engineering. 2008 Oct 31; 67(1): 30–50.
33. Algergawy A, Schallehn E, Saake G. Improving XML schema matching performance using Prüfer sequences. Data and Knowledge Engineering. 2009 Aug 31; 68(8): 728–47.
34. Delibašić B, Vukićević M, Jovanović M, Kirchner K, Ruhland J, Suknović M. An architecture for component-based design of representative-based clustering algorithms. Data and Knowledge Engineering. 2012 May 31; 75: 78–98.
35. Zhang T, Tang YY, Fang B, Xiang Y. Document clustering in correlation similarity measure space. IEEE Transactions on Knowledge and Data Engineering. 2012 Jun; 24(6): 1002–13.
36. Hu X, Zhang X, Lu C, Park EK, Zhou X. Exploiting Wikipedia as external knowledge for document clustering. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. 2009 Jun 28; 389–96. Crossref
37. Prathima Y, Supreethi KP. A survey paper on concept based text clustering. International Journal of Research in IT and Management. 2011;1(3): 45–60.
38. Miller GA. Word Net: a lexical database for English. Communications of the ACM. 1995 Nov 1; 38(11): 39–41.
39. Zhao Y, Karypis G. Criterion functions for document clustering: Experiments and analysis.
40. Van Rijsbergen CJ. Foundation of evaluation. Journal of Documentation. 1974 Apr 1; 30(4): 365–73.
41. Larsen B, Aone C. Fast and effective text mining using linear-time document clustering. In Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining 1999 Aug 1; 16–22. Available from: Crossref
42. Nikhath AK, Subrahmanyam K. Incremental Evolutionary Genetic Algorithm Based Optimal Document Clustering (ODC). Journal of Theoretical and Applied Information Technology. 2016 May 31; 87(3).
43. I haka R, Gentleman R. R: a language for data analysis and graphics. Journal of computational and graphical statistics. 1996 Sep 1; 5(3): 299–314.