

AN EFFICIENT INFORMATION FILTERING USING QUERY REORGANIZATION BASED SELF-ORGANIZING MAP METHOD

Dr. S. Arulselvarani

Assistant Professor, Department of Computer Science,
Ururu Dhanalakshmi College (Affiliated to Bharathidasan University),
Tiruchirappalli, Tamil Nadu, India.

ABSTRACT

Information at World Wide Web (WWW) is retrieved through queries that are submitted by the users to the search engine. For a given query, various techniques are followed to access the WWW to process it and retrieve the most related documents. All techniques retrieve more or less same documents, but vary in time that it consumes to generate such results. Therefore, the best method is needed to reduce the time required for information retrieval. This requirement is focused by enormous researchers to discover and invent many techniques. Though the invented techniques retrieved the documents quickly, they did not concentrate on the relevancy of retrieved documents. Hence, an essential technique that depletes less time and retrieves more accurate document is necessary. In this paper, a new clustering method has proposed for query re-organization in the information filtering of WWW.

Key words: World Wide Wide, Information Filtering, Clustering, Query Re-organization.

Cite this Article: S. Arulselvarani, An Efficient Information Filtering Using Query Reorganization Based Self-Organizing Map Method, *International Journal of Advanced Research in Engineering and Technology*, 11(12), 2020, pp. 3447-3460.
<https://iaeme.com/Home/issue/IJARET?Volume=11&Issue=12>

1. INTRODUCTION

Text clustering is a technique used to gather the documents which have similar content [1]. The main objective of text clustering is to divide the unstructured set of objects into clusters. The algorithm can be used to represent the concept and to measure the similarity among the concept present in the document. The clustering process is widely applied for summarizing the corpus and document classification [2]. Traditionally, quantitative data are focused for clustering, which contain numeric data as their attributes. Following this, categorical data are also studied where the attributes hold the nominal values. Nevertheless, these techniques do not work well for clustering the text data. Since the text data has the following unique properties, it requires a specialized algorithm for the task.

1. Dimensionality representation [3] of the text is very large, whereas the underlying data is sparse.
2. The total number of concepts in the data is much smaller than the feature space, which made the design of clustering algorithm very complex.
3. Normalization of document representation is required since the word count varies for different documents.

The high dimensional representation and the sparse nature of the documents require the design of text-specific algorithms for document representation and processing. Many existing clustering algorithms are used to improve the document representation for clustering. Usually, the vector space based Term Frequency - Inverse Document Frequency (TF-IDF) [4] representation is used for text clustering. In such type of representation, the Term Frequency (TF) [5] for each word is normalized by Inverse Document Frequency (IDF) [6]. In addition to the IDF, term-frequencies are appended with the sub-linear transformation function. This is carried out to avoid the undesirable dominating effect of any single term that might be frequent in the document. The clustering algorithms for text are widely classified into a range of different types. Partitioning algorithm, Agglomerative algorithm, and EM-algorithm are the different types of clustering algorithms. Different tradeoffs exist among the different clustering algorithms in terms of efficiency and effectiveness.

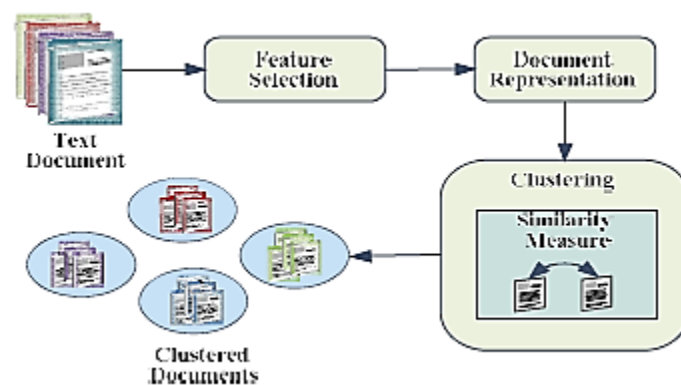


Figure 1 Overall Process of Text clustering for Query Reorganization

2. FEATURE SELECTION

Simple unsupervised methods [7] can also be used for feature selection in text clustering. Document Frequency-based selection, Term strength, Entropy-based ranking and term contribution are the various popular techniques that are used for feature selection. The descriptions of these techniques are presented in the following subsections.

2.1. Document Frequency - Based Selection

The simplest technique [8] for selecting the feature in the document clustering is that of the uses of document frequency to filter out the features that are irrelevant. The words that occur too frequently, i.e. stop words in the corpus, are removed since they are not discriminative from the clustering perspective. On the other side, the most infrequent words present in the text are also removed. In addition to this, noisy data are also removed. Some research scholars define the document frequency based feature selection as purely depending on the infrequent terms. This is due to the reason that these terms contribute the least to the similarity calculations. However, the words which are not discriminative for the clustering process should be removed.

2.2. Term Strength

This is the most aggressive technique for removing the stop words [9]. The strength of the term is computed to detect how informative a word is for identifying the documents that are related to each other. Consider the two related documents x and y , for which the term strength can be computed from the following probabilistic equation:

$$s(t) = P(t \in y | t \in x)$$

One major advantage of this technique is that there is no need of initial supervision or training data for the selection process.

2.3. Entropy - Based Ranking

The quality of the term in the document is measured through this technique [10]. The entropy of a term in the documents can be determined through the equation:

$$E(t) = - \sum_{i=1}^n \sum_{j=1}^n (X_{ij} \cdot \log(X_{ij}) + (1 - X_{ij}) \cdot \log(1 - X_{ij}))$$

In the above equation, $X_{ij} \in (0,1)$ shows the similarity among i^{th} and j^{th} document in the collection, after the term t is removed. The mathematical representation of X_{ij} is represented in equation

$$X_{ij} = 2^{-\frac{d(i,j)}{d}}$$

From the above equation has $d(i,j)$ which denotes the distance between the terms i and j after the term t is removed. The computation of $E(t)$ requires $O(n^2)$ operations. With this requirement, it becomes impractical to implement for the corpus holding many terms.

2.4. Term Contribution

This method depends on the fact that the clustering of texts is highly dependent on the similarity present in the document. Here, the term contribution is considered the contribution of document similarity [11].

3. FEATURE REPRESENTATION

Similar to feature selection, feature transformation [12] is a technique that improves the quality of the retrieving process. The transformation technique defines the new features as its functional representation of the features in the original data set. The most common method is dimensionality reduction. In dimensionality reduction, the features are transformed to a new space of smaller dimensionality [18][19][20][21][22]. Here, the features are usually the combination of features in the original data. Non-negative Matrix Factorization, Latent Semantic Indexing and Probabilistic Latent Semantic Indexing are some of the transformation techniques [23][24][25][26][27][28].

4. PROPOSED QUERY RE-ORGANISATION METHOD USING CLUSTERING

It is essential for the proposed text clustering method to extract the relation between verbs and their associated arguments in the same sentence. This extraction has potential information for analyzing terms within a sentence. To identify and clarify the contribution of each term of a particular sentence the information about who is doing what to whom should be used.

QR-C technique captures the semantic structure of each term within a sentence and document rather than the frequency of the term frequency within a document alone. The contexts on three angles depending on the corpus, document and sentence levels are computed. The contexts can be either word or phrases and that are entirely dependent on the semantic structure of the sentence. On arrival of a new document, the contexts from it are extracted and matched with the previously processed documents. Along with this, a new similarity measure is proposed to find the similarity between the documents present in a corpus. This depends on the combination corpus-based, document-based, and sentence-based context analysis.

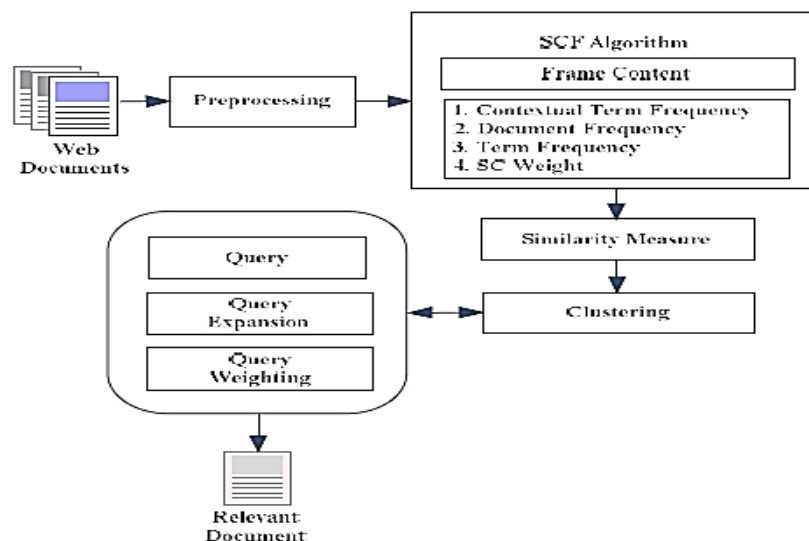


Figure 2 System Architecture

4.1. Steps in Proposed Method

Input to the proposed technique is the raw text document. The input documents are pre-processed through the following steps. Preprocessing has the following five different steps (1) Document's individual sentences are separated, (2) HTML tags in the web document are removed, (3) Stop words are detected, (4) Stemming and (5) POS tagging. Consider an example to compute the SV parameter. Consider a document that contains the following sentences.

“The employees must abide by the rules of the company. Bill always abides by his promises. Problems always arise during such protests for human rights. Disputes arose whom would be the first to speak”.

During pre-processing the above paragraph is separated to sentences as below.

- a. The employees must abide by the rules of the company
- b. Bill always abides by his promises.
- c. Problems always arise during such protests for human rights
- d. Disputes arose whom would be the first to speak

Once the lines are separated then, the stop words and other words that are not discriminate are removed. For example after this the first statement looks as “**Employees abide Rules Company**”. The parameters and verbs are computed and their results are given below.

Param0: employees, **Verb:** abide, **Param1:** rules company.

Param0: Bill always, **Verb:** abides, **Param1:** his promises.

Param0: Problems always, **Verb:** arise, **Param1:** such protests human rights.

Param0: Disputes arose, **Verb:** be, **Param1:** first speak.

Param0: first, **Verb:** speak. While calculating the individual CTF for the contexts (discussed as follows), the sentences are dissected into SV Parameters.

Each sentence will be having conjugation called object that resolves the terms which donate the sentence semantics, associated with their subject verb- argument structure. Context is defined to either phrases or words that depend on the subjective part of a sentence. This is repeated for a whole document and for all documents aggregated on a whole by iterating the above steps.

The estimation of similarity is computed for each context x presented in the sentence s , document d and in the corpus. The similarity analysis part holds three stages.

- (a) Computing CTF, TF and DF
- (b) Similarity measure
- (c) Query weighting based on clusters

4.2. Computing CTF, TF and DF

The value of conceptual term frequency (*CTF*) denotes the number of occurrences of c in SV argument structures of sentence s . This shows the local measure on the sentence measure. The occurrence of c is measured since it has a major role of contributing to the meaning of s . The context c has different *CTF* values for different sentences in a document. Therefore, the *CTF* value of c in a document d is manipulated through the following equation.

$$CTF_d = \frac{\sum_{n=1}^s CTF_n}{s}$$

In the above equation, s denotes the number of sentences that contain the context c in document d . The average value of CTF_d value of the context c in its sentences of document d measures the overall impact of context c to its meaning of the corresponding sentence s in document d . The context that has higher *CTF* values in most of the sentences has foremost contribution to the meaning of its sentence that leads to discover the topic of the document. Therefore, the value obtained through the above equation computes the overall importance of each context to the semantics of a document through the sentences.

Similarly, the term frequency (TF) is computed for the whole documents through counting the number of occurrences of a context c in a document. The corpus may contain a set of documents as $D = \{d_1, d_2, \dots, d_n\}$ and each document contains a set of sentences as $S = \{s_1, s_2, \dots, s_n\}$. Say d_i contains $S = \{s_1, s_2, \dots, s_n\}$, which implies that the document d_i contains n number of sentences. Algorithm explains the procedure for computing the aforementioned terms in the document d_i .

In algorithm, the steps 6-9 are used to compute the *CTF*, *TF* and *DF* values. The weight for the context is computed by using the steps from 11 through 14 and is used to manipulate the weight of each context which is compared with the other documents. The measuring of context

weight is discussed in detail in the following subsection. The sentence *CTF* values will be differing and hence overall *CTF* for *D* is computed through above equation. The *CTF* values are depending on the number of verbs present. If the predicative part in the sentence follows with more than one verb, then the *CTF* value for the parameters following the verbs will be having higher *CTF* values. The following algorithm QR-C context based analysis.

Step 1: begin

Step 2: Consider a document d_i

Step 3: Consider a sentence in document d_i

Step 4: Frame Semantic context by evaluating SV parameter

Step 5: for each context c_i in d_i do

Step 6: Evaluate CTF_i for a context C in d_i

Step 7: Evaluate TF_i for a context C in d_i

Step 8: Evaluate DF_i for a context C in d_i

Step 9: Frame the context catalog L_k from S_i in S

Step 10: for each context $l_j \in L_k$

Step 11: if $l_i = l_j$ then

Step 12: update DF_i of c_i

Step 13: Compute $CTF_{Weight} = \text{avg}(CTF_i, CTF_j)$

Step 14: end if

Step 15: end for

Step 16: end for

Step 17: end.

4.3. Similarity Measure

The most important and significant part in clustering the text document is measuring the similarity between the set of documents, which helps to group the documents effectively. The similarity measure is considered as a noteworthy process since the result of similarity process judges the efficiency of the clustering. The contexts in the document are extracted to determine the semantic structure of each sentence. The occurrence of each context is computed to determine the benefaction of a particular context in the document. The documents are distinguished from the others existing documents by their availability of the context. The sequential flow of computing the context frequency semantically proceeds according to the algorithm QR-C Context Frequency Algorithm.

Following factors are considered for measuring the similarity between the documents.

(a) m : number of matching context is measured for each document.

(b) n : the total number that contains the matching context C_i is computed for all documents.

Along with this, *CTF* for each sentence in a document is computed. The value of *CTF* is computed for all the documents present in the corpus. The *CTF* computation for each C_i in S for each document d_i where $i = 1, 2, 3, \dots, m$ is exported. Similarly, for each document d_i the similarity depends on DF_i , the frequency for the documents where $i \in \{1, 2, 3, \dots, n\}$

The value of *CTF* is the pre-judging factor for evaluating the similarity between documents. The fact lies in frequency of the context lying in the verb SV parametric structure. If the frequency rate is higher, then document is also more similar. The similarity measure between two documents is estimated by the equation below.

$$Sim(d_1, d_2) = \sum_{i=1}^n \max\left(\frac{c1}{C1}, \frac{c2}{C2}\right) \times w_i \times w_j$$

$$w_i = (TFw_i + CTFw_i) + \log\left(\frac{N}{DF_i}\right)$$

In the following equation TFw_i value denotes the weight of context i in document d

$$TFw_i = \frac{TF_{ij}}{\sqrt{\sum_{j=1}^n (TF_{ij})^2}}$$

The TFw_i (Term Frequency weight) corresponds to its contribution in document level.

$$CTFw_i = \frac{CTF_{ij}}{\sqrt{\sum_{j=1}^n (CTF_{ij})^2}}$$

where $CTFw_i$ represents the weight of context i in document d (expressing how far that context is semantically related) and $j = \{1, 2, \dots, n\}$. The summation of TFw_i and $CTFw_i$ represents the effective contribution of the context providing semantic meaning to the document. The above equation exemplifies similarity for each context in the verb argument structure in each document d , its length is evaluated which is denoted as ci . Ci is evaluated on considering each verb agreement structure that is enclosing a matched context, and total number of documents is denoted as N , in the context.

In the equation, $\log\left(\frac{N}{DF_i}\right)$ denotes the value the weight of the context i on the extent of context occurrence. The summation of TFw_i , $CTFw_i$ and $\log\left(\frac{N}{DF_i}\right)$ denotes precise dimension of each context with respect to its semantically contributing perspective over the entire context. The above-said steps are conceded for all documents, finally similarity matrix is acquired for all documents. After computing the similarity between each document, the clustering is done. Clustering is carried out by means of the similarity matrix.

4.4. Query Weighting based on Cluster

Query weighting is the term given when a search engine is adding search terms to a user's weighted search. The goal of query weighting in QR-C mechanism is to improve precision and/or recall. IR is the vital process on the web. The amount of data on the web is always increasing. In 1999, a survey report presented that Google had 135 million pages. It now has over 3 billion. Search engines follow specific mechanism trends with their searches. In the proposed technique, the IR is conceded having Weighted Querying mechanism.

Cluster is a collection of documents having similar terms. For a given query "Y", relevant clusters are extracted through the searching process. Each cluster computes a probability metric (D) for Y. If the query is related to that cluster then, D has the value one, otherwise zero.

$$D = \begin{cases} 1, & \text{if 'Y' exist} \\ 0, & \text{otherwise} \end{cases}$$

The document that has the term 'Y' is extracted. The given query is compared with the document and depends on the similarity, the query weight is computed. This process is carried for all the documents in the cluster. The query weight for a single document can be computed from the equation

$$\sigma = \frac{P X a^2}{b}$$

where 'a' indicates the number of query terms in that sentence, b is the term used to represent the total number of documents that are extracted. Moreover, coefficient P is judged as the Query coefficient. Similarly, the query weight for the cluster is manipulated through the Equation

$$D_i = \sum_{i=1}^n \sigma_i$$

where, i represents the total number of documents in the cluster. The query weight is an important decisive factor to retrieve the relevant documents in the cluster. From this, one can extract the first N number of documents that are matched with query Y. The document extraction is carried out through the equation

$$S = D_i / N$$

Here, S is the status coefficient. The documents are prioritized and ranked. The relevant documents are retrieved based on their rank.

5. RESULT AND DISCUSSION

The performance of the proposed QR-C text clustering technique is carried out through experiments. Two different datasets are used for the experiments, namely (1) Reuters and (2) Usenet. Initially, the datasets are trained in a way to extract and evaluate the right context. Perform POS tagging on the trained dataset using the Stanford Log-linear Part-of-Speech Tagger version 3.1.0 to remove the words that are not discriminative. Consider a snippet as below.

“To resolve the aforementioned problems, we propose a novel method named Navigation-Pattern-based Relevance Feedback to achieve the high retrieval quality of CBIR with RF by using the discovered navigation patterns”.

On applying the POS tagging to the above snippet, the following is obtained.

To/TO resolve/VB the/DT aforementioned/JJ problems/NNS/, we/PRP propose/VBP a/DT novel/NN method/NN named/VBN Navigation-Patternbased/ JJ Relevance/NNP Feedback/NNP to/TO achieve/VB the/DT high/JJ retrieval/NN quality/NN of/IN CBIR/NNP with/IN RF/NNP by/IN using/VBG the/DT discovered/VBN navigation/NN patterns/NNS ./.

With the above POS tagged sentences, the stop words are removed to construct a parsed sentence as given below.

Verb: resolve Param1: aforementioned problems novel method named Navigation-Pattern-based Relevance Feedback achieve high retrieval quality CBIR RF discovered navigation patterns;

Param0: method, Verb: named Param1: Navigation-Pattern-based Relevance Feedback achieve high retrieval quality CBIR RF discovered navigation patterns;

Param0: Feedback, Verb: achieve Param1: high retrieval quality CBIR RF discovered navigation patterns;

Param0: RF, Verb: discovered Param1: navigation patterns.

The tagging of POS is stripped with VBN, VB, VBP that are extracted and the CTF, TF and DF are computed. The result of this computation is shown in Table 1.

Table 1

SContext	CTF	TF	DF
Resolve	1	1	2
aforementioned problems novel method named Navigation-Pattern-based Relevance Feedback achieve high retrieval quality CBIR RF discovered navigation patterns	1	1	0
method	2	1	2
named	2	1	2
Feedback	3	1	2
high retrieval quality CBIR RF discovered navigation patterns	3	1	2
RF	4	1	2
Discovered	4	1	2
Navigation patterns	4	1	2
Aforementioned	1	1	2
Problems	1	1	2
Novel	1	1	2
Relevance	2	1	2
High	3	1	2
Retrieval	3	1	2
Quality	3	1	2
CBIR	3	1	2
Navigation	4	1	2
Patterns	4	1	2

With the computation of the above, clustering algorithms are applied to the documents. The efficiency of both the algorithms such as SOM and Single Pass algorithm is compared. The comparison results are presented as graph in Figure 3. From Figure 3, SOM algorithm is selected for further process.

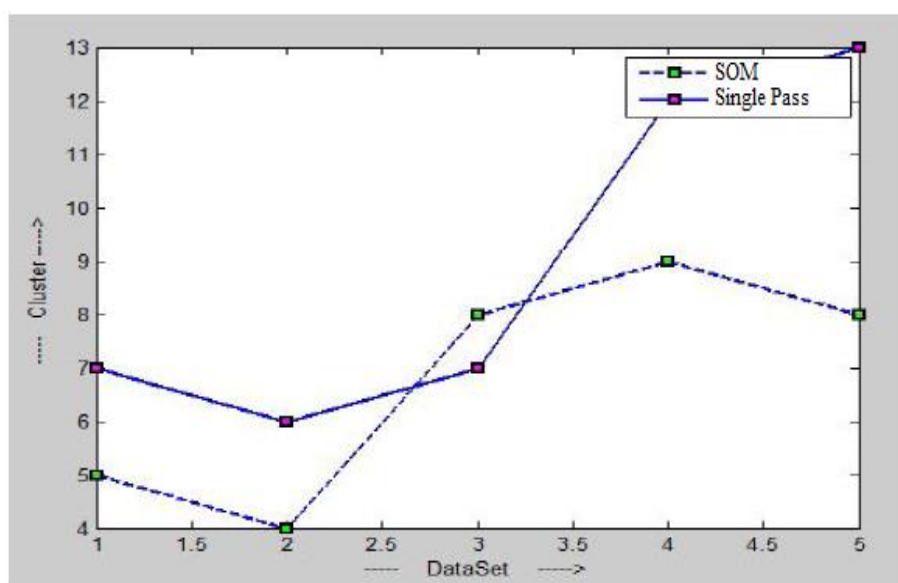


Figure 3 Comparison between SOM and single pass clustering algorithm

Figure 3 expresses the cluster formation of the corresponding algorithm. The X-axis represents five datasets (1-5), with increasing number of documents. Table 2 shows the number of documents present in each dataset.

Table 2 Dataset and number of documents

Dataset	Number of Documents
1	50
2	60
3	70
4	75
5	90

For the dataset-1, 5 clusters for SOM and 7 for Single Pass are procured. Similarly, for dataset-5, 8 clusters for SOM and 13 for Single Pass are evolved. Dataset 4 and 5 comprise of Usenet Dataset. The results are compared with the experimental results of SVM technique presented. The results outperform SVM weighted approach. The proposed approach QR-C is showing good latency rate in processing time. On experimenting it is predicted that the proposed approach consumes less time than the previous SVM approach. Of 50 documents (Dataset 1), QR-C approach has taken 2 seconds, and for SVM it is 3 seconds. Similarly, the experiment is repeated for the remaining datasets.

Moreover, Query weighting scheme is applied for IR. While performing the query weighting mechanism, decisive factor value is kept as 0.2 and the results are estimated for this value. It is healthier to keep the value of decisive factor less than 0.25.

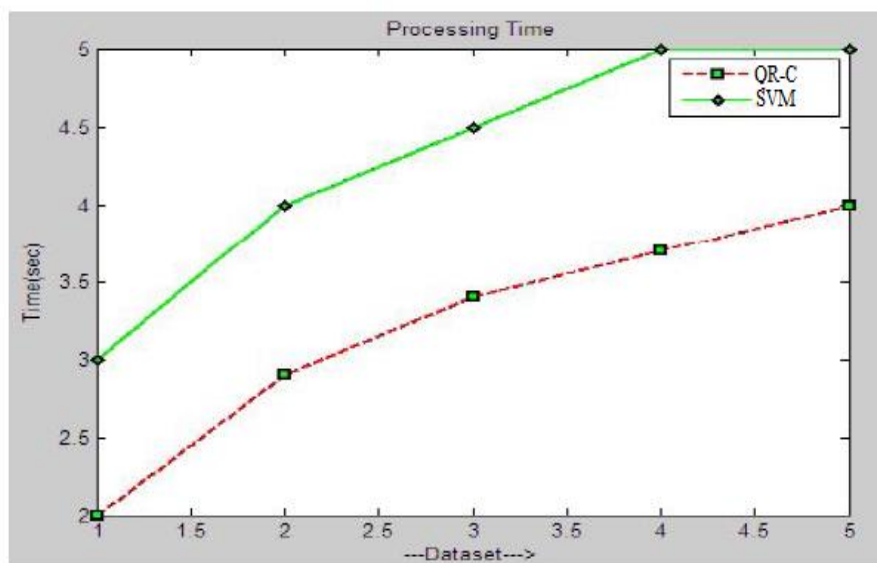


Figure 4 Comparison between two approaches with respect to processing time

Figure 4 represents the time required for the proposed and the existing technique. It is explicit from the above figure that the time taken by the SVM technique is much more than the QR-C technique. It is clear from this analysis that the proposed technique outperforms the existing technique irrespective of the dataset size. The study also shows that the overall time required for the proposed technique is less than the existing techniques for all range of datasets.

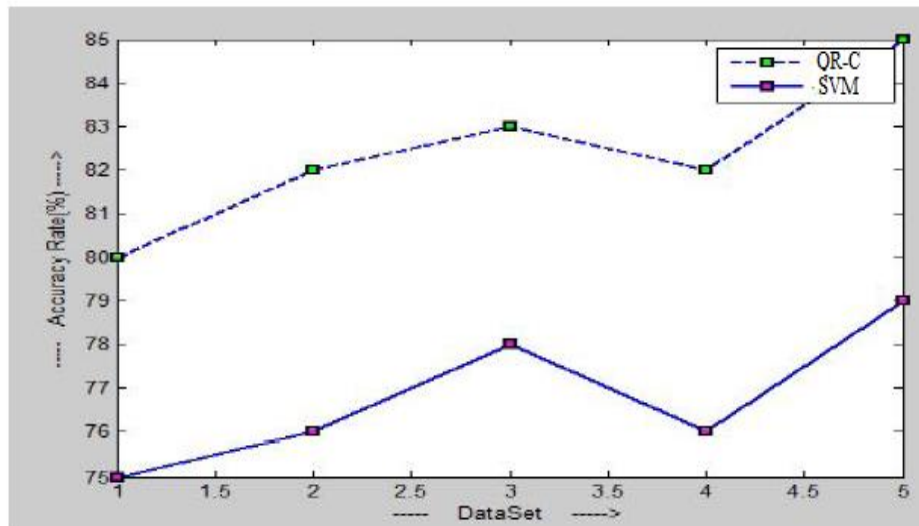


Figure 5 Comparison between two approaches with respect to accuracy rate

Figure 4.5 portrays that the accuracy value is high for QR-C when compared with SVM approach. The precision is defined as the quantity of documents that are retrieved which are appropriate to the search. The term recall is presented as quantity of documents that are retrieved successfully and which are appropriate to the query.

The *precision* P and *recall* R of a cluster j with respect to a class i are defined as:

$$P - Precision(i, j) = C_{ij}/C_j$$

$$P - Recall(i, j) = C_{ij}/C_i$$

where C_{ij} denotes the number of candidates of i and cluster j , C_j denotes the number of candidates of cluster j , C_i denotes the number of candidates of class i . The *F-measure* of a cluster i is defined as in equation

$$F - Measure = \frac{2 \times p \times r}{p + r}$$

Two different clustering techniques, namely single pass and SOM have been tested to cluster similar documents. They are evaluated based on three quantifying measures, namely precision, recall and F-Measure. Table 3 represents the performance of both the clustering techniques for SVM and QR-C approaches.

Table 3 Comparison between single pass and SOM clustering technique based on qualifying measures

Quantifying Measures	Single Pass		SOM	
	SVM	QR-C	SVM	QR-C
Precision	0.34	0.45	0.4	0.49
Recall	0.29	0.37	0.35	0.48
F-measure	0.313	0.406	0.36	0.48

On comparing, SOM clustering outperforms the Single Pass clustering. But the quality of clustering depends on the term frequency, context frequency and document frequency. When compared with SOM, this single-pass clustering is highly sensitive to noise. From the above discussion the conclusion is that the proposed technique clusters the document effectively than

the existing technique. With the help of this technique, the clustered documents are used for information retrieval.

6. CONCLUSION

In this chapter a novel strategic approach called QR-C is implemented for information retrieval. On adopting this approach, initially the SV parametric structure is extracted. This investigates similarity based on the semantic meaning that it affords to the document. The three measures based on contextual term frequency, term frequency and document frequency are estimated offering their semantic merits to a good extent. As the clustering result primarily depends on the similarity matrix, the proximity of that matrix is quite increased. Moreover, the clustering results are finally processed for query expansion and query weighting methodologies. The utility of clustering algorithms is enclosed with superior extent. But the query expansion approach should be enhanced further, so that it should be applied for huge search engine related searches.

REFERENCES

- [1] Allahyari, Mehdi, et al. "A brief survey of text mining: Classification, clustering and extraction techniques." *arXiv preprint arXiv:1707.02919* (2017).
- [2] Wu, Zongda, et al. "An efficient Wikipedia semantic matching approach to text document classification." *Information Sciences* 393 (2017): 15-28.
- [3] Yang, Bao-Qing, et al. "Simultaneous dimensionality reduction and dictionary learning for sparse representation-based classification." *Multimedia Tools and Applications* 76.6 (2017): 8969-8990.
- [4] Hu, Kai, et al. "A domain keyword analysis approach extending Term Frequency-Keyword Active Index with Google Word2Vec model." *Scientometrics* 114.3 (2018): 1031-1068.
- [5] Lucini, Filipe R., et al. "Text mining approach to predict hospital admissions using early medical records from the emergency department." *International journal of medical informatics* 100 (2017): 1-8.
- [6] Beel, Joeran, Corinna Breiter, and Stefan Langer. "Evaluating the CC-IDF citation-weighting scheme: how effectively can 'Inverse Document Frequency' (IDF) be applied to references." *Proceedings of the 12th iConference* (2017).
- [7] Smalheiser, Neil R., and Gary Bonifield. "Unsupervised Low-Dimensional Vector Representations for Words, Phrases and Text that are Transparent, Scalable, and produce Similarity Metrics that are Complementary to Neural Embeddings." *arXiv preprint arXiv:1801.01884* (2018).
- [8] Abualigah, Laith Mohammad, and Ahamad Tajudin Khader. "Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering." *The Journal of Supercomputing* 73.11 (2017): 4773-4795.
- [9] Pereira, Rafael B., et al. "Categorizing feature selection methods for multi-label classification." *Artificial Intelligence Review* 49.1 (2018): 57-78.
- [10] Onan, Aytuğ, and Serdar Korukoğlu. "A feature selection model based on genetic rank aggregation for text sentiment classification." *Journal of Information Science* 43.1 (2017): 25-38.

- [11] Tran, Tram, et al. "Text clustering using frequent weighted utility itemsets." *Cybernetics and Systems* 48.3 (2017): 193-209.
- [12] Khan, Faraz Ahmad, et al. "Robust off-line text independent writer identification using bagged discrete cosine transform features." *Expert Systems with Applications* 71 (2017): 404-415.
- [13] Kayser, Victoria, and Knut Blind. "Extending the knowledge base of foresight: The contribution of text mining." *Technological Forecasting and Social Change* 116 (2017): 208-215.
- [14] D'Urso, Pierpaolo, et al. "Exponential distance-based fuzzy clustering for interval-valued data." *Fuzzy Optimization and Decision Making* 16.1 (2017): 51-70.
- [15] Kang, Mangi, Jaelim Ahn, and Kichun Lee. "Opinion mining using ensemble text hidden Markov models for text classification." *Expert Systems with Applications* 94 (2018): 218-227.
- [16] Bryant, Ivory C., and Krzysztof J. Cios. "SOTXTSTREAM: Density-based self-organizing clustering of text streams." *PloS one* 12.7 (2017): e0180543.
- [17] Anoop, V. S., and S. Asharaf. "Distributional Semantic Phrase Clustering and Conceptualization Using Probabilistic Knowledgebase." *International Conference on Next Generation Computing Technologies*. Springer, Singapore, 2017.
- [18] Moharasan, Gandhimathi, and Tu-Bao Ho. "Extraction of Temporal Events from Clinical Text Using Semi-Supervised Conditional Random Fields." *International Conference on Data Mining and Big Data*. Springer, Cham, 2017.
- [19] Subhashini, M., & Gopinath, R., Mapreduce Methodology for Elliptical Curve Discrete Logarithmic Problems – Securing Telecom Networks, *International Journal of Electrical Engineering and Technology*, 11(9), 261-273 (2020).
- [20] Upendran, V., & Gopinath, R., Feature Selection based on Multicriteria Decision Making for Intrusion Detection System, *International Journal of Electrical Engineering and Technology*, 11(5), 217-226 (2020).
- [21] Upendran, V., & Gopinath, R., Optimization based Classification Technique for Intrusion Detection System, *International Journal of Advanced Research in Engineering and Technology*, 11(9), 1255-1262 (2020).
- [22] Subhashini, M., & Gopinath, R., Employee Attrition Prediction in Industry using Machine Learning Techniques, *International Journal of Advanced Research in Engineering and Technology*, 11(12), 3329-3341 (2020).
- [23] Rethinavalli, S., & Gopinath, R., Classification Approach based Sybil Node Detection in Mobile Ad Hoc Networks, *International Journal of Advanced Research in Engineering and Technology*, 11(12), 3348-3356 (2020).
- [24] Rethinavalli, S., & Gopinath, R., Botnet Attack Detection in Internet of Things using Optimization Techniques, *International Journal of Electrical Engineering and Technology*, 11(10), 412-420 (2020).
- [25] Priyadharshini, D., Gopinath, R., & Poornappriya, T.S., A fuzzy MCDM approach for measuring the business impact of employee selection, *International Journal of Management*, 11(7), 1769-1775 (2020).

- [26] Poornappriya, T.S., & Gopinath, R., Application of Machine Learning Techniques for Improving Learning Disabilities, International Journal of Electrical Engineering and Technology, 11(10), 403-411 (2020).
- [27] Poornappriya, T.S., & Gopinath, R., Employee Attrition In Human Resource Using Machine Learning Techniques, Webology, 18(6), 2844-2856 (2020).
- [28] Poornappriya, T.S., & Gopinath, R., Rice Plant Disease Identification using Artificial Intelligence Approaches, International Journal of Electrical Engineering and Technology, 11(10), 392-402 (2020).