IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 08-14

A Scalable Framework for Big Data Summarization using modified K-means clustering on Map reduce framework Approach

Shilpa G. Kolte[1], Jagdish W. Bakal[2]
[1](Research Scholar, Terna Engineering College, Navi Mumbai, University of Mumbai, India.)
[2](Professor in Computer Engineering, Shivajirao S. Jondhale College of Engineering, Mumbai,)

Abstract: *This paper proposed a novel framework for big data summarization,. The proposed framework works in four stages and gives a measured execution of various records outline. In this paper we proposed modified clustering algorithm and semantic approach for big summarization. The exploratory outcomes utilizing Iris dataset demonstrate that the proposed modified k-means algorithm performs superior to K-means and K-medodis algorithm. The execution of Big Data synopsis is assessed utilizing Australian legal cases from the Federal Court of Australia (FCA) database. The experimental results demonstrate that the proposed method can summarize the big data document superior as compared with existing systems.*

Keywords: *Big Data, Data Summarization, MapReduce, Data Generalization, Semantic term Identification*

I. Introduction

Big data bring new challenges to data mining because large volumes and different varieties must be taken into account. Big Data can be portrayed by three V: volume (a lot of information), Veracity (incorporates distinctive sorts of information), and Velocity (continually gathering new information) [1].The common methods and tools for data processing and analysis are unable to manage such amounts of data, even if powerful computer clusters are used[3,4]. To analyze big data, many new data mining and machine learning algorithms as well as technologies have been developed. So, big data do not only yield new data types and storage mechanisms, but also new methods of analysis. When dealing with big data, a data clustering problem is one of the most important issues [5]. Often data sets, especially big data sets, consist of some groups (clusters) and it is necessary to find the groups [6]. Clustering methods have been applied to many important problems, for example, to discover healthcare trends in patient records, to eliminate duplicate entries in address lists, to identify new classes of stars in astronomical data, to divide data into groups that are meaningful, useful, to cluster millions of documents or web pages. To address these applications and many others a variety of clustering algorithms has been developed. There exist some limitations in the existing clustering methods; most algorithms require scanning the data set for several times, thus they are unsuitable for big data clustering. There are a lot of applications in which extremely large or big data sets need to be explored, but which are much too large to be processed by traditional clustering methods. Summarizing large volume of text is a challenging and time consuming problem particularly while considering the semantic similarity [7] computation in summarization process.

The proposed system a Scalable Framework for Big Data Summarization using modified K-means clustering on Map reduced framework [8] Approach (BDS-MKM) provides the solution of these problems. In this chapter modified k-means clustering algorithm is used for big data summarization. The proposed system works in four phases and provides a modular implementation of multiple documents summarization. The experimental results using Iris dataset show that the proposed clustering algorithm performs better than K-means and K-medodis algorithm. The performance of big data summarization is evaluated using Australian legal cases from the Federal Court of Australia (FCA) database. The experimental results demonstrate that the proposed method can summarize the big data document superior as compared with existing systems.

This paper is organized as follows. Section II present related work for Big data Summarization Section III describes framework BDS-MKM. Section IV illustrates experimental setup of the proposed data summarization system. This section also gives performance evaluation with the existing algorithms. At last we conclude the chapter.

II. RELATED WORK

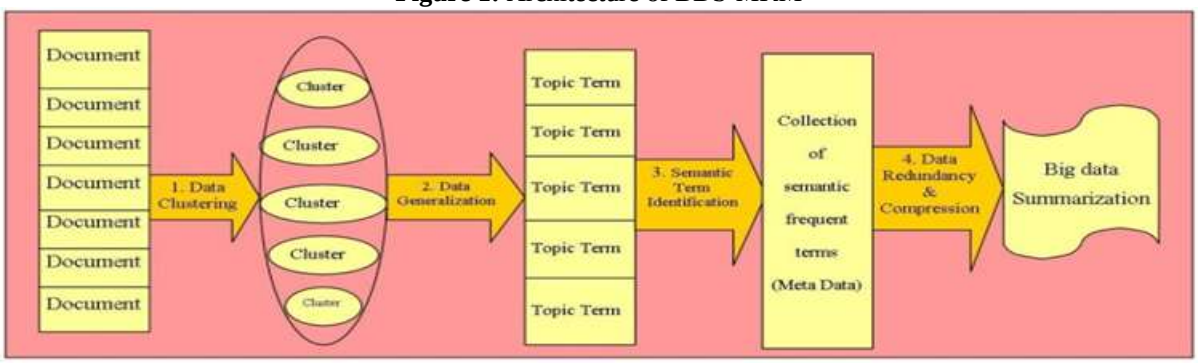
Gong and Liu proposed rundown technique utilizing LSA (Latent Semantic Analysis). This technique separates the imperative sentence which has the biggest list an incentive concerning the essential particular vector by LSA [9]. Zha utilized the shared fortification rule (MRP) and sentence bunching for the bland

rundown. Their technique bunches sentences of records into a few topical gatherings by sentence grouping strategy. And after that, sentences are removed from each topical gathering by saliency scores utilizing the MRP (i.e., altered LSA strategy) [10]. Yeh et al. proposed the outline technique utilizing LSA and the content relationship delineate). Their technique finds semantic sentences utilizing LSA. TRM is developed by the semantic sentences, and the imperative sentences are extricated by the quantity of connections in TRM [11]. Li et al. broadened the bland multi-record outline utilizing LSA for the question based report synopsis [12]. Han et al. proposed a content rundown strategy utilizing importance criticism with inquiry part (i.e., a question development process by part the underlying question into a few pieces) [13]. Diaz and Gervas proposed a thing outline strategy for the personalization of news conveyance frameworks. The technique utilizes three expression determination heuristics that construct rundowns utilizing two nonexclusive synopses and one customized synopsis relying upon RF from news things [14]. Additionally, they proposed a programmed customized rundown utilizing a mix of non specific and customized techniques. Their non specific rundown strategies join the position technique with the topical word strategy. Their customized strategy chooses those sentences of a record that are most applicable to a given client display [15]. Kumar et al. created customized outlines utilizing non specific and client particular techniques in light of likelihood. This technique extricates the best positioning sentences by methods for the nonexclusive sentence scoring and the client particular sentence scoring [16]. Ko et al. proposed a web bit era strategy from website pages utilizing PRF and an inquiry one-sided synopsis in view of the likelihood demonstrate [17]. Li and Chen separated customized content bits utilizing the likelihood succession investigation and the shrouded Markov show [18]. S. Stop et al. are proposed the archive rundown techniques utilizing sentence positioning relying upon the semantic highlights of the NMF [19, 20, 21].

III. BDS-MKM FRAMEWORK

Big data not only refers to datasets that are large in size, but also covers datasets that are complex in structures, high dimensional, distributed, and heterogeneous. An effective framework when working with big data is through data summaries, such as data integration, data redundancy, and etc. Instead of operating on complex and large raw data directly, these tools enable the execution of various data analytics tasks through appropriate and carefully constructed summaries, which improve their efficiency and scalability The Fig. 1 depicts the architecture of proposed system with its essential components. The working of proposed data summarization is described below in detail.

Figure 1: Architecture of BDS-MKM



3.1 Data Clustering using modified K-means algorithm

The first stage is the document clustering stage where clustering technique is applied to the multi document data collection to create the document clusters. The purpose of this stage is to group the similar documents for making it ready for summarization and ensures that all the similar set of documents participates in a group of summarization process. For big data summarization highly scalable clustering algorithms are required which deals with large databases and different kinds of attributes. The algorithm should be capable to be applied on any kind of data such as interval-based (numerical) data, categorical, and binary data. The clustering algorithm should not only be able to handle low-dimensional data but also the high dimensional space. The varieties of clustering algorithm are available like K-means, K-medoids which has some limitations. All these clustering algorithm centroids are initially selected by the user. Therefore, performance of these algorithms depends on this manual selection of centroids. It works inefficiently for large data sets due to its complexity. This is the major motivation behind the work presented in this chapter. The proposed clustering algorithm initially calculates centroids appropriately; this results in the proper creation of the clusters.

The proposed modified K-means clustering algorithm consists of three steps, determining the centroids, grouping and removing grouped patterns. These three steps are described below in detail. The number of

clusters constructed depends on the user defined parameters α and β, called as centroid and grouping factors, respectively and the values of these parameters are problem dependent. Assume $R \in \{R_h \mid h = 1,2,..,P\}$ where $R_h = (r_{h1}, r_{h2}, ..., r_{hn})$ is the *n*-dimensional h^{th} pattern belonging to the set R containing *P* patterns to be clustered

(i) Determining the centroid: To determine the centroid of the cluster, all the patterns are applied to each of the pattern and the patterns having Euclidian distance less than or equal to α are counted for all the patterns. If R_h is the pattern with the maximum count then it is selected as the centroid of the cluster.

(ii) Grouping: The patterns which are falling around the centroid and having the Euclidian distance less than or equal to β are bunched in a cluster. The centroid of the cluster is recalculated by calculating the average of all the patterns bunched in a cluster. Thus the cluster boundaries are governed by the value of grouping factor.

(iii) Removal of the grouped patterns in a cluster: The patterns included by created cluster in the previous step are eliminated. Thus, the next pass uses unclustered pattern set consisting of remaining patterns for clustering. These three steps are repeated till all the patterns are clustered. Let R_p, R_c and R_n represent set of patterns used in the current pass, set of patterns clustered in the current pass and set of patterns that will be used in the next pass, respectively. Then R_n can be described as,

$$R_n = R_p - R_c = \{R_n \mid R_n \in R_p \text{ and } R_n \notin R_c\} \qquad (1)$$

The R_n calculated in the current pass becomes R_p for the next pass. The steps described above are repeated until all the patterns are clustered and the process stops when R_n becomes empty

3.2 Data Generalization

Data generalization is process that abstracts a large set task relevant data from low conceptual to high conceptual level for better prediction and summarization. The goal is to find short descriptions of the members of a collection that enable efficient processing of large collections while preserving the essential statistical relationships that are useful for basic tasks such as classification, novelty detection, and similarity and relevance judgments. The first challenge is database Enrichment for the purpose of effective data generalization. The second challenge is the generalization process, if automatized, must be controlled by some decision system. The system usually makes a decision based on the set of attributes describing data being generalized. The challenge of selecting the right attributes to control generalization process may be as important as creation of decision system using those attributes. To efficiently data generalization we, LDA topic modeling technique is proposed to each individual document cluster to generate the cluster topics and terms belonging to each cluster topic.

Step1: For each cluster get the documents it contains and extract the text collection from these documents. For each $C_i \in \{C_1, C_2, ...C_n\}$ cluster

Step 2: Extract the documents C_i as $\{D_{i1}, D_{i2}, ...D_{in}\}$. For each document extract and merge the text from the text collection. $Text = Text \cup \{Text_{ij} \in D_{ij}\}$

Step 3: Apply LDA topic modeling to these collection and get the list of topics for the cluster C_i as $T_{i1} = \{T_{i1}, T_{i2}, ...T_{iik}\}$

Step 4: Integrate the topics of all the clusters for each cluster $C_i \in \{C_1, C_2, ...C_n\}$

Step 5: Extract the topics discovered by LDA in the documents in as $T_{i1} = \{T_{i1}, T_{i2}, ...T_{iik}\}$

Step 6: For each document extract the text and computer the text collection.

$Topics = Topics \cup \{T_{ij} \in T_i\}$

3.3 Semantic term Identification

Semantic term identification presents the data in a more efficient manner and makes it useful for a source of knowledge discovery and comprehension, for example by making search engines work more quickly and efficiently. Data representations play an important role in the indexing of data, for example by allowing data points/instances with relatively similar representations to be stored closer to one another in memory, aiding in efficient information retrieval. However, the high-level abstract data representations need to be meaningful and demonstrate relational and semantic association in order to actually confer a good semantic understanding and comprehension of the input. To enable the discovered patterns/results to be useful domain knowledge, the data analysts must provide process how global frequent terms are generated from the collection of multiple documents. For frequent terms generation of the multiple documents in each cluster, we proposed new

process/algorithm based on the principle that data (words) that are used in the same contexts tend to have similar meanings.

In this stage, semantic similar terms are computed for each topic term generated in previous stage. WordNet Java API [22] is used to generate the list of semantic similar terms. The semantic similar terms are generated over the MapReduce framework and the generated semantic terms are added to the vector. Semantic similar term finding is an intensive computing operation. It requires going through with the vocabulary and synonyms data for the given term in the hierarchy of semantic relationship. MapReduce framework is utilized efficiently for handling this operation. The Mapper computes the semantic similar terms for each topic term generated by the document cluster and reducer aggregate these terms and counts the frequencies of these terms (topic terms and semantic similar terms of topic terms)[23] aggregately. The mapper and reducer for semantic terms generation from cluster topic terms is presented as follows

// Mapper:

a. for each keyword term in keyword list {K_1, K_2, …….K_n}
b. get the semantic similar term KS_i= ComputeSematicSimilar(K_i)
   (Pass the term K_i in wordnet API and extract smilar term in the set KS_i)
c. for all keyword n, n € KS_i present all document D do.

// Reducer:

d. For each keyword K, count {C_1,C_2, …….C_n}
e. Initialize the sum term of keyword frequency as 0.
f. For all count C € count {C_1,C_2, …….C_n} do
g. Update count sum=sum+C;
h. Count sum.

3.4 Remove Data redundancy & Summarization

Data redundancy aims to reduce redundant information in data to save the consumption of resources such as storage space or I/O bandwidth, and therefore this technique has very important application in the areas of data summarization. In the last stage, the original text document collection is distributed over the Mappers and using parsing techniques, sentences are extracted from individual document by the Mappers. The sentences which are consisting of the frequent terms and its semantic similar terms are filtered from the original text collection and added to the summary document (in other words the filtered terms participates in the summary document). The final summary is generated after traversing all the documents in the document collections. The steps are as follows:

Step 1: Select one document at a time from the document collection. For each $D \in \{D_1, D_2, ...D_n\}$.

Step 2: Extract the sentences from Document *D* as S_{i1}, S_{i2}, using parsing.

Step 3: If contains the term present in TS_i filter redundant sentences S_{ik}. containing the terms and add it to Vector. $Vector = Vector \cup \{S_{ik}\}$

Step 4: Integrate all the filtered sentences and Produce a single document presenting $Summary = Summary \cup \{S_{ik}\}$.

IV. Experiments

The implementation is carried using the Java based open source technologies. The LDA implementation is performed using MALLET API and the Map Reduce implementation is performed using Hadoop API. A textual corpus of around 4000 legal cases for automatic summarization is selected for performing the experiments; the dataset is available on UCI machine learning repository. https://archive.ics.uci.edu/ml/datasets/Legal+Case+Reports. The dataset contains Australian legal cases from the Federal Court of Australia (FCA) all files from the year 2006, 2007, 2008 and 2009. The dataset is earlier used in the work of Galgani et al. [89-90]. The experiments are performed over the dual core processor based systems with CPU speed 2.8 GHz, 4 GB of RAM and 1.333 GHz bus clock in Windows XP operating system. The systems (up to four nodes) are interconnected over a 50 Mbps Local Area Network (LAN).

4.1 Performance evaluation of clustering algorithms using Iris dataset

In order to check the performance of the proposed clustering algorithm, the algorithm is first applied to real data set, 'Iris' data, whose true classes are known. The Iris data set is available in UCI repository ftp://www.ics.uci.edu/ pub/machine-learning-databases/. The performance was measured by the accuracy which is the proportion of objects that are correctly grouped together against the true classes. To investigate the performance more objectively, a simulation study was carried out by generating artificial data sets repetitively and calculating the average performance of the method.

The proposed modified K-means method, K-means, and K-medodis are applied to create three clusters using this data without the class information. The class of an object cannot be predicted by a clustering algorithm but it may be estimated by examining the cluster result for the class-labeled data. TABLE 3.1 show the confusion matrices by K-means , K-medodis methods and proposed modified k-means method.

TABLE 3.1: Cluster result of Iris data by the proposed and traditional methods

Algorithms	Clustering Performance for each set		
	Set 1 (Setosa)	Set 2 (Versicolor)	Set 3 (Virginica)
K-means	100 %	63%	52%
K- medoids	100%	67%	64%
Proposed Modified K-mean	100 %	68%	65%

4.2 Performance evaluation of big data summarization using UCI machine learning

Precision, Recall and F-Measure are among the simplest evaluation approaches available that measure the relevance of a summary by the relevance of the sentences it contains. Precision (P) is the number of sentences appearing in both the system summary and the reference summary divided by the number of sentences in system summary. Recall (R) is the number of sentences occurring in both system and reference summaries divided by the number of sentences in the reference summary. F-Score is a composite combining both P and R. The F-Score can be computed with the following formula:

$$F = \frac{(1+\beta)^2 PR}{\beta^2 P + R} \qquad (2)$$

where β is a weighting variable that is adjustable to affect precision and recall. The Precision/Recall measure is not without its limitations.

Recall Oriented Understudy for Gisting Evaluation (ROUGE) was proposed in 2003 [96] at the Information Science Institute. It is roughly based on BLEU but focuses on recall instead. Also, it measures words overlaps in sequences and was found to correlate better with human evaluations than many other systems. ROUGE-N is an n-gram recall between a candidate summary and set of reference summaries. ROUGE measures include several automatic evaluations such as ROUGH-N, ROUGH-L, ROUGH-W, ROUGH-S, ROUGH-SU

ROUGE-N is an n-gram recall computed as given in the

$$ROUGH - N = \frac{\sum_{Se\{\mathrm{Re}\, f\, .Summ\}} \sum_{gram_n es} Count_{match}(gram_n)}{\sum_{Se\{\mathrm{Re}\, f\, .Summ\}} \sum_{gram_n es} Count(gram_n)} \qquad (3)$$

where n represents the length of the n-gram, and ref represents the reference summaries. $Count_{match}(gram_n)$ represents the number of n-grams co-occurring in a candidate and the reference summaries, and $Count(gram_n)$ represents the number of n-grams in the reference summaries. ROUGE-L measure uses the longest common subsequence (LCS). In this work the average precision, recall and F-measure scores generated by ROUGE-1, ROUGE-2, and ROUGE-L are used to measure the performance of the summaries and to compare the presented algorithm over the MapReduce framework

4.3 Result Analysis

We have implemented three different summarization method such as LSA , BDS and proposed BDS-MKM system . LSA presented by Gong and Liu's summarization method using latent semantic analysis in 2001 and BDS explored by Yoo-Kang et al in 2014.

The scalability is calculated using different nodes and different numbers of text document reports for generating the summary using the proposed MapReducer based summarizer. Scalability tends to increase in proportion to the number of text documents with maximum numbers of nodes. The scalability of the proposed work is also supported by the Amdahl's law. As per the Amdahl's law , the optimal speedup possible for a computation is limited by its sequential components. If f is the fraction of the computational task then the theoretically maximum possible speedup for N parallel resources is:

$$S_N = \frac{1}{\left(f + \frac{1-f}{N}\right)} \qquad (4)$$

The scalability of the proposed work in MapReduce framework up to four nodes is shown in the fig 2.

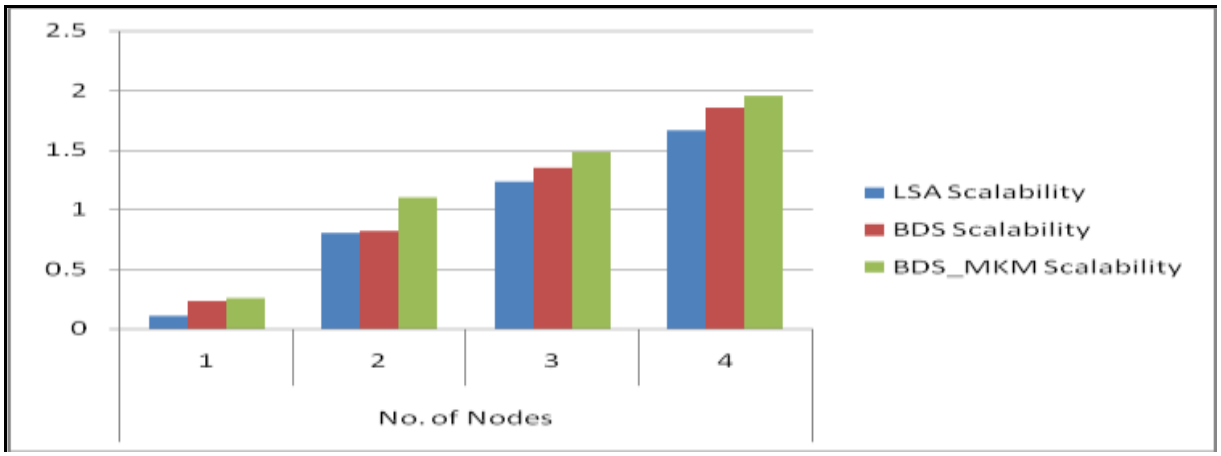


Figure 2: Scalability of Map Reducer based Summarizer

Fig.3 illustrated Precision (P), Recall (R) and F-measure (F) for ROUGE-1, ROUGE-2 and ROUGE-L for all three summarization approaches

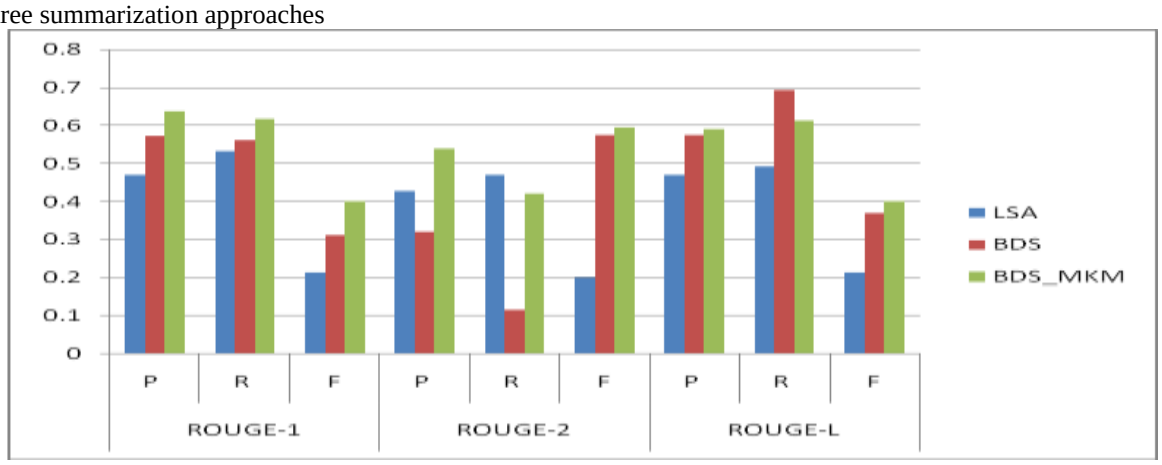


Figure 3: ROUGE evaluation results of LSA, BDS and BDS-MKM for the different cases

V. Conclusions

This paper describes A Scalable Framework for Big Data Summarization using modified K-means clustering on Map reduced framework approach. The result from various simulations using Iris data set shows that the proposed modified K-means clustering algorithm performs better than K-means and K-medoid clustering, which helps to improve scalability and F-measure. Traditional document summarization methods are restricted for summarizing suitable information from the big document data where as in the proposed big data summarization which the information is summarized from a big document data. F-score and Time complexity of proposed system is better than LSA and BDS.

References

[1]. S. Schmidt, Data is exploding: the 3 V's of big data. Business Computing World, 2012.
[2]. Y. Zhai, Y.-S. Ong, and IW. Tsang, The Emerging "BigDimensionality". In Proceedings of the 22nd International Conference on World Wide Web Companion, Computational Intelligence Magazine, IEEE, vol. 9, no. 3, pp. 14–26, 2014.
[3]. V. Medvedev, G. Dzemyda, O. Kurasova, and V. Marcinkevi˘cius, "Efficient data projection for visual analysis of large data sets using neural networks", Informatica, vol.22, no. 4, pp. 507–520, 2011.
[4]. G. Dzemyda, O. Kurasova, and V. Medvedev, "Dimension reduction and data visualization using neural networks", in Maglogiannis, I., Karpouzis, K., Wallace, M., Soldatos, J., eds.: Emerging Artificial Intelligence Applications in Computer Engineering. Volume 160 of Frontiers in Artificial Intelligence and Applications, IOS Press, 2007, pp. 25–49.
[5]. A. McCallum, K. Nigam,and L. Ungar, "Efficient clustering of high-dimensional data sets with application to reference matching", in Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 169–178, 2000.
[6]. M.H. Dunham, Data Mining: Introductory and Advanced Topics, Prentice Hall PTR, Upper Saddle River, NJ, USA, 2002.
[7]. MacQueen, "Some methods for classification and analysis of multivariate observations", in Le Cam, L.M., Neyman, J., eds.: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley and Los Angeles, CA, USA, University of California Press, vol. 1, pp. 281–297, 1967.
[8]. T. Kohonen, Overture. Self-Organizing neural networks: recent advances and applications, Springer-Verlag, New York, NY, USA, 2002, pp. 1–12.

[9]. Y. Gong, X. Liu, "Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis", in proceedings of the 24th annual international 1102 *Yoo-Kang Ji et al.* ACM SIGIR conference on research and development in information retrieval (SIGIR'01), pp.19-25, New Orleans, USA, 2001.

[10]. H. Zha, "Generic Summarization and Keyphrase Extraction Using Mutual Reinforcement Principle and Sentence Clustering", In proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'02), pp.113-120, Tampere, Finland, 2002.

[11]. J. Y. Yeh, H. R. Ke, W. P. Yang, I. H. Meng, "Text summarization using a trainable summarizer and latent semantic analysis", Information Processing and Management 41, pp.75-95, 2005.

[12]. W. Li, B. Li, M. Wu, "Query Focus Guided Selection Strategy for DUC 2006", In proceedings of the Document Understanding Conference (DUC'06), 2006

[13]. K S Han, D H Bea, and H C Rim, "Automatic Text Summarization Based on Relevance Feedback with Query Splitting," In proceedings of the 5th International Workshop on Information Retrieval with Asian Language, Hong Kong, pp.201-2, Sep. 2000.

[14]. A Diaz, and P Gervas, "Item Summarization in Personalisation of News Delivery Systems," In proceeding of the 7th International Conference on Text, Speech and Dialogue (TSD), LNAI 3206, Brno, Czech Republic, pp. 49-56, Sep. 2004.

[15]. A Diaz, and P Gervas, "User-model based personalized summarization," Information Processing and Management, vol. 43, pp.1715-34, Mar. 2007.

[16]. C Kumar, P Pingali, and V Varma, "Generating Personalized Summaries Using Public Available Web Documets," In proceeding of the International Conference on Web Intelligence and Intelligent Agent Technology, Sydney, Australia, pp.103-6, Dec. 2008.

[17]. Y J Ko, H K An, and J Y Seo, "Pseudo-relevance feedback and statistical query expansion for web snippet generation," Information Processing Letters, vol. 109, pp.18-22, 2008.

[18]. Q Li, and Y P Chen, "Personalized text snippet extraction using statistical language models," Pattern Recognition, vol. 43, pp.378-86, 2010.

[19]. J H Lee, S Park, C M Ahn, and D H Kim, "Automatic Generic Document Summarization Based on Non-negative Matrix Factorization," Information Processing and Management, vol. 45, pp.20-34, Jan. 2009

[20]. S Park, B R Cha, and D U An, "Automatic Multi-document Summarization Based on Clustering and Nonnegative Matrix Factorization," IETE TECHNICAL REVIEW, vol. 27, no. 2, pp.167-78, Mar. 2010.

[21]. S Park, B R Char, C. U. Kwon "Personalized Document Summarization using Pseudo Relevance Feedback and Semantic Feature" IETE JOURNAL, vol. 58, no. 2, pp.155-165, 2012.

[22]. J. Sander, M. Ester, H-P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: the algorithm GDBSCAN and its applications," Data Mining and Knowledge Discovery, 2(2):169–194, 1998