A Narravite Cloud Storage Management Using Multi-level Semantic Bonding Technique

K. Thamizhchelvi, Y. Kalpana

Abstract--- The problem of cloud storage management has been well studied. The growing size and types of data increases the challenge in storage and retrieval. Number of approaches has been discussed for the problem of storage management. Text based clustering algorithms and Semantic based approaches are defined to improve the performance of storage management. However, the methods suffer to achieve higher performance in indexing and retrieval in terms of storage management. To solve this issue, an efficient semantic bonding measure based clustering and data management algorithm is presented. The method maintains ontology of various classes where each class has been mentioned in multiple levels. Each level of a class has specific properties and values. Using the semantic ontology, the method estimates MSB (Multi-level semantic Bonding) measure for different class of data. The same has been estimated for different level of semantic classes. Indexing of document class is performed based on MSB where the documents similarity has been measured using Topical Closure Measure (TCM). According to the value of TCM, the documents which are similar are identified and merge. The proposed algorithm improves the performance of document clustering and storage management in cloud environment.

Index Terms--- Cloud Data, High Dimensional Clustering, Semantic Clustering, Topical Measures, MSB.

I. INTRODUCTION

The modern trend in representing information has been changed. This allows the relational information to be combined to produce customized data. This increases the dimension of the data and increases the requirement of higher space. The organizations would have such huge sized data but they face the issue in space complexity. The organizations would not afford huge cost to maintain higher storage capacity. This introduces the cloud to the problem. The cloud is an environment where different resources can be deployed and accessed through different services provided by any CSP (Cloud Service Provider). The loosely coupled environment (CLOUD) has opened the gate for the organization to store their data and access them whenever required.

As like above discussion, when the size and number of data of the cloud increases, identifying or searching the specific type of data becomes quite challenging one. Consider a document set Ds, which has documents related to different categories of CS (Category set). If the documents are stored in the cloud data servers in a random fashion, then identifying the document related to a category C of CS, needs to access all the documents of cloud server to identify the specific document. This increases the complexity of time and false ratio

The problem of clustering huge data would require efficient approach and when the dimension of data increases, the requirement of high dimensional clustering becomes huge. In earlier days, the clustering of data points has been performed by measuring the similarity in specific feature. Considering single or few dimensions in similarity measurement encourages the poor clustering and false classification ratio. It is necessary to include maximum features or dimensions in similarity measurement. In the same way, the document clustering is performed based on the content available and the information present in the document. Consider a class "Books" which covers numerous types or domains. In order to cluster the documents of books, it is necessary to identify the class of document. But, the book class itself can be classified into number of sub classes like comics, lyrics, medicine, subject and so on.

The problem of document clustering has to consider semantic meanings between documents. The semantic ontology has been used to measure the semantic measures. The ontology contains the relation of different classes and features with others. It can be used to measure the semantic relation between the documents. By grouping the documents using semantic relationship, the documents are more closure and more relevant. The proposed approach considers semantic measures and topical measures in measuring the relevancy of documents. The detailed approach is discussed in next section.

II. RELATED WORKS

Number of methods of clustering has been discussed in several articles. This section details few of them related to clustering in cloud environment.

Fast and Reliable Restoration Method of Virtual Resources on Open Stack [1] propose a fast and reliable restoration method with a uniform way for plural types virtual resources. In our method, Pacemaker only detects a physical server failure and notifies a failure to a virtual resource arrangement scheduler, and then a virtual resource arrangement scheduler determines multiple physical servers to restore virtual resources and calls Open Stack APIs to rebuild.

In [3], the author proposes a secure cloud computing based framework for big data information management in smart grids, which we call "Smart-Frame. The method use identity based encryption to enforce security. In [5], a public auditing scheme with dynamic support has been presented.

In [6], the author present a framework named Log Drive towards forensic support of IaaS in cloud.

Manuscript received September 16, 2019.

K. Thamizhchelvi, Research Scholar, Department of Information Technology, Vels Institute of Science and Technology & Advance Studies, Pallavaram, Chennai. T.N, India. (email: thamizhchelvi.k@gmail.com)

Dr.Y. Kalpana, Professor, Department of Information Technology, Vels Institute of Science and Technology & Advance Studies, Pallavaram, Chennai. T.N, India. (email: ykalpanaravi@gmail.com)

In [7], presents a dynamic control algorithm without violating the average temperature constraint.

In [8], the problem of duplication in maintaining user files in cloud has been well studied. This introduces higher storage complexity and requires many storage nodes [9]. In [10], a relational content similarity based cluster and data management is presented. In [11] an inter cloud architecture has been presented which combines the services of multiple cloud providers. Due to the highest number of users, optimized management of the storage space around the data centers is required [12]. Many studies have been focused on storage space optimization; researchers are trying to create some complex compression algorithms without considering their execution time and the user's behavior with a slow system [13]. Users want to access their files one-hundred percent of the time [14]. This model is based on a user who is willing to spend extra money in return for more space. Document clustering is an automatic grouping of text documents into clusters so that documents compared with internal segmentation to measure the similarity [15].

All the methods suffer to achieve higher performance in data management and clustering of documents in cloud environment.

1. Semantic Bonding Based Cloud Data Management:

The proposed multi-level semantic bonding measure based approach reads the input document set. Each document has been identified with list of keywords and uses them to estimate semantic bonding measure. Based on the semantic measure, the target class of document has been identified. Then the documents of any class have been measured with the topical closure measure. Using these two, the method indexes or clusters the documents. The same has been used to perform merging of document and remove the redundancy in cloud data. The detailed approach is discussed below.

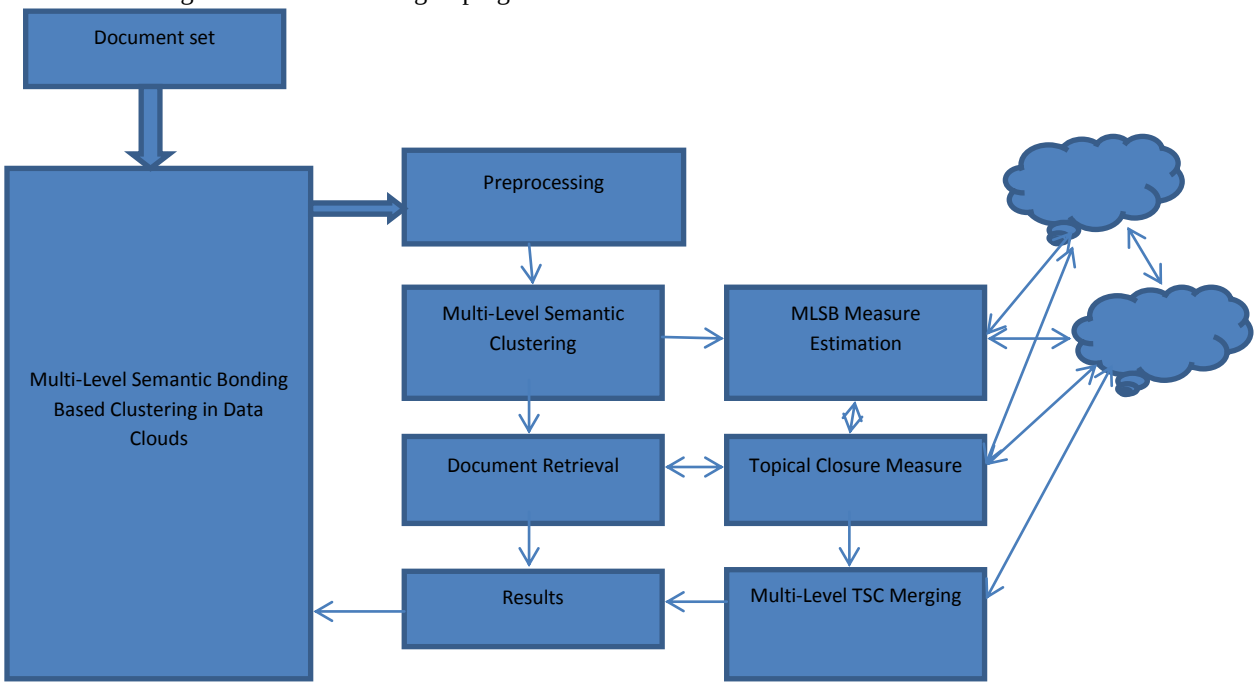

Figure 1: Architecture of proposed semantic bonding based clustering algorithm

The Figure 1 shows the abstract schematic diagram of proposed clustering algorithm. Each functional stage has been discussed in detail in this section.

Preprocessing

The input document set given Ds, has been read and the text features of each document has been extracted. The document text has been split into number of statements. From each statement, the method extracts the key terms by removing the stop words. The key terms are added to the term set. The key terms or nouns of any document have been identified using part of speech tagger. The term set generated has been used to estimate various measures.

Given a document D, the text feature of the document has been extracted as follows:

Document Text DT = Text-Feature (D).

From the document text, the statements are generated as follows:

$$\text{Statement Set St} = \int_{i=}^{size(DT)} Split(DT,'.','\backslash n') \text{ -- (1)}$$

Now the term set for entire document has been identified as follows:

$$\text{Term Set Ts} = \int_{i=1}^{size(ST)} Split(ST(i),'') \text{ (2)}$$

The term set generated Ts would have many stop words and the pure term set has been generated as follows:

$$\text{Ts} = \int_{i=1}^{Size(Ts)} \left(PoS\left(Ts(i)\right) == !Noun\right) \cap Ts(i) \text{ (3)}$$

The term set and sentence sets has been used for clustering and merging which is performed next.

Semantic Bounding Estimation

The semantic bound measure represents the bonding the documents have with the documents of any class according to the semantic relations. Each semantic class has number of properties and relations. In order to become a member of the class, the document should possess certain number of relations of the class. To measure the semantic bounding measure, the term set Ts of the document D, and the semantic ontology O has been used. First, the number of semantic features present in the document towards a class has been estimated.

Similarly, the number of relations present in the document for a specific class. Using these two, the semantic bonding measure has been estimated.

The number of semantic features present in the document D has been measured as follows:

$$NSF=\int_{i=1}^{size(Ts)} \int_{j=1}^{size(O(Class))} \sum Ts(i) == O(Class(j)) \quad (4)$$

Next, compute the number of relations present NRP in document D.

$$NRP=\int_{i=1}^{size(Ts)} \int_{j=1}^{size(Relations(O(Class)))} \sum Relations(j) \in Ts \quad (5)$$

Using these two, the semantic bounding measure (SBM) has been measured as follows:

$$SBM = \frac{NSF}{Size(Ts)} \times \frac{NRP}{size(Relations(Class))} \text{-- (6)}$$

The estimated semantic bounding measure has been used for the clustering of the document and classification.

Topical Closure Measure

The topical closure measure represents the documents strength in discussing the topic of the class. The document would contain number of terms and features. But in order to get assigned to a category, it should discuss the features of the topic. For example, if the document should come under the class of "Data Mining", it should discuss the topic in detail. The previous algorithms would select the class, if the document just speaks about the topic. However, there will be other features which may not relate to the topic considered. This really affects the fitness of the document to the class considered. The Topical Closure measure represents the fitness of the document for the class. To compute the TCM value, the taxonomy of topics has been used. First, the method computes the topical coverage measure (TcoM) for different classes. Using the value estimated, the method estimates TCM measure as follows:

First the topical coverage measure TcoM as below:

$$TcoM = \frac{\sum_{i=1}^{size(Ts)} Ts(i) \in Taxonomy(C)}{size(Taxonomy(C))} \quad (7)$$

Second the topical coverage measure for other class as follows:

$$TcoM_a = \frac{\sum_{i=1}^{size(Ts)} Ts(i) \in Taxonomy(OC)}{size(Taxonomy(OC))} \quad (8)$$

Finally, the topical closure measure TCM has been estimated as follows.

$$TCM = \frac{TcoM * TcoMa}{size(Ts)} \quad (9)$$

Multi-Level Semantic Bound Clustering

The multi-level semantic bound clustering algorithm reads the input document set given. The input document set and each document of it has been preprocessed to extract the term set and sentence set. Using the term set, the method estimates the semantic bound measure (SBM) for each level or stage considered. Similarly, the method compute the Topical Closure Measure (TCM) for each level considered. Using the estimated values of SBM and TCM, the method computes the MSBM measure for the input document. The class or level which has higher MSBM value has been selected for indexing.

Algorithm

Input: Document Set Ds, Ontology O, Taxonomy T

Output: Cluster Cs

Start

Read document set Ds.

For each document Di

Term Set Ts = preprocessing (Di)

For each class C

For each Level

SBM = Compute Semantic Bound (Ts, O, T)

TCM = Compute Topical Closure Measure (Ts,T)

Compute MSBM = SBM×TCM. -- (10)

End

End

C = Choose the class with Higher MSBM value.

Index Document Di to Class C.

EndStop.

The above discussed algorithm estimates multi-level semantic bound measure to identify the class of the document.

Multi-Level TSC Merging

The proposed algorithm merges the related documents based on the topical support closure. To perform this, the method estimates Topical Support Closure measure between each document. For each document of the cluster, the method estimates TSC value towards each document. Based on the value of TSC the methods select the similar documents and merge them.

Algorithm

Input: Cluster C

Output: Cluster C

Read C.

For each document Di of C

For each Document Dk of C

Term Set Ts = Preprocessing (Di)

Term Set Ts1 = preprocessing (Dk)

Compute TSC = $\frac{\sum_{i=1}^{size(Ts)} \sum_{j=1}^{size(Ts1)} Ts(i)==Ts1(k)}{size(Ts)}$ -- (11)

If TSC>Th then

Di= Merge (Di, Dk)

end End End

The above discussed algorithm computes the topical support closure between different documents. If the value of TSC is greater than threshold then it will be merged.

III. RESULTS AND DISCUSSION

The proposed storage management and SBM measure has been implemented and evaluated for its performance. The method has been implemented in real time cloud environment like Microsoft Azure with Java programming language. The results produced by the proposed algorithm has been measured and compared with other methods. The performance of the method has been measured in clustering, accuracy, time complexity, Recall.

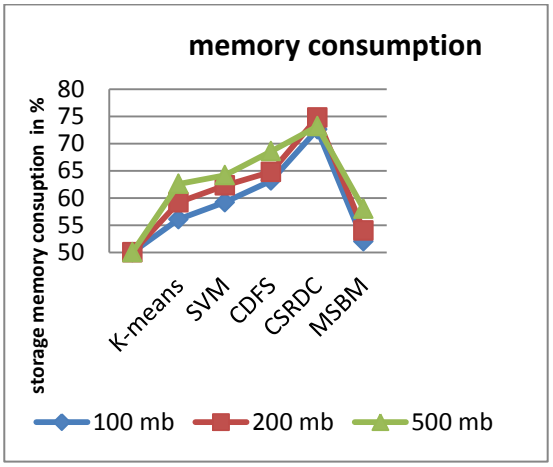

Figure 2: comparison of memory consumption

The result of performance analysis memory consumption by different methods has been presented in Figure2. The proposed method reduces the memory consumption by reducing data replication.

Table 1: comparison of the Memory consumption

	Comparison of memory consumption (%)			
Techniques /datasets	K-means	SVM	CDFS	MSBM
100 mb	56.1	59.2	63.2	52.2
200 mb	59.2	62.3	64,8	56.3
500 mb	62.6	64.2	68.6	59.4

The performance analysis on memory consumption on varying size of data set has been measured and compared. The comparative result is presented in Table 1. The proposed system reduces the storage consumption than other methods.

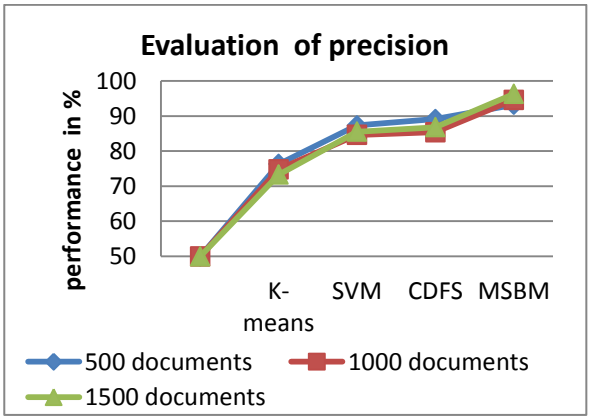

Figure 3: Evaluation of a precision rate

Figure 3, shows the comparative result on precision produced by various methods. The proposed MSBM algorithm has produced higher precision in different number of documents.

Table 2: comparison of the precision rate

	Impact of precision in %			
Techniques /datasets	K-means	SVM	CDFS	MSBM
500 documents	76.3	87.3	89.1	93.2
1000 documents	74.8	84.6	85.4	94.6
1500 documents	73.2	85.5	86.8	96.3

Table 2, evaluation of precision rate which this system produce higher efficiency compared to another system. The proposed MSBM system produce up to 96.3 % well accuracy than other methods.

Table 3: comparison of recall

	Impact of recall in %			
Techniques /datasets	K-means	SVM	CDFS	MSBM
500 documents	76.3	87.3	91.3	93.5
1000 documents	74.8	84.6	92.2	94.2
1500 documents	73.2	85.5	94.6	97.3

The above table 3 shows the comparison of recall state analyses by various techniques. The proposed MSBM system produces higher recall state of evaluation up to 97.3 %compared to the other system.

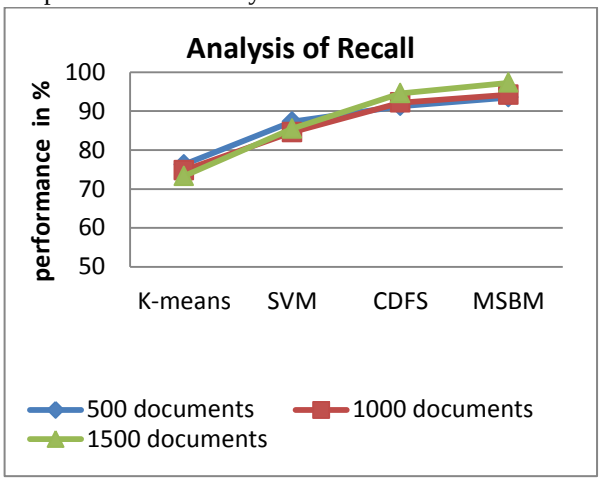

Figure 4: Comparison of recall

The performance on recall has been measured for various methods. The proposed MSBM algorithm has improved the performance on recall higher than other methods.

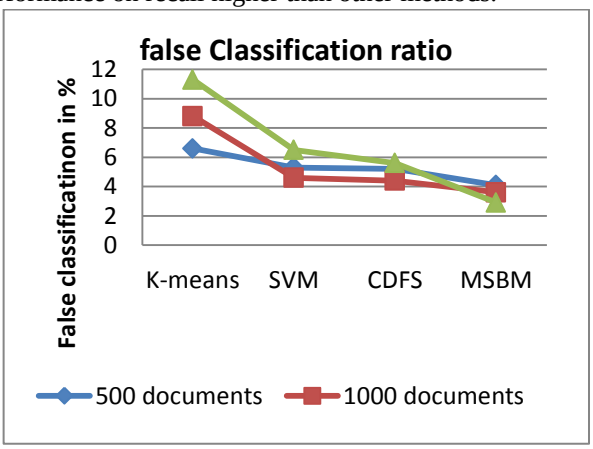

Figure 5: Comparison of false classification

The ratio of false classification produced by different methods has been measured and presented in Figure5. The result shows that the proposed MSBM algorithm produces less false classification ratio than other methods.

Table 4: comparison of false classification ratio

	Comparison of false classification in %			
Techniques /datasets	K-means	SVM	CDFS	MSBM
500 documents	6.6	5.3	5.2	4.1
1000 documents	8.8	4.6	4.4	3.6
1500 documents	11.3	6.5	5.6	2.9

The Table 4, shows the comparison of the mean extraction ratio higher level of the state compared to the other methods .the evaluated performance shows that the proposed MSBM approach produces less false extraction ratio up to 2.9 % well.

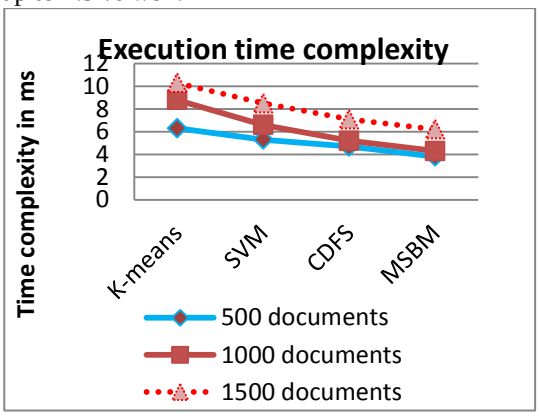

Figure 6: Execution Time Complexity

The time complexity produced by different methods have been measured and presented in Figure 6. The proposed algorithm has reduced the time complexity than other methods.

Table 5: Execution of time complexity

	Execution time complexity in seconds (ms)			
Techniques /datasets	K-means	SVM	CDFS	MSBM
500 documents	6.3	5.3	4.7	3.8
1000 documents	8.8	6.6	5.2	4.3
1500 documents	10.3	8.5	7.1	6.2

The above table 5 shows the time complexity of the dataset analyzed with dissimilar methods has a various preference of proposed method. The implementation of the proposed method produces a higher performance with lower complexity in 2.8 (m/s).

IV. CONCLUSION

In this paper, an efficient data storage management system for cloud environment is presented. The proposed, multi-level semantic bonding measure based clustering algorithm starts with preprocessing with the input document set. Then the method estimates the semantic bonding measure (SBM) for each class at each level. Also, the method estimates the topical closure measure (TCM) for each class at each level. Using these two measures, the method compute the MSBM measure. Based on the value of MSBM the method identifies the class of the document and index the document to the selected cluster. Similarly, the method computes Topical Support Closure (TSC) value for each document with other documents of the cluster. If there is any document with the value TSC less than specific threshold, then the documents are considered as more similar and replicated. Such documents are merged to reduce the redundancy and reduce the storage complexity. The proposed algorithm improves the performance of clustering and reduces the time complexity and false classification ratio.

REFERENCES

1. Yoji Yamato, Fast and Reliable Restoration Method of Virtual Resources on Open Stack, IEEE Transaction on Cloud Computing, 6, 2, 2018.
2. Y. Yamato, Y. Nishizawa, M. Muroi, K. Tanaka, "Development of resource management server for carrier IaaS services based on Open Stack", J. Inform. Process. vol. 23, no. 1, pp. 58-66, Jan. 2015.
3. Joonsang Baek, A Secure Cloud Computing Based Framework for Big Data Information Management of Smart Grid, IEEE Transaction on Cloud Computing, 3, 2, 2015.
4. J. Baek, Q. Vu, A. Jones, S. Al Mulla, C. Yeun, "Smart-frame: A flexible scalable and secure information management framework for smart grids", Proc. IEEE Int. Conf. Internet Technol. Secured Trans., pp. 668-673, 2012.
5. HaoJin, Dynamic and Public Auditing with Fair Arbitration for Cloud Data, IEEE Transaction in Cloud Computing, 6, 3, 2018.
6. Manabu Hirano, LogDrive: a proactive data collection and analysis framework for time-traveling forensic investigation in IaaS cloud environments, Springer, Cloud Computing, 7,18,2018.
7. Lijun Fu, Dynamic thermal and IT resource management strategies for data center energy minimization, Springer, Journal of Cloud Computing, 6:25, 2017.
8. De Lucia, A. Clustering Algorithms and Latent Semantic Indexing to Identify Similar Pages in Web Applications. In Web Site Evolution, 2007.
9. Lamprier, "Using Text Segmentation to Enhance the Cluster Hypothesis," Springer, vol. 5253: pp. 69–82, 2008.
10. B.Peng, "Implementation Issues of A Cloud Computing Platform," IEEE Computer Society Technical Committee on Data Engineering, 2, 1672-1694, 2009.
11. M. Armbrust, "AView of Cloud Computing," Comm. ACM, 53, 4, pp. 50-58, 2010.
12. Turney, 'From frequency to meaning: Vector space models of semantics,' Journal of artificial intelligence research, 37, 1, 141–188, 2011.
13. G. Xu, "Expander code: A scalable erasure-resilient code to keep up with data growth in distributed storage," IEEE 32nd International performance computing and communications conference. pp. 1-9, 2013.
14. Yung-Shen Lin; "A Similarity Measure for Text Classification and Clustering IEEE Transactions on Knowledge and Data Engineering, pp 1575 – 1590. 26, 7, 2014.
15. S.S. Sengar, "E-DAVID: An Efficient Distributed Architecture for In-Line Data De-duplication," IEEE International Conference on Communication Systems and Network Technologies. pp. 438-442, 2012.