HCMX: AN EFFICIENT HYBRID CLUSTERING APPROACH FOR MULTI-VERSION XML DOCUMENTS

VIJAY SONAWANE[1], D.RAJESWARA.RAO[2]
[1]Research Scholar, Department of CSE, K.L.University, Green Fields, Guntur, Andhra Pradesh
[2]Professor, Department of CSE, K.L.University, Green Fields, Guntur, Andhra Pradesh
E-mail: vijaysonawane11@gmail.com[1], rajeshduvvada@kluniversity.in[2]

ABSTRACT

In order to retrieve useful information from large number of growing XML documents on the web, effective management of XML document is essential. One solution is to cluster XML documents to find knowledge that promote effective information management and maintenance. But in the real world XML documents are dynamic in nature. In contrast to static XML documents, changes from one version of XML document to another version cannot be predicted. So clustering technique of static XML documents cannot be used to cluster multiple versions of XML documents. In case of multiversion XML documents, preliminary clustering solution is not become valid after document versions appear. XML documents are self descriptive in nature, which results in large document size. To find new clustering solution after change, comparisions between all documents is not viable solution. In this paper we have proposed hybrid clustering approach to cluster multiversion XML documents. This approach improves speed of clustering by limiting the growing size of XML documents by using homo-morphic compression scheme and using distance information from preliminary clustering solution with the changes recorded in compressed delta

Keywords: *HCMX, Hybrid Clustering, Cluster re-evaluation, Multiversion, PCP, CSRP, compressed Delta.*

1. INTRODUCTION

With the huge growth of the Internet, XML has now become a universal standard for information representation and exchange on over Internet [1]. Due simple and flexible nature of XML, various applications like scientific writing and technical papers to handle news summaries [2] use XML in information exchange and representation. XML is also used to represent the web based free content encyclopedia known as Wikipedia, it has more than 3.4 million XML documents. XML offers many features of online business functions such as content integration and intelligence.

The growing popularity of XML has lift up many concerns regarding the methods of how to efficiently maintain and manage the XML data and retrieve these XML documents in large collections. One of the feasible solutions to handle large XML collection is to make group of similar XML documents in the form of cluster. Clustering similar XML documents is supposed to be one of the more effective way for document handling by facilitating better information retrieval, data indexing, data integration and query processing .The clustering of similar XML documents has been perceived as potentially being one of the more effective solutions to improve document handling by facilitating better information retrieval, data indexing, data integration and query processing [3].

But there are several challenges in clustering XML documents. Unlike the clustering of text documents or normal data, clustering of XML documents is a complex process [4] and as a result the most commonly used clustering methods for text clustering cannot be replicated for clustering these documents.

This is due to the fact that changes in XML documents are application specific. In real world content of XML documents is dynamic in nature and changes in it are limitless and not predictable [5].Every time changes in original XML document gives birth to the new version of it. Dynamic XML documents are applicable in many fields of information maintenance and management, so it creates the demand for multiversion support [6].So it is also necessary to store different versions of XML documents. In spite its potential, storage of all the versions of an XML document increases the redundancy and make searching and querying harder on growing documents collection. In clustering solution of multiversion XML documents, new document version is not available in advance, or not completely new documents. Only they are different in certain degree from their previous one. XML is self describing in nature, this provides enormous flexibility to it, but also introduces the problem of "verbosity" which results in huge document size [7] and results in poor response time.

In this paper we introduces an hybrid clustering approach named HCMX to find clustering solution of multi-version XML documents which changes dynamically. In this we judge the amount of document affected, rather than considering each coming document as new version. To find new clustering solution, after changes in preliminary one, instead of comparing all members of clustering solution, we make the use of distance information during preliminary clustering phase, with the changes responsible for document version to produce. To improve clustering speed and response time homomorphic compression scheme used which retain documents original structure.

2. MOTIVATION

With the increasing number of XML documents on web the need becomes usual to properly organise these XML documents in order to retrieve useful information. The deficiency of such an effective organisation of the XML documents causes a search of the total collection of XML documents, which results in poor response time. So to efficiently maintain XML document collection, it is crucial to cluster these documents based on their similarity. Along with the information retrieval, clustering can also be used to discover knowledge for web mining and query execution [12].

XML documents clustering process is not as straight forward as the process of the clustering of text documents [10]. It has to pass many challenges in the clustering of XML documents due to nature of XML documents. That are: 1) XML documents structure are having hierarchical relationship with its elements, so this relationship should be preserved. 2) In real-world, the difference between consecutive versions of an XML document varies, so static XML document clustering technique cannot be used [11].

3. METHODOLOGY

Clustering solution S of given set of XML document is represented as complete graph with S*(S-1)/2 number of weighted edges. To find single link clusters of level L, the edges with weight w ≥L have to remove, remaining connected edges gives resultant cluster [8]. If correspondence between two XML documents is used as measure to find clustering solution then weight of the edges connecting documents symbolize the distance between them.

Preliminary Clustering Phase (PCP) - (one time execution): Figure-1(a). Shows preliminary clustering phase, in this we first: i) compress input XML documents using homo-morphic compression technique (XGrind in our case). ii) Find cluster using any distance based clustering algorithm. iii) Save distance matrix between all XML document pair with the set of operations analogous to each minimum distance.

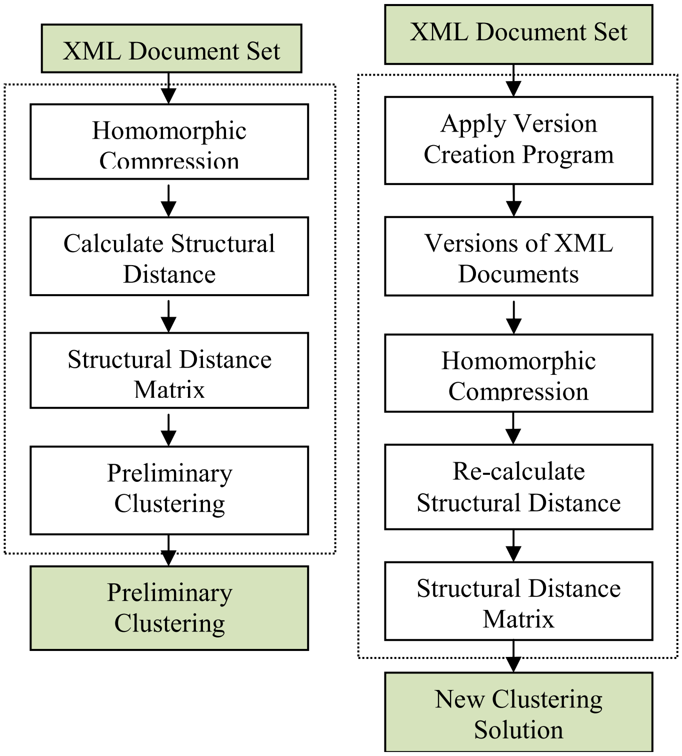
XML Document Set
Homomorphic Compression
Calculate Structural Distance
Structural Distance Matrix
Preliminary Clustering
Preliminary Clustering
XML Document Set
Apply Version Creation Program
Versions of XML Documents
Homomorphic Compression
Re-calculate Structural Distance
Structural Distance Matrix
New Clustering Solution

Figure-1. a) Preliminary Clustering Phase. b) Expensive Re-Clustering Method

Distance:- To get the distance between two XML documents, edit operations *Op (insert, delete, update)* are used with least cost value that transform one document into other.

$$d(D1, D2) = \text{least}\ ((\text{dist}(D1 \rightarrow D2\ ),\text{dist}(D2 \rightarrow D1)))$$

Least distance between two documents indicates larger resemblance between them. Two documents are said to be equal when total cost of operation is equal to zero. Any amount of changes in any document in preliminary clustering solution will affect the distance with rest documents in initial cluster.

If distance between two documents is smaller, it indicates the high resemblance between them. Hence when total cost operation between two document is equal to zero then these document are said to similar. When any document in preliminary clustering solution changes then its distance with rest of the documents in the cluster also changes. These number and type of changes are responsible for document residence in the same cluster or form its own cluster. To find new clustering solution after change in documents in preliminary cluster, reassessment of modified distances between all XML documents pairs is must.

Figure -1(b). shows one of the possible solution. As shown, to get new clustering solution after changes in preliminary clustering solution, comparison between all the documents pair is possible (Full Comparison). But it is not cost effective solution because i) It incur redundancy in calculating the distances between each pair of document by making full comparison between the documents. ii). It does not consider degree of changes in the document, most of time new version may not modified at all or may carry small amount of change, hence most of the operations are needlessly repeated.

Our proposed HCMX approach is depicted in figure-2. HCMX is divided into two stages.

1) **Preliminary Clustering Phase**- This phase forms the base for HCMX.
2) **Clustering solution re-evaluation phase (CSRP):** This phase is repeated whenever documents from preliminary clustering solution changes. 1) Use set of changes from preliminary clustering phase and current time stamp, recorded in compressed delta. 2) Read distance matrix saved during preliminary clustering phase. 3) Re-evaluate the distances between documents based on distances calculated during PCP, changes responsible for document version recorded in compressed delta and least cost direction. Output of this phase form the base for next iterative run of CSRP.

The most important part in this CSRP is re-evaluating the distances based on distances calculated during PCP and set of changes recorded in compressed delta. Next section of the paper presents the method to achieve this.

4. DISTANCE MEASUREMENT

Hierarchical relationship between elements of XML document makes easy to perform operation on documents. Homomorphic compression scheme used maintains document original structure with reduced in size. In multi-version XML documents, new versions are obtained by applying insert, update and delete operations in combination on document version nodes and sum of these operations are stored in compressed delta.

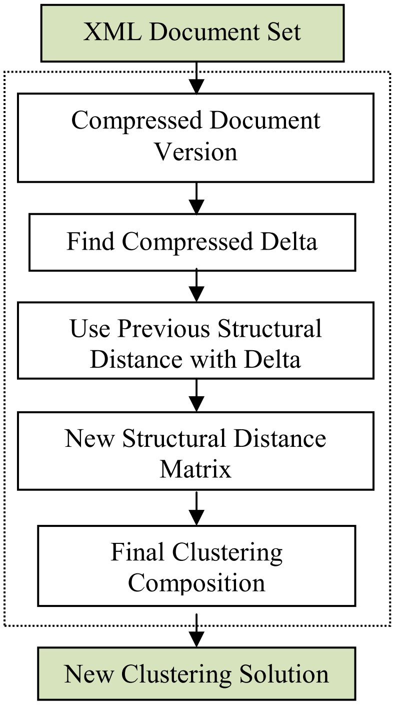
XML Document Set
Compressed Document Version
Find Compressed Delta
Use Previous Structural Distance with Delta
New Structural Distance Matrix
Final Clustering Composition
New Clustering Solution

Figure -2. HCMX clustering approach.

Compressed Delta (C∆) - Given dynamic XML document D with its version D*, compressed delta records the changes from one state of document to another. It consists of a set operations *Op(insert, delete, update)* execution of it on document D will return document in state D*.

Cost Compressed Delta (CC∆) – It is sum of the operation *Op(insert, delete, update)* recorded in compressed delta, those are responsible to convert D in to D*.

Inverted Operation (O^{invert}) - Given dynamic XML document D with its version D*, If an execution of operation Op (insert, delete, update) on document D returns its version D*, then execution of inverted operation on version D* returns original document D. i.e insert is inverted operation of corresponding delete.

If preliminary clustering solution S contains D1 and D2 with d(D1,D2) is distance between them and (D1→D2) is least cost direction and set of changes saved in compressed delta (responsible for document versioning) , then new distance d* can be defined as:

1. If set of changes stored in CΔ transform D1 into D1* then new distance between D1* and D2 can be defined as :

$$d^*(D1^*, D2) = (d(D1,D2) \cup C\Delta(D1, D1^*)) - (D1,D2) \cap C\Delta(D1,D1^*))$$

i.e when D1 changes to D1* then to find its new distance with D2, common set of operations that transforms D1 to D1* with least distance need to be deducted as their effect is equal, so only unique operations need to considered.

2. If set of changes stored in CΔ transform D2 into D2* then new distance between D1 and D2* can be defined as :

$$d^*(D1,D2^*) = d(D1,D2)+C\Delta(D2,D^*) - \sum_{i=1}^{q} Q^i + Q_i^{invert}$$

Here Q^i are the q operations from d(D1,D2) which have consequent inverted operations O_i ^{invert} in CΔ (D2, D2*) , 1≤ i ≤ q. i.e set of operations which gives least distance between D1 and D2, and were subsequently inverted during D2 transformation into D2* need to be deducted when calculating the distance between D1 and D2, as their combined effect is null, whereas only the distinct non-inverted operations need to be considered.

3. If set of changes stored in $C\Delta_1$ transform D1 into D1* and set of changes stored in $C\Delta_2$ transform D2 into D2* then new distance between D1* and D2* can be defined as :

$$d^* (D1^*, D2^*) = [(d(D1,D2) \cup C\Delta_1(D1,D1^*)) - (d(D1,D2) \cap C\Delta_1(D1,D1^*)] + C\Delta_2(D2,D2^*) - \sum_{i=1}^{p} Q^i + Q_i^{invert}$$

Here O_i are p residual operations from d(D1, D2) after removing repeated operations from $C\Delta_1$ which have consequent inverted operations O_i^{invert} in $C\Delta_2$ (D2,D2*), 1≤ i ≤ p. i.e when both D1 and D2 have changed into its consequent versions D1* and D2* , above both formulas are applicable to find new distance d*(D1*,D2*).

4. if both documents D1 and D2 do not change, then the new distance d* is the same with the previous distance d:

$$d^*(D1, D2) = d(D1, D2)$$

If there is no change in distance, it indicates the total similarity between the documents.

5. HCMX ALGORITHM

Input: Clustering solution of compressed dynamic XML documents.

Output: Re-evaluated clustering Solution.

For each compressed documents [D_1, D_2] belongs to clustering solution and $D_1 \neq D_2$.

```
1.1. if document $D_1$ changes but document $D_2$ does
     not change
     then shared_op_cost = 0;
       for each operation $O_i$ belongs to $d(D_1, D_2)$
         if operation $O_i$ is belongs to $\Delta(D_1,D_1^*)$
           then shared_op_cost= shared_op_cost +
             2*cost($O_i$)
     for next operation $O_i$, the new distance between
     $d^*(D_1^*,D_2) \leftarrow d(D_1, D_2)+cost(\Delta(D_1,D_1^*))-$
     shared_op_cost

1.2. if document $D_2$ changes but document $D_1$ does
     not change
     then inverted_op_cost = 0;
     for each operation $O_i$ belongs to $d(D_1, D_2)$
         if inverted operation $O_i^{invert}$ belongs to
         $\Delta(D_2,D_2^*)$
       then inverted_op_cost = inverted_op_cost +
         cost ($O_i$)+ Cost ($O_i^{invert}$)
for next operation $O_i$, the new distance between
$d^*(D_1,D_2^*) \leftarrow d(D_1,D_2)+cost(\Delta(D_2,D_2^*))-$
inverted_op_cost.

1.3. if both documents $D_1$ and $D_2$ changes
     then for each operation $O_i$ belongs to $d(D_1, D_2)$
         if operation $O_i$ is belongs to $\Delta(D_1,D_1^*)$
       then shared_op_cost = shared_op_cost + 2*
           cost($O_i$)
      if inverted operation $O_i^{invert}$ belongs to delta
     of $\Delta(D_2,D_2^*)$
      then shared_op_cost = shared_op_cost +
                 cost $O_i$ + cost ($O_i^{invert}$)

for next operation $O_i$, the new distance between
$d^*(D_1^*,D_2^*) \leftarrow d(D_1,D_2)+cost(\Delta(D_1,D_1^*))+cost(\Delta(D_2,$
$D_2^*))$ - shared_op_cost – inverted_op_cost

1.4. if both documents Di and Dj does not changes
 then new distance $d^*(D_1^*,D_2^*)$ will be same as
$d(D_1,D_2)$
```

Table-1. Experimental Results

No of Documents	Input documents size (Kb)	Size after compression (Kb)	% of Changes Applied	Clustering Time (millisecond)	
				Full Comparison	HCMX
50	500	350	10	3225	964
100	1000	700	20	5610	1465
150	1500	1040	50	8256	2104
200	2000	1390	80	11034	2811
250	2500	1740	100	13813	3520

6. EXPERIMENTAL EVALUATION

Proposed HCMX approach is implanted in java. To assess the performance of HCMX approach we used documents of variable size from XML data repository [9]. As shown in figure1. Preliminary clustering solution of given input XML documents are obtained by using least distance between all the compressed document pairs. Preliminary clustering solution of given input XML documents are obtained by recording and using least distance between all the document pairs with the set of operations corresponding to each least distance. New document versions are created with different percentage of changes by using version creation program. The main objective of evaluation is to assess time required to find new clustering solution when documents change their distance in preliminary clustering solution. We evaluated clustering time of HCMX with full comparison. Experimental results are shown in table1.

Result charts shown in figure-3. to figure-5. reveal that our proposed HCMX performs better in spite the increase in documents, their size and applied changes.

7. CONCLUSION

In this paper we have proposed hybrid clustering approach HCMX to cluster multi-version xml documents when preliminary clustering solution becomes outdated. To find updated clustering solution after document versioning comparision between all documents incur large amount of redundant operations. Our proposed HCMX approach judge amount of document affected and re-evaluate cluster by using effect of temporal changes recorded in compressed delta and distances recorded during preliminary clustering phase. Experimental results shows proposed approach is time effective, as used homomorphic compression scheme highly reduces the documents size and operations involved in cluster solution re-evaluation (CSRP) are reduced.

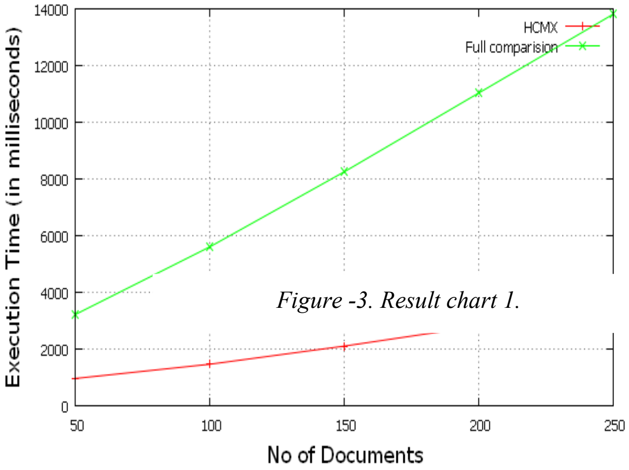

Figure -3. Result chart 1.

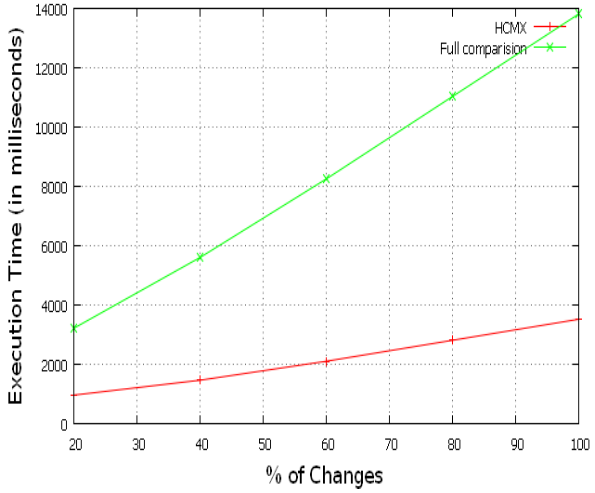

Figure -4. Result chart 2.

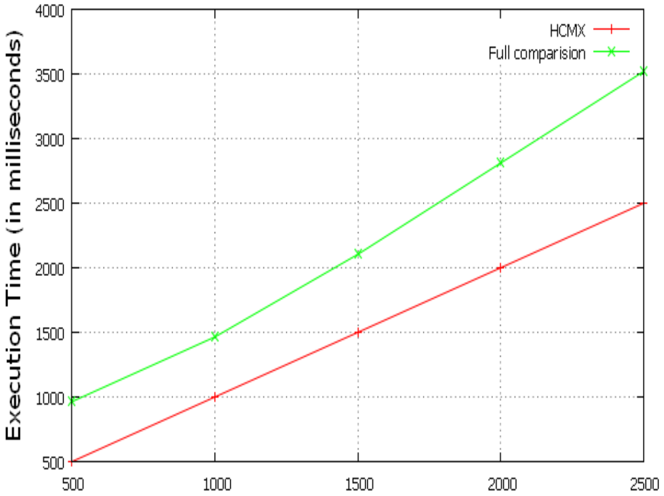

Figure -5. Result chart 3.

REFERENCES:

[1]. E. Wilde and R. J. Glushko. XML fever. J. Comm. ACM, 2008, (51) pp. 40–46.

[2]. A. Tagarelli and S. Greco. Semantic clustering of XML documents. ACM Transactions on Information Systems, Vol. 28, No1, 2009, pp. 1–56.

[3]. T. Tran, S. Kutty, and R. Nayak. Utilizing the structure and content information for xml document clustering. Lecture Notes in Computer Science. Springer Berlin Heidelberg. Volume 5631, 2009, pp.460–468

[4]. S. Kutty, T. Tran, R. Nayak, and Y. Li. Clustering XML documents using closed frequent subtrees: A structural similarity approach. Lecture Notes in Computer Science, Springer Berlin / Heidelberg. Volume 4862, 2008, pp.183–194.

[5]. Dyreson. C, and Grandi. F. Temporal XML, Database Systems, 2009, pp.3032-3035.

[6]. Sidra F., Mansoor S. Temporal and multi-versioned XML documents: A survey, Information Processing and Management, Vol. 50, No 1, 2014, pp.113-131.

[7]. Sherif Sakr. XML compression techniques: A survey and comparison. , Journal of Computer and system Science Vol 75, No 5, 2009 pp.302-322..

[8]. Dalamagas, T., Cheng, T., Winkel, K.J. and Sellis, T. Clustering XML documents by Structure. SETN LNAI Springer. 3025, 2004, pp. 12-121.

[9]. XML data repository, online at www.cs.washington.edu/research/projects/xmltk/xmldata

[10]. Vijay Sonawane and D.Rajeshwara Rao. An Optimistic Approach for Clustering Multi-version XML Documents Using Compressed Delta. International Journal of Electrical and Computer Engineering. Vol. 5, Issue 6. 2015 , ISSN- 2088-8708.

[11]. Vijay Sonawane and D.R.Rao. A Comparative Study: Change Detection and Querying Dynamic XML Documents. International Journal of Electrical and Computer Engineering. Vol. 5 No 4. 2015. ISSN- 2088-8708.

[12]. Baeza-Yates and R. Ribeiro-Neto, B. Modern information retrieval: The concepts and technology behind search. ACM Press/Addison-Wesley. 2011.

[13]. Gao,M and Chen F. Clustering XML Data Streams by Structure based on SlidingWindows and Exponential His-tograms. Proceedings of the international conference on advances in databases, knowledge, and data applications . 2013. pp.224-230.

[14]. Wuwongse V, Yoshikawa M., and Amagasa, T. Temporal versioning of XML documents. Proceedings of the Seventh International conference on digital libraries: International collaboration and cross-fertilization . 2004. pp.419-428.

[15]. Cavalieri F, Guerrini. G, Mesiti M. and Oliboni, B. On the reduction of sequences of XML document and schema update operations. Proceedings of the IEEE twenty seventh international conference on data engineering workshops . 2011. Pp.77-88.

[16]. A. Tagarelli. and S. Greco. Semantic clustering of XML documents. ACM Transactions on Information Systems. Vol. 28, No 1. 2010. pp. 1-56.

[17]. M.X. Gao, W.J. Yao, and G.J. Mao. Exponential histogram of cluster feature for XML tream. Journal of Beijing University of Technology. vol. 37, 2011. pp. 1242-1248.