ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

Dynamic Clustering And Mining Of XML Documents

[I]Rutuja S Shinde, [II]Sonali V Gunjal, [III]Darshana Mutadak, [IV]Rupali Sanap, [V]Prof.Abhale B.A.
[I,II,III,IV,V]IT, Department

Abstract
Clustering is a process that partitions data in such a way that homogeneous data items are grouped into sets referred to as clusters. Clustering dynamic XML documents when their content or structure changes over time. In real-world applications, the number of changes from one version of an XML document to another cannot be predicted. It's always possible that an initial clustering solution becomes obsolete after the modification take place. XML clustering algorithms is to calculate pair-wise distances between documents. A time-efficient technique requests the pair-wise distances to be determined in a timely manner. In case of clustered dynamic XML documents, if changes were or if they affected only some of the clustered documents, recalculating pair-wise distances every time would be highly redundant. In our system a time-efficient technique to reassess pair- wise distances between clustered dynamic XML documents which change in time, without performing redundant calculations but considering the previously known distances and the set of changes which might have affected the documents versions. XML mining includes both the structure from XML documents (XML doc). In multiversion XML documents, a distance are calculated between each incoming XML document and the existing clusters using the level structure. This distance is determined by matching the nodes from the incoming document to the nodes of the existing clusters. The similarity is determined at the cluster level, rather than pair-wise, for individual documents in the clusters. We used novel technique of determining how the knowledge discovered from initial XML documents changes in time when the documents structure fluctuates.

I. Introduction

We emphasize the novelty and usefulness of the technique proposed, for mining variable association rules from dynamic XML documents, as the user might faces a acute problem of keeping up to date with the interesting knowledge, discovered from the business data. Our proposal in this paper focuses on XML variable association rules and the technique involves a quick reanalysis of the effective and possible association rules based on the history of changes supported by the dynamic XML documents. An automated implementation of the proposed approach would keep the user informed at any time about the changes in the extracted knowledge, so he/she could take the proper business decisions without re-running each time specific mining algorithms.

XML document is use for data storage and data exchange between application types of XML documents: static XML documents and dynamic XML documents.

1. Static XML documents :-Static XML documents do not change or modify their content and structure over time. For example, an XML document containing details of paper presented at a conference.
2. Dynamic or multi-versioned XML documents:-Dynamic or multi-versioned XML documents can modify or change their structure and content over time. For example, if the content of an online banking were represented in XML format, it would change daily based on e-customer behaviour.

XML[Extendible Markup Language] has vital role in increasingly extension use of it as standard language forum formation representation and data exchange on the web. Most web applications deal with web data by translating them into XML document format, In order to organize these data efficiently grouping XML documents because of their structure, content and semantics hidden inside them is a possible solution. In mining literatures one organizing process is referred as clustering which group similar XML data across heterogeneous ones. Clustering is also called " unsupervised Clustering,learning ". It is an intelligent technique for mining XML documents has been utilised as an excellent way of grouping the documents by their content or structure [2]. A distance based XML clustering algorithms is use to calculate pair-wise distances between documents. naturally, a time-efficient technique requests the pairwise distances to be determined within a time. In case of dynamic or multi-version XML documents, the amount of changes between versions cannot be predicted. Therefore, in case of Clustering and Mining Multi-version XML Documents dynamic XML documents, if changes were little or if they affected only some of the clustered documents, recalculating pair-wise distances each time would be highly redundant.

We will propose a time-efficient technique to reassess pair-wise distances between clustered dynamic XML documents which changes in time, without performing redundant calculations. But it is consider the previously known distances and the set of changes which affected the documents versions. In distance-based clustering techniques, each object from the given set is first assigned to a cluster. Then, distances between pairs of clusters are computed, and the closest clusters (the most similar) are grouped to form a new (bigger) cluster. In other words, when two XML documents are more similar compared to other pairs of XML documents, the distance between them is smaller; hence,they can become members of the same cluster .Mining XML documents has only been approached so far from a static point of view.Techniques is used for extracting association rules, clustering or classifying XML documents have used for collection of static XML documents. We are looking to the issue of variable knowledge. which is identifying how the changes suffered by a multi-version XML document affect the initial discovered knowledge[2]. The novelty of our project is the first attempt to analyse this problem for XML documents, we are focusing on the knowledge in form of association rules. We will determines which of the initial association rules are still valid after a number of changes suffered by the XML documents,so we find out that which ones becameweaker or stronger, and even discovers new association rules, these validations

Application:

Clustering of Dynamic XML documents that is our system is use

for real time application like banking system where the data of XML file is store, update and delete.

II. What Is To Be Developed ?

We proposing an intelligent and time- efficient technique for reassessing the distances between clustered dynamic XML documents after they change, not by running full pair-wise comparisons but by calculating the effects of the changes on the previously known distances, that is on the distances before documents have changed[1].

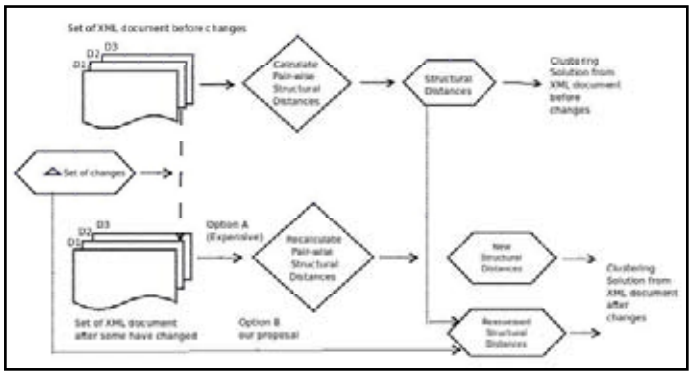


Fig. 1: Overview of the proposed technique to reassessed clustering solution composition

As shown in above figure 1 an overview of the identified problem. As it can be noticed, one straight forward option (option A) would be to recalculate, after each set of changes and the distances between the XML documents by doing a full pairwise comparison of them. This option would be very expensive from the operational point of view, because there is no distinction made between documents affected more or less by the set of changes; hence, in case of:

(i) New versions of documents carrying only a small amount of changes
(ii) Documents not modified at all, some or all operations involved in the full comparison of each pair of documents would be unnecessarily repeated.

The second option (option B - i.e our proposal) is to make use of the already known distances between pairs of XML documents in the clustering composition before the changes and the set of temporal changes, and use them together to reassess the new modified distances.In short we are going to perform following modules[1].

Example of similar and dissimilar XML documents

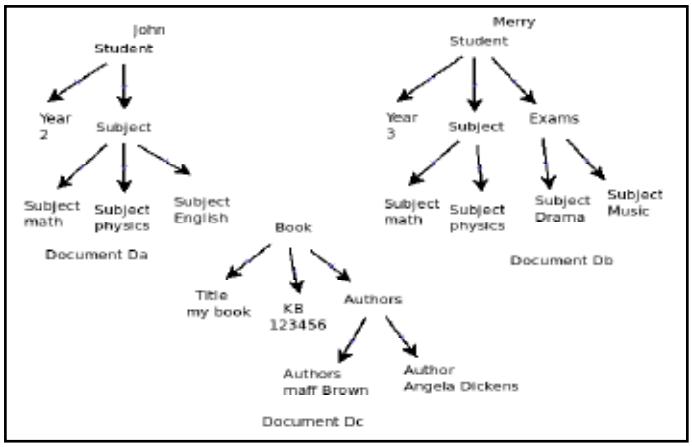


Fig. 2: Example of similar and dissimilar XML documents

Figure 2 explains the xml documents similarity. In this figure there are three XML documents in which, document DA and DB are highly similar. That is there is a similarity in a attributes of these documents, but the document DC that is the third document is not similar to either DA or DB. Documents DA and DB contains the information about two student John and Merry respectively. Which includes attributes like year of study, subjects, exams and student names. Where the third document DC list the information about book, which includes the attributes like title, ISB number, and author names. In Figure 1, any queries regarding students' details are applicable only to the relevant documents (that is, DA and DB) and not to any other documents which contain a different kind of information, such as DC. Intuitively, documents DA and DB are grouped in a cluster, while DC forms a cluster by itself.

A. Clustering Architecture

Clustering is very useful technique for grouping data objects such that objects within a single group or cluster have similar features, while objects in different groups are dissimilar. Architecture of an XML document clustering system can be illustrated as shown in Figure 3.

(1) Document preprocessing: documents are represented in a common data model then necessary preprocess is applied on structure and content of them to prepare them for extracting information for clustering. Different tasks are done based on the document representation.
(2) Similarity Measure: we should define an appropriate similarity measure due to the representation model in order to determine degree of similarity between pairs of objects.
(3) Clustering: the similar data objects are grouped together based on similarity measure using clustering algorithms. A lot of work has been done in clustering metric or spatial data, and several types of algorithms have been proposed. A few of them are:

Density-based algorithms

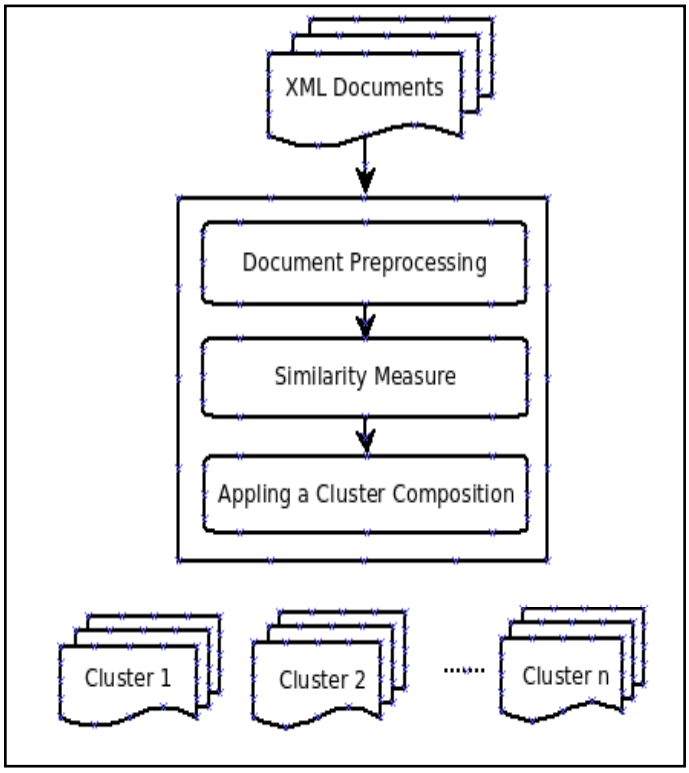


Fig. 3: XML document clustering architecture

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

Density-based algorithms

Density based clustering algorithm has played a vital role in finding non linear shapes structure based on the density. Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is most widely used density based algorithm. It uses the concept of density reachability and density connectivity[3].

Density Reachability

A point "p" is said to be density reachable from a point "q" if point "p" is within ε distance from point "q" and "q" has sufficient number of points in its neighbors which are within distance ε.

Density Connectivity

A point "p" and "q" are said to be density connected if there exist a point "r" which has sufficient number of points in its neighbors and both the points "p" and "q" are within the ε distance. This is chaining process. So, if "q" is neighbor of "r", "r" is neighbor of "s", "s" is neighbor of "t" which in turn is neighbor of "p" implies that "q" is neighbor of "p".

Algorithmic steps for DBSCAN clustering

Let $X = \{x_1, x_2, x_3, ..., xn\}$ be the set of data points. DBSCAN requires two parameters: ε (eps) and the minimum number of points required to form a cluster (minPts).

1) Start with an arbitrary starting point that has not been visited.
2) Extract the neighborhood of this point using ε (All points which are within the ε distance are neighborhood).
3) If there are sufficient neighborhood around this point then clustering process starts and point is marked as visited else this point is labeled as noise (Later this point can become the part of the cluster).
4) If a point is found to be a part of the cluster then its ε neighborhood is also the part of the cluster and the above procedure from step 2 is repeated for all ε neighborhood points. This is repeated until all points in the cluster is determined.
5) A new unvisited point is retrieved and processed, leading to the discovery of a further cluster or noise.
6) This process continues until all points are marked as visited.

III. Mining

In this mining, we have to search the documents with cluster and without cluster. In the mining with cluster, we have some threshold value to cluster. Then for searching the documents with cluster, we compare the distance of documents that is threshold value of documents with the threshold value of cluster. If the threshold value of document is less than the threshold value of cluster then search that document in a respective cluster. Otherwise search that document in the another cluster. For this overall process we XML documents and this process is called as XML mining and from that we will get the knowledge discovery of respective XML documents[4]. XML documents, featuring a very low degree of redundant information. The consolidated delta is built starting from the first (initial) version of the XML document, stored with each subsequent change which affects the document during its multi-versioned life on top of the initial document.

Our proposal in this paper uses the previously introduced concept of consolidated delta [6]. This is a way of storing valuable information from multi-versioned XML documents, featuring a very low degree of redundant information. The consolidated delta is built starting from the first (initial) version of the XML document, with each subsequent change which affects the document during its multi-versioned life being stored on top of the initial document. We assign unique identifiers to the initial nodes in the document, to be able to track the changes. More, at any time Tk, k>0, the information stored in the consolidated delta is enough to extract any historic document version Vi (0<i<k), by directly enquiring the consolidated delta document for changes at time Tk and searching backwards only for the unchanged nodes (see [6] for more details). In the next subsection, we will use the consolidated delta (which stores, as described above, the changes affecting the XML document during its multi-versioned life) to extract the sets of changes for any period of time, Based on this changes, our proposed algorithm identifies the variable (adaptive) association rules.

our proposed technique looks to discover what knowledge is available at time Tk, 0<k<n, based on the knowledge extracted at time T_i ($0 \le i < k$) and considering the effects of the C_{i+1}, C_{i+2},… C_k sets of changes, where any C_j ($0 < i < j < k$) is the set of changes between version V_{j-1} and version V_j of the document. The *flexibility* of our proposal resides in the

ability of using it with a initial knowledge available and its subsequent set of changes. For example, if some variable static knowledge was determined at time Tj based on the initial knowledge discovered at time T_0 and the C_1, C_2,…C_j set of changes, at time k (k>j>0) there would be no need to reiterate the process of using T_0 initial rules and C_1, C_2…C_k set of changes, instead we use the latest determined rules (at time T_j) and get the knowledge at time T_k by considering only the effects of C_{j+1}, C_{j+2},…C_k sets of changes (see Figure 1). In Figure 4, SK stands for "static knowledge". The process of discovering variable knowledge at any time T_k, 0<k<n (where *n* is number of versions of the XML document), involves two major steps, where the first one, i.e. preparation step, is performed only once, to extract the initial knowledge (e.g. the first set of valid association rules).

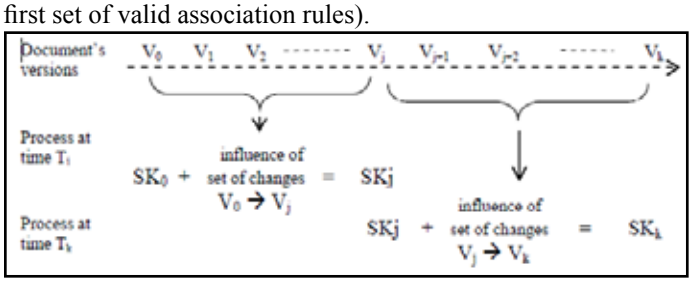
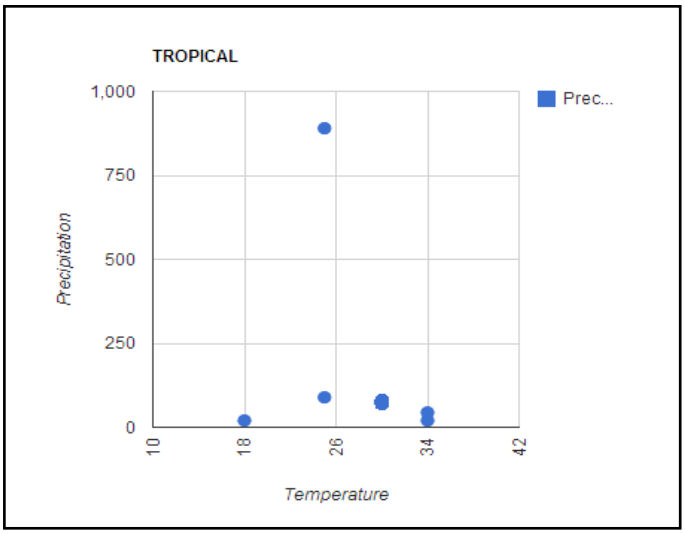
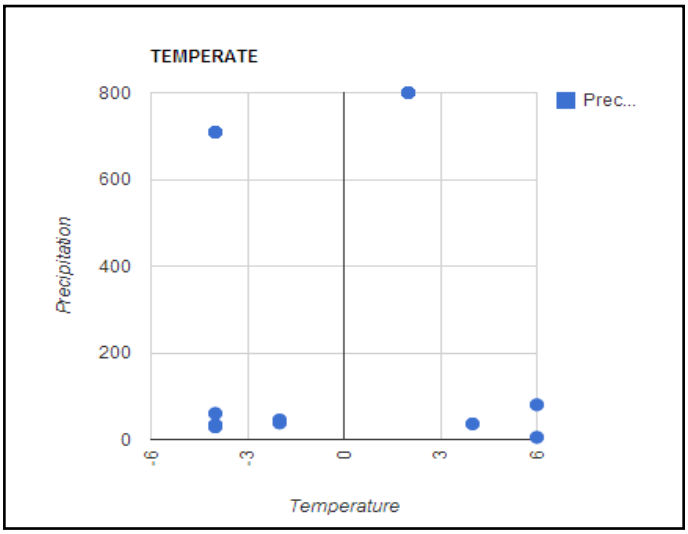


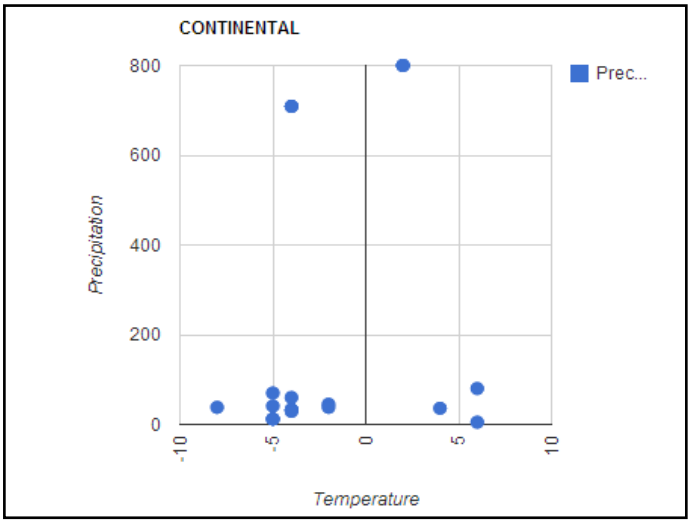
Fig. 4: Overview of the general process of determining variable knowledge

IV. Experimental Results

To test our proposed method we used XML documents extracted from the [5], with an weather level. Firstly, we clustered the documents to get the initial clusters composition , using minimum pair-wise distances; at this stage we also stored the distances between documents in the clustering solution together with the set of operations corresponding to each minimum distance. Then, we applied different attribute of changes to the documents in the clustered solution, in order to obtain new versions. The purpose of the tests was to compare, after each set of changes, the time required to reassess the distance between documents using the same method as for the initial clustering.To show the graph using the value of attribute ranges (like humidity& temperature) as shown in table 1.







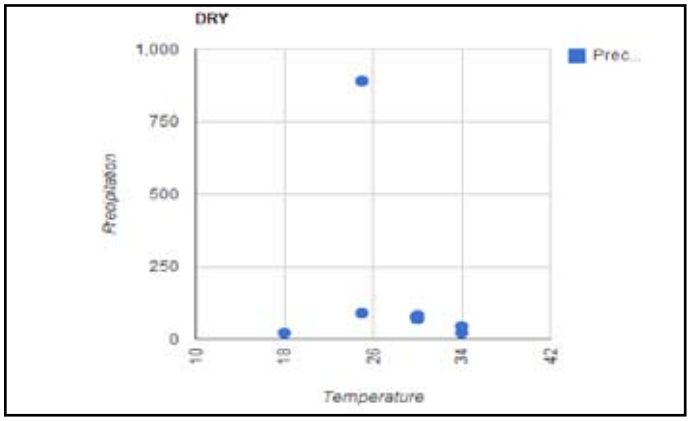


Table 1: Different region & ranges of graph

Region	Temperature Range
Tropical	15 to 35
Dry	10 to 35
Temperature	-5 to 15
Continental	-10 to 15

V. Conclusion

In this paper we have proposed an intelligent and efficient technique to reassess the distances between dynamic XML documents when one or all of the initially clustered documents have changed. After the changes, the initial clustering solution might become obsolete - the distances between clustered XML documents might have changed more or less, depending on the degree of modifications (insert, update, delete) which have been applied. Re-running full pair-wise comparisons on the entire set of modified documents is not a viable option, because of the large number of redundant operations involved. Our proposed technique allows the user to reassess the pair-wise XML document distances, not by fully comparing each new pair of versions in the clustering solution, but by determining the effect of the temporal changes on the previously known distances between them. This approach is both time and I/O effective, as the number of operations involved in distance reassessing is greatly reduced.

References

[1] *Rusu L.I., RahayuW. and Taniar D., Intelligent Dynamic XML Documents Clustering, In Proceed of The 22nd Inter national Conference on Advanced Information Networking and Applications.(IEEE-2008).*

[2] *Rusu L.I., RahayuW. and Taniar D., Extracting Variable Knowledge from Multiversioned XML Documents, In Proceed of The 6thInternational Conference onData mining.(IEEE-2006).*

[3] *Density-based clustering algorithms – DBSCAN and SNN by Adriano Moreira, Maribel Y. Santos and Sofia Carneiro.*

[4] *Laura Irina Rusu, XML data mining, Part 3: Clustering XML documents for improved data mining, May 2012.*

[5] *XML data repository, online at http://www.cs.washington. edu / research / projects / xmltk /xmldata.*

[6] *Mining XML Documents with Association Rule Algorithm âˇA ¸S Gorkem Gurel.*