Clustering XML Document Based On Path Similarities Using Structure Only

EiEiMon
University of Computer Studies, Yangon
eieimonucsy@gmail.com

Khin Nweni Tun
University of Computer Studies, Yangon
knntun@gmail.com

Abstract

We propose a methodology for clustering XML documents on the basis of their structural similarities. This research combines the methods of common XPath and K-means clustering that improve the efficiency for those XML documents with many different structures. The common XPath is used for searching similarities between huge numbers of XML documents' paths. K-means clustering algorithm is essentially used to accurate clusters. In order to cluster the documents' paths we indicate the steps by step methods. The first step includes frequent structure mining for searching similarities between the huge amounts of XML documents' structures by using the F-P growth method. The second step builds dimensional feature vector matrix by using extracted paths. Based on the set of common path vectors collected, we compute the structure similarity between the XML documents. And the last step utilizes the K-means clustering algorithm is used to create accurate clusters which are based on the idea of using path based clustering, which groups the documents according to their common XPaths, i.e. their frequent structures. The quality of clustering can be measured on the dissimilarity of document structures. Also, experimental evaluation performed on both synthetic and real data shows the effectiveness of our approach.

Keywords: common XPath, K-means clustering, XML Document Clustering, Data Mining, Frequent Structure Mining

1. Introduction

As the heterogeneity of XML sources increases, the need for organizing XML documents according to their structural features has become challenging. The detection of structural similarities among documents can help in solving the problem of recognizing different sources providing the same kind of Information. Structural analysis of Web sites can benefit from the identification of similar XML documents, which can serve as the input for wrappers working on structurally similar Web pages. The XML language is becoming the standard Web data exchange format, providing interoperability and enabling automatic processing of Web resources. While the processing and management of XML data are popular research issues [1], operations based on the structure of XML data have not yet received strong attention. Applying structural transformations and grouping together structurally similar XML documents are examples of such operations. Structural transformations are the basis for using XML as a common data exchange format. Grouping together structurally similar XML documents refers to the application of clustering methods using distances that estimate the similarity between tree structures in terms of the hierarchical relationships of their nodes.

The main contribution of this work is a methodology for grouping structurally similar XML documents. Data mining approach to XML document clustering is pursued. Thus, data mining is treated as a feature extractor for documents clustering. The concept of frequent tree pattern for determining XML similarity was introduced in [2], [3], [4] and [5]. This approach makes use of data mining techniques to find the repetitive document structure for determining the similarity between documents. In [5], another attempt to mine maximal frequent tag tree patterns in semi structured documents is reported. In [2], the structural similarity is defined as the number of paths that are common and similar between the hierarchical structure of an XML document using automata and determine the frequent path of a tree using an adapted sequential mining approach. In order to mine the frequent tree patterns or structural sequences, all XML-sequences are extracted and then mining of the common frequent XPaths.

The Clustering for the whole structure of XML documents and all XPaths can sparse the feature vector. To solve this problem, this research combines the methods of common XPath and K-means clustering of partitioning that improve the efficiency for those XML documents with many different structures. Recent studies have proposed techniques for clustering XML documents. In clustering XML documents need to consider both element and its structure. XML paths can represent both element tags and their position information and express the structure of XML. A path represents the tags from root node to terminal node. It includes the XML tags but also reflecting the structure of the XML documents. Using XML documents path feature is a good method to compute the similarity among XML

documents. To reduce the number of path features, [6] used apriori algorithm to find the frequent paths and take these frequent paths as XML document feature that called common Xpaths. By using all paths less than or equal to length L is a user-specified parameter value as feature vectors for XML documents [7] is usually sparse, due to the feature vector matrix. The similarity matrix made up of path feature vector is very big and sparse. The feature vector matrix is a high-dimensional matrix in which many entries are zero, its row numbers and column numbers are very big. To reduce the dimensionality of the vectors, [8] use principal component analysis (PCA) to identify significant dimensions and condense the matrix. [6] use aprior algorithm to mining the frequent Xpath as feature. [6] and [8] both use an approximate method to reduce the dimensional space.

The rest of the paper is organized as follows. In section 2, the first step of generating XPaths and Mining frequent XPaths with Frequent Patten Growth are described. Session 3 discusses building the matrix for XPaths and the section4 provides with the K-means clustering methods for measuring similarity between XML documents. Finally, we conclude the paper. Figure 1 shows the overview of clustering XPaths structure.

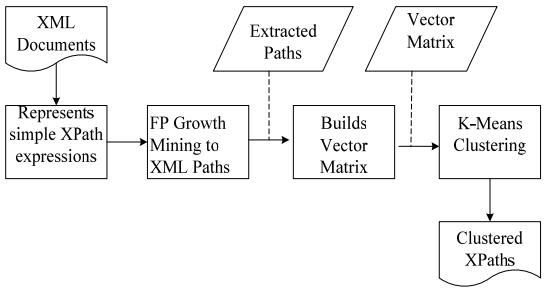

Fig:1 Step by Step approach for clustering XPaths Structure

2. Finding Duplicated Path with Frequent Pattern Growth

For feature extraction, use path-based feature extraction. Each path contains the node properties from the root node to the leaf node. Frequent structure mining is an implication of mining methodology for frequent structures without candidate generation. It constructs FP-tree that is used highly compact data structure with FP-Growth algorithm (Frequent Pattern Growth) to compress the original documents structures. The resulting is greater efficiency than Apriori based algorithms. In preprocessing phase, an XML document is first parsed and modeled as the labeled tree. The tree is decomposed into XPaths to represent the structured path information called node paths of the XML document. XPath provides a way to describe the structure of a source document so that can transform the document. Duplicated XPaths in a document structure are eliminated. After the pre-processing of XML documents, documents are represented as a collection of distinct XPaths. The structural similarity between XML documents can be computed by determining the number of paths and their level of hierarchy that are similar. Similar documents can be grouped by the same cluster.

Figure 2 shows the correspondence between an XML document and its XML tree. In this paper XML document is represented as a labeled tree and the values of the elements or attributes in the tree will not be considered, i.e, only the structure of the XML document is considered.

```
<SigmodRecord>
  <issues>
      <issue>
        <volume>15</volume>
        <number>2</number>
        <articles>
          <article>
            <title articleCode="152033"> </title>
            <authors>
              <author AuthorPosition="03">F Andersen</author>
              <author AuthorPosition="04">H Blanken</author>
              <author AuthorPosition="02">K Kuespert</author>
              <author AuthorPosition="01">P Dadam</author>
            </authors>
          </article>
        </articles>
      </issue>
  </issues>
</SigmodRecord>
```

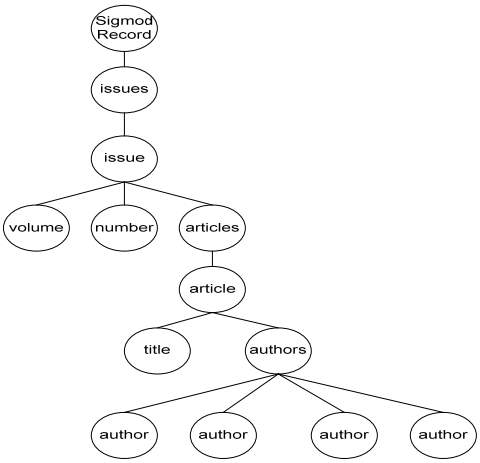

Fig:2 Example of XML tree generation

For searching the duplicate path identifier we apply the method for generating frequent itemsets without candidate generation in order to leading to good performance gain. Frequent-pattern growth or simply FP-growth adopts a divide-and-conquer strategy.

Table 1 shows execution time of 271 frequent sets which are generated by FP-growth Algorithm with Java implementation for frequent paths in Figure 2.

Confidence (%)	Execution Time (sec)
100	0.11
90	0.13
80	0.14
70	0.14
60	0.16
50	0.19

Table 1. The results for generating times with support count 2

3. Create Vector Matrix

Our clustering algorithm is based on the mining results of the common XPaths. As described in the previous section, the FP mining algorithm find the maximal common paths(i.e. maximal frequent sequences) from the XML paths of the documents for comparison. If the two documents are very similar to each other, more common paths are mined and have good matches to the extracted path from the original documents. Based on the mining results, the similarity between two documents can be measured by the maximal common paths.

XML document can be viewed as a labeled tree. In our case, we define here XML document tree *d* .

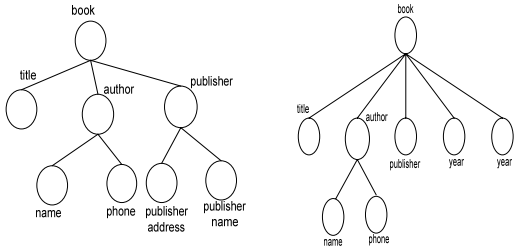

Figure 3: Example of XML document trees d1 and d2

Figure 3 show an example of XML document trees. We extract all distinct paths extracted from all XML trees with duplicate paths removed, then the dimensional feature vector matrix is defined. Feature extracted paths from three XML documents in Figure 3 is

p1=book/title
p2=book/author/name
p3=book/author/phone
p4=book/publisher/publisheraddress
p5=book/publisher
p6=book/year
p7=book/publisher/publishername

We consider a document collection to matrix $D_{mxn} = (d_1, d_2, \ldots, d_n)$, where n is the cardinality of the collection and m is the number of paths extracted from D. Each column is an m-dimensional vector relevant to document d_k , k $\in$[1...n] and every row corresponds to one *path*. The i_{th} row of d_k is the number of the corresponding *path* occurrences if *path* exists in d_k ; otherwise, the i_{th} row of vector is 0.

Paths	Real Path Sequence	d1	d2
p1	book/title	1	1
p2	book/author/name	1	1
p3	book/author/phone	1	1
p4	book/publisher/publisheraddress	1	0
p5	book/publisher	0	1
p6	book/year	0	2
p7	book/publisher/publishername	1	0

Each document id represented by an n-dimensional vector matrix. The feature vector matrix is defined if an XML document contains the common path is set to 1; otherwise it is set to 0. Based on the set of common path vectors collected, we compute the structure similarity between the XML documents.

4. Similarity Computation and Clustering Phase between XML documents

Clustering algorithms are used to find groups of "similar" data points among the input patterns. K-means algorithm is used to compute the similarities of XML documents. This is an effective algorithm to extract a given number of clusters of patterns from a training set. Once done, the cluster locations can be used to classify data into distinct classes. Partitioned clustering algorithms e.g., K-mean are more suitable for clustering large datasets. The similarity between two XML documents is defined as using the Euclidean distance (formula 1). For each pattern X, associate X with the cluster Y closest to X using the Euclidean distance:

$$\text{Dist}(X, Y) = \sqrt{\sum_{i=1}^{m} (X_i - Y_i)^2} \quad \text{——————} \quad (1)$$

We chose to use the k-means algorithm that is a commonly used partition clustering technique, where k is the number of desired clusters, either given as input, or determined in the loop. In the experiments, for simplicity and to allow easier comparison, we set k to be equal to the desired number of classes. The algorithm relies on an initial partition of the collection that is repeatedly readjusted, until a stable solution is found.

Algorithm: XMLClustering
Input: A feature vector matrix of XML documents
Output: clusters
1. Initialization:
– k points are chosen as initial centroids
– Calculate the similarity matrix using formula (1)
– Assign each point to the closest centroid
2. Iterate:
– Compute the centroid of each cluster
– Assign each point to the closest centroid
3. Stop condition:
– As soon as the centroids are stable

5.Conclusion

In this paper, we proposed path-based clustering that clusters the data with similar paths. Clustering is done by applying the partitioning technique to the path similarity matrix. The proposed clustering method is a flexible method which can process various types of XML documents efficiently. In order to cluster high-volume XML documents, we have proposed the step by step methods which are suitable for clustering the XPaths efficiently. We believe that the proposed methods efficient and valuable for various XML based applications.

References

[1]. S. Abiteboul, P. Buneman, D. Suciu, *Data on the Web.*, Morgan Kaufmann, 2000.

[2]. Lee JW, Lee K, Kim W (2001) *"Preparations for semantics-based XML mining"*. In: Proceedings of the 2001 IEEE international conference on data mining, San Jose, CA, December, pp 345–352

[3]. Miyahara T, Shoudai T, Uchida T, Takahashi K, Ueda H (2001) *."Discovery of frequent tree structured patterns in semi-structured Web documents"*. In: Proceedings of the Fifth Pacific-Asia conference on knowledge discovery and data mining (PAKDD), Hong Kong, China, April, pp 47–52

[4]. Miyahara T, Suzuki Y, Shoudai T, Uchida T, Takahashi K, Ueda H (2002). *Discovery of frequent tag tree patterns in semistructured Web Documents*. In: Proceedings of the sixth Pacific-Asia conference on knowledge discovery and data mining (PAKDD), Taipei, Taiwan, May, pp 341–355

[5]. Chang CH, Lui SC, Wu YC (2001) *Applying pattern mining to Web information extraction*. In: Proceedings of the fifth Pacific-Asia conference on knowledge discovery and data mining (PAKDD). Hong Kong, China, April, pp 4-16

[6] Ho-pong Leung, Fu-lai Chung, Chan, S.C.F., Luk, R. *XML Document Clustering Using Common Xpath*. Proc. of the International Workshop on Challenges in Web Information Retrieval and Integration. 2005, pp.91-96.

[7] Jin-sha Yuan1, Xin-ye Li1, Li-na Ma2. *"An Improved XML Document Clustering Using Path Feature"*. Fifth International Conference on Fuzzy Systems and Knowledge Discovery 978-0-7695-3305-6/08 $25.00 © 2008 IEEE DOI 10.1109/FSKD.2008.66

[8] Jianghui Liu, Jason T. L. Wang, Wynne Hsu, Katherine G. Herbert. *XML Clustering by Principal Component Analysis.* Proc. of the 16th IEEE International Conference on Tools with Artificial Intelligence. 2004,pp. 658 – 662.

[9] W3C's Document Object Model (DOM) home page [http://www.w3.org/DOM/]

[10] W3C's Extensible Markup Language (XML) home page [http://www.w3.org/XML/]

[11]http://www.sigmod.org/record/xml