Combining Structure and Content Similarities for XML Document Clustering

Tien Tran **Richi Nayak** **Peter Bruza**

Faculty of Information Technology
Queensland University of Technology
GPO Box 2434, Brisbane QLD 4001, Australia

Emails: {t4.tran, r.nayak, p.bruza}@qut.edu.au

Abstract

This paper proposes a clustering approach that explores both the content and the structure of XML documents for determining similarity among them. Assuming that the content and the structure of XML documents play different roles and importance depending on the use and purpose of a dataset, the content and structure information of the documents are handled using two different similarity measuring methods. The similarity values produced from these two methods are then combined with weightings to measure the overall document similarity. The effect of structure similarity and content similarity on the clustering solution is thoroughly analysed. The experiments prove that clustering of the text-centric XML documents based on the content-only information produces a better solution in a homogeneous environment, documents that derived from one structural definition; however, in a heterogeneous environment, documents that derived from two or more structural definitions, clustering of the text-centric XML documents produces a better result when the structure and the content similarities of the documents are combined with different strengths.

Keywords: XML, clustering, latent semantic kernel, vector space model.

1 Introduction

Over the past years, electronic documents in several formats, such as XML, HTML and XHTML, have been proposed to represent the textual content of the documents in a structural manner. For data representation and exchange, formatting in XML has emerged as a standard (Bray et al. 2004). With the continuous growth of the XML documents, data management issues, such as retrieval and storage of the large number of documents, have also arisen (Nayak et al. 2002). Clustering of these documents is one way of handling this issue. XML clustering is a task which can be applied to organize the massive amounts of XML documents into groups without the prior knowledge (Han & Kamber 2001); each group containing the documents that share similar characteristics. Clusters can be derived based on the content or based on the structural information of the XML documents. For example, clustering of XML documents based on the content is for dealing with XML datasets in a homogeneous environment, documents that use the same structure to represent different topics or themes e.g. IEEE transactions. This type of clustering application is useful in information retrieval and document engineering. On the other hand, clustering of XML documents based on the structure is for dealing with XML datasets in a heterogeneous environment, documents that use different structures to represent the same information, such as, a purchase order has different representations according to its originator where its information may represent differently. This type of clustering application is useful in database indexing, data-warehouse, data integration and document engineering.

A number of XML clustering approaches has been proposed in recent years; however, there is still very little work on the clustering of semi-structure documents that effectively combines the content and the structure information of the XML documents for clustering, especially for XML datasets in the homogeneous environment. Assuming that the content and the structure of the XML documents play different roles and importance according to the use and purpose of an application, we propose an approach to cluster text-centric XML datasets, datasets in which the content is the most important feature in determining the document similarity, by calculating each of the content similarity and the structure similarity of a document separately, and then combining them with appropriate strengths, defined by the user, for document similarity. The structure similarity is determined by the commonality and co-occurrence of paths between document structures. A latent semantic kernel (Cristianini et al. 2002) is used to determine the semantic association within document contents.

The empirical analysis reveals that clustering of the text-centric XML datasets based on the content-only information produces a better solution in a homogeneous environment; however, in a heterogeneous environment, clustering of the text-centric XML datasets produces a better result when the structure and the content similarities of the documents are combined with different strengths. Our contributions are as follows: (1) Using Latent Semantic Kernel (LSK) for measuring the semantic associations of the textual content of XML documents, and; (2) Exploiting the semantic of the document contents and the commonality of the document structure for XML clustering.

1.1 Related Work

There has been a myriad of clustering approaches proposed in recent years. Some of these approaches (Kurgan et al. 2002, Shen & Wang 2003) discard the structural information of the XML documents and the similarity learning is based on the content-only information. However, a good clustering process should not discard the use of the structure since XML is popularly known for its representation and storing of the structural content that can be easily processed by systems such as the databases.

Clustering approaches are varied according to the representation of the XML data such as tree-based, path-based, graph-based, etc. The method of calculating document similarity varies accordingly. The similarity matrix generated by these approaches usually becomes an input to a traditional clustering method such as the hierarchical agglomerative algorithm or the k-means algorithm (Han & Kamber 2001). Several approaches (Nierman & Jagadish 2002, Dalamagas et al. 2004) have been proposed to represent the XML documents as tree-based and use the tree edit distance to measure the similarity between the documents using the document structure. Lian et al. approach (Lian et al. 2004) represents the XML document as graph-based and measures the common set of nodes and edges appearing between the documents. To retain the structure information from the XML documents, some approaches (Jeong & Keun 2004, Leung et al. 2005, Jeong & Keun 2005) use the sequential pattern mining to extract the frequent paths from XML documents and then use them for clustering. XClust (Lee et al. 2002) introduces a complex computational technique to map the element similarity between the schemas by considering the semantics, immediate descendent and leaf-context information. Its purpose is to be used as the preprocessing stage for applications such as data integration.

The approaches which previously discussed consider only the structure information. Content mining has been well explored in area such as information retrieval where the content of the document can be represented as a vector space model (Salton & McGill 1983). Methods such as tf*idf weight (Salton & McGill 1983), feature reduction methods such as principal component analysis (Liu et al. 2004) and latent semantic analysis (Landauer et al. 1998) have been widely used to measure the similarity between a document to a query (Kim et al. 2005, Yang et al. 2005). The latent semantic analysis (Landauer et al. 1998) constructs a semantic space wherein terms and documents that are closely associated are placed near one another. This space reflects major associative patterns in the data and ignores less important patterns.

Recognizing the importance of the content with the structure of the XML documents, a number of approaches (Shen & Wang 2003, Kc et al. 2006, Yang et al. 2005) have been proposed to incorporate the content and the structure of the XML documents for clustering. Shen and Wang (2003) approach breaks the XML documents into a number of macro-path sequences where each macro-path contains the properties of an element such as its name, attributes, data types and textual content. A matrix similarity of the XML documents is then generated based on the macro-path similarity technique. The clustering of XML documents is performed based on the similarity matrix with the support of approximate tree inclusion and isomorphic tree similarity. Kc et al. (2006) uses the self-organizing maps (Kohonen 1990) for learning the structure of the XML documents. However when it attempts to use the self-organizing maps for including both the content and the structure of XML documents, it performs poorer than the structure-only clustering solutions on the INEX datasets. This shows that for certain datasets, using the structure and the content information together in the clustering process degrades the performance of the clustering solutions (Denoyer et al. 2006). Taking this into consideration when dealing with different datasets, our approach measures the content and the structure similarities separately, and then combines them with different strengths. This gives a relative importance to the structure and to the content according to the type of the datasets.

2 Overview of the Proposed Clustering Approach

Figure 1 illustrates the overview of the proposed clustering approach. The XML dataset is pre-processed to extract the content and the structural information. The content of the XML documents, here, refers to the textual data, and the structure is referring to the elements (or tags) which are used to structure the content. The content of a document is represented by a collection of unique terms after stop-word removal and stemming (Porter 1980). Stop-word is term that considered not to be important such as "is", "or", "a", etc. Only the keywords of the content are used for the content similarity measure. Whereas, the structural information of the XML documents is represented as paths, containing element names in hierarchical order, which are used for structure similarity measure.

Both the content and the structure information are represented using the vector space model (Salton & McGill 1983). As the proposed approach addresses the problem of combining the structure and the content similarities for text-centric dataset, sophisticated structure measure is not required since the text-centric dataset is conformed to the same structural definition and various instances of the dataset do not vary much in their structure representations. The content is measured separately from the structure using a different method. The document similarity is measured by combining the structure similarity value and the content similarity value. The output of the document matching is a pair-wise document similarity matrix which contains the document similarity between each pair of XML documents in the dataset. This matrix is then used to cluster the dataset. The next section describes how the document similarity is measured in more detail.

3 Document Similarity Measure

The document similarity between two XML documents, d_x and d_y, is defined as:

$$\begin{aligned} docSim(d_x, d_y) &= (contSim(d_x, d_y) \times \lambda) \\ &\quad + (structSim(d_x, d_y) \times (1 - \lambda)). \end{aligned} \tag{1}$$

The document similarity is a combination of the content similarity value and the structure similarity value. The λ, ranging from 0 to 1, is defined by the user to adjust the importance of the content similarity (*contSim*) and structure similarity (*structSim*). A pair-wise document similarity matrix is generated by computing the similarity between each pair of XML documents in the dataset using the document similarity measure as defined in equation 1. A clustering method such as k-means or hierarchical agglomerative can be applied to find clusters in the pair-wise document similarity matrix.

3.1 Structure Similarity Measure

The structure of an XML document relates to how the content in the XML document is structured. Information such as element names, data types, constraints, parents, ancestors, children, etc. can be used to discover the structural similarity between XML documents. To simplify the structure matching

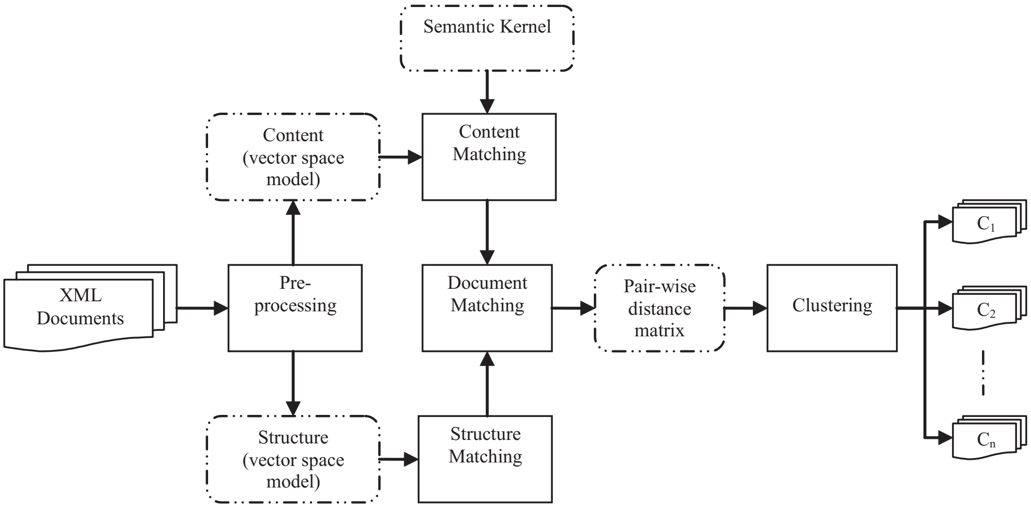

Figure 1: Overview of the proposed clustering approach.

d_1

$\langle R \rangle$
$\quad\langle E_1 \rangle\, t_1, t_2, t_3$
$\quad\langle E_2 \rangle\, t_4, t_3, t_6$
$\quad\langle E_3 \rangle\, t_5, t_4, t_7$
$\qquad\langle E_{3.1} \rangle\, t_5, t_2, t_1$
$\qquad\langle E_{3.2} \rangle\, t_7, t_9$

d_2

$\langle R \rangle$
$\quad\langle E_1 \rangle\, t_1, t_4$
$\quad\langle E_2 \rangle\, t_3, t_3$
$\quad\langle E_3 \rangle\, t_4, t_7$
$\qquad\langle E_{3.1} \rangle\, t_2, t_9$
$\qquad\langle E_{3.1} \rangle\, t_2, t_7, t_8$

d_3

$\langle R \rangle$
$\quad\langle E_1 \rangle\, t_1, t_2$
$\quad\langle E_2 \rangle\, t_3, t_3$
$\quad\langle E_3 \rangle\, t_5, t_4, t_7$
$\qquad\langle E_{3.1} \rangle\, t_5, t_2, t_1$
$\qquad\langle E_{3.2} \rangle\, t_7, t_9$

d_4

$\langle R \rangle$
$\quad\langle E_1 \rangle\, t_1, t_4$
$\quad\langle E_3 \rangle\, t_4, t_7$
$\quad\langle E_3 \rangle\, t_4, t_8$
$\quad\langle E_1 \rangle\, t_1, t_4$

Figure 2: Dataset D containing 4 XML documents.

process, only the element names, the most important property of the elements, are used for structure matching. The structure of an XML document is represented as a tree-based in which it is broken down into a collection of distinct paths. These paths are used to measure the structural distance between XML documents. Given a dataset of XML documents $\{d_1, d_2, ..., d_n\}$, denoted by D, a set of distinct paths $\{p_1, p_2, ..., p_f\}$, denoted by P, are extracted from D.

Definition 1 (Path). *A path, p_i, contains element names from the root element to the leaf element. The leaf element is an element that contains the textual content.*

Definition 2 (Structure Modeling). *The structure of a document, d_i, is modelled as a vector $\{p_{i,1}, p_{i,2}, ..., p_{i,f}\}$, where each element of the vector represents the frequency of a path in P that appears in the document.*

Definition 3 (Structure Matching). *Given two documents, d_x and d_y, and their corresponding vectors, $\{p_{x,1}, p_{x,2}, ..., p_{x,f}\}$ and $\{p_{y,1}, p_{y,2}, ..., p_{y,f}\}$ respectively. The distance between the two documents is computed using the Euclidean distance.*

Table 1: A matrix Y representing the structure information of the dataset.

path/doc	d_1	d_2	d_3	d_4
R/E_1	1	1	1	2
R/E_2	1	1	1	0
$R/E_3/E_{3.1}$	1	2	1	0
$R/E_3/E_{3.2}$	1	0	1	0
R/E_3	1	1	1	2

$$structSim(d_x, d_y) = \sqrt{\sum_{i=1}^{f} (p_{x,i} - p_{y,i})^2}. \quad (2)$$

The structSim is normalised between 0 and 1.

Example. Let us assume a collection, D, containing 4 XML documents $\{d_1, d_2, d_3, d_4\}$, as shown in figure 2; element names in the documents are shown as embraced within brackets, $\langle R \rangle$ is the root element and $\langle E_i \rangle$ is the internal element or leaf element. The content of a document is denoted by T. The structure of a document is extracted and represented as a vector. The structures of all the documents in the dataset can be put together as a path-document matrix, $Y_{f \times n}$, where f is the number of distinct paths in P and n is the number of documents in D, as shown in table 1. Each cell in matrix Y is the frequency of a distinct path appearing in a document.

3.2 Content Similarity Measure

The semantic association among the document contents is measured using a latent semantic kernel (Cristianini et al. 2002). Consider the example documents in figure 2, a set of distinct terms $\{t_1, t_2, ..., t_m\}$, denoted by T, is extracted from the dataset D. A term-document matrix, $X_{m \times n}$, where m is the number of terms in T and n is the number of documents in dataset D, is constructed as shown in table 2.

The singular value decomposition (SVD) decomposes the term-document matrix, $X_{m \times n}$, into three matrices (equation 3), where U and V have orthonormal columns values of left and right singular vectors

Table 2: A matrix X representing the content information of the dataset.

term/doc	d_1	d_2	d_3	d_4
t_1	2	1	2	2
t_2	2	2	2	0
t_3	2	2	2	0
t_4	2	2	1	4
t_5	2	0	2	0
t_6	1	0	0	0
t_7	2	2	2	1
t_8	0	1	0	1
t_9	1	1	1	0

respectively and S is a diagonal matrix of singular values ordered in decreasing magnitude.

$$X = USV^T. \tag{3}$$

SVD can optimally approximate matrix X with a smaller sample of matrices by selecting k largest singular values and setting the rest of the values to zero. Matrix U_k of size $m \times k$ and matrix V_k of size $n \times k$ may be redefined along with $k \times k$ singular value matrix S_k (equation 4). This can approximate the matrix X in a $k-$dimensional document space.

$$\hat{X}_{m \times n} = U_k S_k V_k^T. \tag{4}$$

Matrix $\hat{X}$ is known to be the matrix of rank k which is closest in the least squares sense to X. Matrix U_k becomes the latent semantic kernel that can be used to measure the semantic associations between two document contents.

Definition 4 (Terms). *A term, t_i, is a keyword that appears in the textual content of the elements in the XML document after stop-word removal and stemming (Porter 1980).*

Definition 5 (Content Modeling). *The content of a document, d_i, is modelled as a vector $\{t_{i,1}, t_{i,2}, ..., t_{i,m}\}$, where each element of the vector represents the frequency of a term in T that appears in the document.*

Definition 6 (Content Matching). *Given two vectors, d_x and d_y, the semantic similarity of the documents content is measured as:*

$$contSim(d_x, d_y) = \frac{d_x^T P P^T d_y}{|P^T d_x||P^T d_y|}. \tag{5}$$

where matrix P is matrix U_k, and P is used as a mapping function to transform the two documents, d_x and d_y, into concept space to determine the semantic association of document contents.

4 Empirical Evaluation

4.1 Dataset

The IEEE and Wikipedia datasets, available from the INEX 2006 Document Mining Challenge (Denoyer et al. 2006), are used to evaluate the proposed clustering approach. The clusters are labelled according to the content theme or topic which makes the content similarity measure more important than the structure similarity measure.

The IEEE dataset is derived from the same structural definition therefore all documents contain the same set of element names. Likewise, the Wikipedia dataset is not conformed to any particular structural definition but documents also contain the same set of element names amongst the dataset. As a result no semantic learning is necessary on the element names. Table 3 shows the detail of the datasets. A subset of the Wikipedia dataset is used in the experiments. Wikipedia and IEEE datasets are homogeneous dataset, meaning, the documents in the dataset are conformed to only one structural definition; whereas, the heterogeneous dataset is a mixture of both the Wikipedia and IEEE documents where they are conformed to two different structural definitions.

Table 3: Datasets.

Datasets	**#Documents**	**#True Categories**
Wikipedia	3000	60
IEEE	6054	18
Heterogeneous dataset	3900	78

4.2 Evaluation Methods

Two evaluation methods are used to measure the accuracy of the clustering solution; micro-average F1 and macro-average F1. Given a particular category, consider the number of positive documents which are clustered as positive (PP), the number of false negative documents which are clustered as positive (NP), and the number of false positive documents which are clustered as negative (PN), precision and recall are defined as follows:

$$Precision(P) = \frac{PP}{PP + NP}. \tag{6}$$

$$Recall(R) = \frac{PP}{PP + PN}. \tag{7}$$

The F1 measure for this particular category can be defined as:

$$F1 = \frac{2PR}{P + R}. \tag{8}$$

Micro-average F1 is calculated by summing up the PP, the NP, and the PN values from all the categories; F1 value is then calculated based on these values. Macro-average F1, on the other hand, is derived from averaging the F1 values over all the categories. The best clustering solution for an input data set is the one where micro- and macro-average F1 measures are close to 1. The Micro-average F1 value is easier to achieve than the macro-average F1 value.

4.3 Experimental Design

In the experiments, a subset, ranging from 1000 to 1300 documents, of each dataset is used for the construction of the latent semantic kernels. Only a subset is used because applying the singular vector decomposition method (SVD) on a large term-document matrix is expensive in terms of computational time and memory requirements, and sometimes infeasible. During the selection of the subset, it is ensured that the kernel is build on a large number of terms that appear in the dataset. Documents that contain large number of frequent terms are selected for the kernel construction. In the experiments, the clustering solution is analysed using different k values for selecting the kernels. Results, as shown in table 4 on the heterogeneous dataset, show that the k dimension of 200 and 400 is good to infer semantic association among the dataset contents. These values have been used

Table 4: The effect of k values on the clustering solution for the heterogeneous dataset.

k	Micro-average F1	Macro-average F1
100	0.299	0.240
200	0.346	0.290
400	0.308	0.247
600	0.283	0.222
800	0.280	0.223

for the evaluation and approaches comparison in this paper. Three different kernels are created for three different datasets as shown in table 3. A hierarchical clustering method (Karypis 2007) is used to cluster the pair-wise document similarity matrix produced from our clustering approach. The hierarchical clustering method performs by first dividing the dataset, in this case the pair-wise document similarity matrix, into two groups, and then one of these two groups is chosen to be bisected further. The process is repeated until the number of bisections in the process equals to the number of clusters defined by the user.

Experiments are conducted to evaluate the effect of the structure similarity and the content similarity on the clustering solutions. Results of the proposed clustering approach on the IEEE dataset are compared with two other approaches (Doucet & Lehtonen 2006, Kc et al. 2006). The first one is the Doucet et al. (2006) approach which uses the vector space model for representing the XML document features, and then k-means to cluster the documents. The other one is the Kc et al. (2006) approach which uses the self-organization maps to combine the structure and content information for document clustering.

4.4 Results and Analysis

The effect of the weighting parameter λ. The structure and content similarities are adjusted with the weighting parameter λ. Figures 3, 4, and 5 show the effect of the weighting importance of the content similarity and structure similarity on the Wikipedia, IEEE and the heterogeneous datasets, respectively. Each figure shows the performance of micro-average and macro-average F1 values with various combinations of weighting parameters that is monitored by λ in equation 1. The graphs in the figures start with λ set to 0, where the importance of the content similarity is set to 0 and the importance of the structural similarity is set to a 1. The F1 values are then recorded each time with an increment of 0.1 in λ, decreasing the structural weight parameters by 0.1 and increasing the content weight parameters by 0.1. In general, the F1 measures become better with each increment in the content weight parameter. When the content weight is set to a higher value, the results are better in comparison to the results when the content weight is set to a lower value. This shows that the content information on these datasets plays an important role on the performance of the clustering solution. This is the expected results as documents are categorized according to the content that they share. Based on the results in figures 3 and 4, it can be ascertained that the structure of the data does not play much importance in the clustering of the datasets in homogeneous environment. However, in heterogeneous environment, results, as shown in figure 5, show that when the structure weight is assigned with a 0.1 or 0.2, the results are slightly better than the result with the content-only information. This emphasizes that by combining the structure and content measures with different strengths produces a better clustering solution for text-centric XML documents from heterogeneous environment.

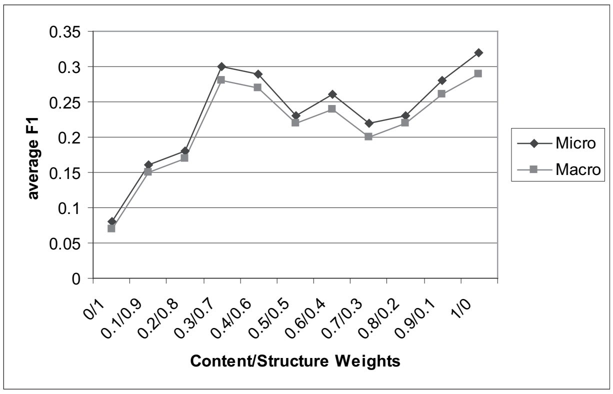

Figure 3: The effect of the structure and content similarities on the clustering solution of the Wikipedia dataset.

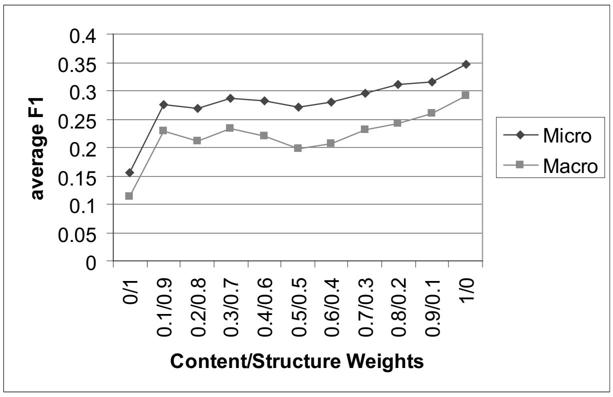

Figure 4: The effect of the structure and content similarities on the clustering solution of the IEEE dataset.

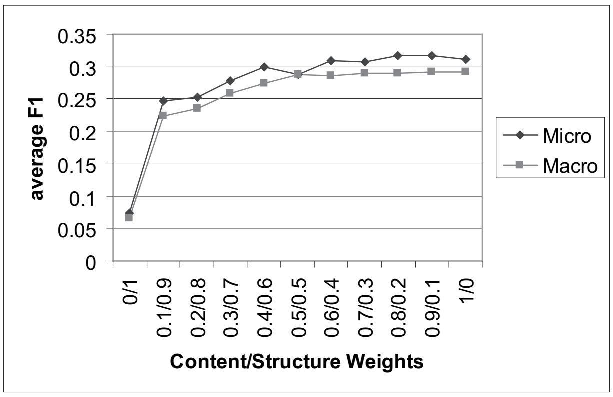

Figure 5: The effect of the structure and content similarities on the clustering solution of the heterogeneous dataset.

In this paper, we employ the structure similarity based on the frequency of the paths represented in the vector space model. We have also employed other representations and measures to exploit the structure information of XML documents in clustering as shown in table 5. The path vector space model approach is the one which has been used in this paper. The path-based approach (Nayak & Tran 2007) measures the structure similarity between documents using the path representation. The paths between documents are measured by considering

Table 5: The structure-only clustering solutions for Wikipedia dataset.

Approach	Micro-average F1	Macro-average F1
Path Vector Space Model	0.08	0.07
Path-based (Nayak & Tran 2007)	0.12	0.04
Tree-based (Kutty et al. 2007)	0.10	0.02

the hierarchical order of the elements in the paths. Whereas, the tree-based approach (Kutty et al. 2007) is to measure the structure similarity based on tree representation where the sibling information of the elements is also exploited for document similarity. All three approaches give very close results, as given in table 5, showing that the structure similarity on the Wikipedia dataset does not improve much with different representations. This shows that the proposed way of determining the structural similarity, in this paper, is sufficient enough for the clustering of the text-centric datasets. The path vector space model is chosen to be used in this proposed approach because it is faster to compute than the other two representation approaches.

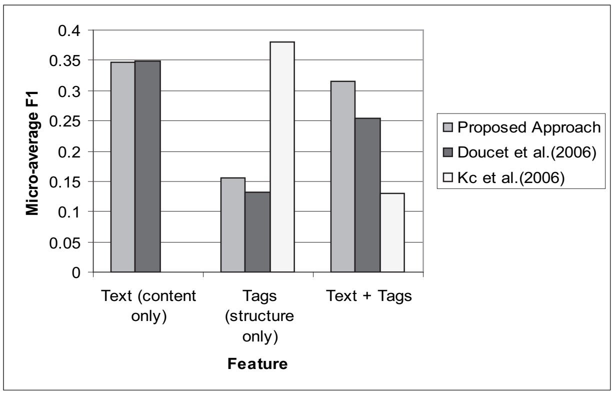

Figure 6: The micro-average F1 of the proposed approach, Doucet et al.(2006), and Kc et al.(2006)

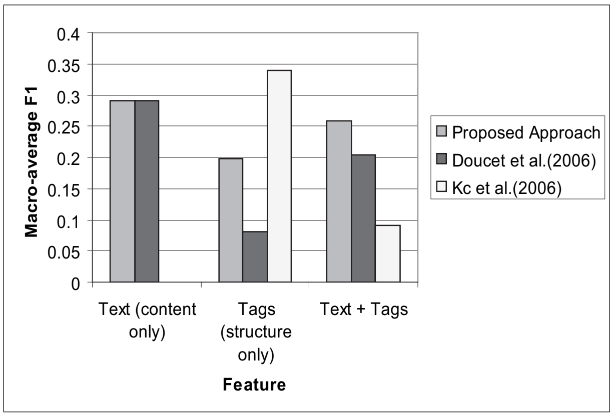

Figure 7: The macro-average F1 of the proposed approach, Doucet et al.(2006), and Kc et al.(2006)

Approaches comparison on the IEEE dataset. Figures 6 and 7 show the comparison of the proposed approach with the other approaches on micro-average F1 and macro-average F1 results, respectively. For the content-only information (content similarity), our approach and Doucet et al. (2006) produce similar results. Even though, our kernel is built on a subset of document features in the dataset however the performance of the kernel is not worse than the vector space models based on the whole dataset features. Kc et al. (2006) uses the self-organization maps that outperforms both the vector space model based methods, Doucet et al. method (2006) and our approach, for using the structure-only information (structure similarity). However, when the structure and the content information are used, Kc et al.(2006) method performs the worse.

In summary, the self-organization maps method (Kc et al. 2006) is much better than the vector space model approach employed in Doucet et al. (2006) and co-occurrence counting of paths used in our approach for learning the structure of XML documents. On the other hand, the content of XML documents are better represented and grouped if it is represented as a vector space model or using the latent semantic kernel in our approach. When both the structure and content information of the XML documents are used for clustering, The proposed clustering approach outperforms the other two approaches because it can adjust the weighting importance on the content similarity and the structure similarity depending on the nature of the dataset.

5 Conclusions

This paper introduces a clustering approach based on two separate measures to explore the structure and content similarities of XML documents. In this paper we propose to adapt the latent semantic kernel to learn the semantic associations of XML document contents for content similarity. The result of the content similarity are combined with the structure similarity of the documents by assigning the two similarity measures with different weightings. This paper produces a systematic study of the effect of the structure and content similarities of the XML documents in the clustering process that has not been done previously in our knowledge. The method is thoroughly analysed and compared with other methods.

Empirical analysis ascertains the following information. In heterogeneous environment, the inclusion of structural similarity with the content similarity can produce a better result. The performance of the proposed approach is better when the dataset is in a heterogeneous environment rather than in a homogeneous environment. This is due to combining both the structure and content similarities using different measures and different weights. This shows the applicability of the proposed approach as this is usually the case in real practice. While grouping the data sets based on theme categories such as the Wikipedia and IEEE datasets, the clustering performance degrades when the structure of the documents is included in the clustering process. The content of the Wikipedia and IEEE datasets plays a major role in determining the clustering solutions, whereas the structure plays a small role.

The structure mining employed by this paper is a trivial method of measuring the structure of XML documents in a heterogeneous environment since hierarchical structural information of the document structure is not fully captured. However, the previous work has shown that the order of nodes is not important in clustering of text-centric XML datasets. Also the focus of this paper is to include the content and structure similarities in the clustering process. The experiments ascertained that the structure similarity and the content similarity can contribute to the overall clustering solution when documents belong to different structural definitions.

References

Bray, T., Paoli, J., Sperberg-McQueen, C., Maler, E. & Yergeau, F. (2004), 'Extensible markup language (xml) 1.0 (third edition) w3c recommendation'.
URL: *http://www.w3.org/TR/2004/REC-XML-20040204/*

Cristianini, N., Shawe-Taylor, J. & Lodhi, H. (2002), 'Latent semantic kernels', *Journal of Intelligent Information Systems (JJIS)* **18**(2).

Dalamagas, T., Cheng, T., Winkel, K. & Sellis, T. K. (2004), Clustering xml documents by structure, *in* 'SETN'.

Denoyer, L., Gallinari, P. & Vercoustre, A.-M. (2006), Report on the xml mining track at inex 2005 and inex 2006, *in* 'INEX 2006', Dagstuhl Castle, Germany, pp. 432–443.

Doucet, A. & Lehtonen, M. (2006), Unsupervised classification of text-centric xml document collections, *in* '5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX', pp. 497–509.

Han, J. & Kamber, M. (2001), *Data Mining: Concepts and Techiques.*, San Diego, USA: Morgan Kaufmann.

Jeong, H. H. & Keun, H. r. (2004), A new xml clustering for structural retrieval, *in* '23rd International Conference on Conceptual Modeling', Shanghai, China.

Jeong, H. H. & Keun, H. R. (2005), Clustering and retrieval of xml documents by structure, *in* 'ICCSA', Singapore.

Karypis, G. (2007), 'Cluto - software for clustering high-dimensional datasets — karypis lab'.
URL: *http://glaros.dtc.umn.edu/gkhome/views/cluto*

Kc, M., Hagenbuchner, M., Tsoi, A., Scarselli, F., Sperduti, A. & Gori, M. (2006), Xml document mining using contextual self-organizing maps for structures, *in* '5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX', Dagstuhl Castle, Germany, pp. 510–509.

Kim, Y.-S., Cho, W.-J., Lee, J.-Y., Oh & Yu-Jin (2005), An intelligent grading system using heterogeneous linguistic resources, *in* 'IDEAL 2005', p. 102108.

Kohonen, T. (1990), 'Self-organisation and associative memory', *Springer, 3rd edition* .

Kurgan, L., Swiercz, W. & Cios, K. J. (2002), Semantic mapping of xml tags using inductive machine learning, *in* '11th International Conference on Information and Knowledge Management', Virginia, USA.

Kutty, S., Tran, T., Nayak, R. & Li, Y. (2007), Clustering xml documents using closed frequency subtrees - a structure-only based approach, *in* '6th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2007', Dagstuhl Castle, Germany.

Landauer, T. K., Foltz, P. W. & Laham, D. (1998), 'An introduction to latent semantic analysis.', *Discourse Processes* (25), 259–284.

Lee, L. M., Yang, L. H., Hsu, W. & Yang, X. (2002), Xclust: Clustering xml schemas for effective integration, *in* '11th ACM International Conference on Information and Knowledge Management (CIKM'02)', Virginia.

Leung, H.-p., Chung, F.-l., Chan, S. & Luk, R. (2005), Xml document clustering using common xpath, *in* 'International Workshop on Challenges in Web Information Retrieval and Integration (WIRI '05)', pp. 91–96. TY - CONF.

Lian, W., Cheung, D. W., Maoulis, N. & Yiu, S.-M. (2004), 'An efficient and scalable algorithm for clustering xml documents by structure', *IEEE TKDE* **16**(1), 82–96.

Liu, J., Wang, J., Hsu, W. & Herbert, K. (2004), 'Xml clustering by principal component analysis', *Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on* pp. 658–662.

Nayak, R. & Tran, T. (2007), 'A progressive clustering algorithm to group the xml data by structural and semantic similarity', *IJPRAI* **21**(3), 1–23.

Nayak, R., Witt, R. & Tonev, A. (2002), Data mining and xml documents, *in* 'The 2002 International Workshop on the Web and Database (WebDB 2002)'.

Nierman, A. & Jagadish, H. V. (2002), Evaluating structural similarity in xml documents, *in* '5th International Conference on Computational Science (ICCS'05)', Wisconsin, USA.

Porter, M. (1980), 'An algorithm for suffix stripping', *Program* **14**(3), 130–137.

Salton, G. & McGill, M. J. (1983), 'Introduction to modern information retrieval', *McGraw-Hill* .

Shen, Y. & Wang, B. (2003), Clustering schemaless xml document, *in* '11th international conference on Cooperative Information System'.

Yang, J., Cheung, W. & Chen, X. (2005), Learning the kernel matrix for xml document clustering, *in* 'e-Technology, e-Commerce and e-Service'.