I J M I

International Journal of Machine Intelligence
ISSN: 0975–2927 & E-ISSN: 0975–9166, Volume 3, Issue 4, 2011, pp-376-380
Available online at http://www.bioinfo.in/contents.php?id=31

CONTENT AND STRUCTURE BASED CLASSIFICATION OF XML DOCUMENTS

SHASHIREKHA H.L.[1*], VANISHREE K.S.[2], SUMANGALA N.[3]
[1]Dept. of Computer Science, Mangalore University, Mangalore, Karnataka, India,
[2]Dept. of Computer Science, Govt First Grade College, Sikaripura, Karnataka, India.
[3]Dept. of MCA, St Joseph Engineering College, Mangalore, Karnataka India.
*Corresponding Author: Email- hlsrekha@gmail.com

Received: November 06, 2011; Accepted: December 09, 2011

Abstract - The ever increasing amount of XML documents available on the World Wide Web demands automated tools and techniques that would make the search and retrieval of XML documents more effective and efficient. Classification of XML documents is one of the significant tasks which are being explored by many researchers in this direction. Due to the presence of inherent structure in the XML documents, conventional text classification methods cannot be used to classify XML documents directly. Hence, there is a need for the development of tools and techniques that automatically classifies XML documents. In this work, we have developed an algorithm based on 'k' nearest neighbors to classify XML documents by considering both the content and structure. The developed algorithm is tested on a subset of MEDLINE dataset for different values of 'k' and varying size of training set and the results are tabulated.

Keywords - XML documents, text classification, 'k' nearest neighbors, cosine similarity, tree structure

Introduction

XML documents are gaining significant importance as it is considered as a standard for data representation and management on the Web. The ever increasing amount of XML documents on the WWW demands the automated tools and techniques that would make the search and retrieval of XML documents more effective and efficient [6, 13]. In contrast to traditional information retrieval systems that deal with flat documents, XML retrieval systems must also take the logical structure of documents into account as every XML document includes both logical and physical structure [1, 2, 8]. The logical structure is like a template that entitles the elements to be included in a document and in the order in which they have to be included. Further, it refers to the organization of different parts of a XML document and indicates how a document is built, as opposed to what a document contains. The physical structure contains the raw data which is composed of all the content used in that document. As a result, in addition to the raw text (pure content), the structural information contained in XML documents serves as a valuable information source for document representation.

XML documents which belong to the class of semi-structured documents [2] have some implicit structure that is generally followed, but not enough of a regular structure to "qualify" for the kinds of management and automation usually applied to structured data. While there is no strict formatting rule; there is enough regularity that some interesting information can be extracted. There are two different views of XML documents: the document-centric view and data centric view [8]. While document-centric XML documents has a much more irregular structure and is often encountered as the means of document markup, data-centric XML documents are characterized by a fairly regular structure and occur as a standard format for data exchange and representation of semi structured data.

With the increase in XML documents, researchers are now focusing on applying the typical text mining tasks such as text classification, text clustering, concept/entity extraction, document summarization and other related tasks on XML corpus which otherwise are applied on flat documents [1, 7, 9, 14, 16, 17-18]. In this research work, our objective is to throw light upon the classification of XML documents. As XML documents are basically text documents containing the content and structure information, they can be classified based on i) content only [10, 13] ii) structure only [14] and iii) a combination of both structure and content [9, 17-18]. A natural tendency for content based XML document classification would be to use conventional text classification approaches where each XML document could be treated as Bag-Of-Words (BOW) [7]. This approach is not an efficient one as it totally ignores the structural component of XML documents; thereby defeating the whole purpose of XML documents itself. The second method which is based on structure generally model XML documents as labeled trees, where the interior nodes represent the XML tags and the leaf nodes represent the content. Similarities between the XML documents are then found by computing the edit distance between the labeled trees. Besides structure, contents also have a major role to play in XML documents which is ignored in this approach. As the XML documents are made up of both structure and content it is quite natural to give equal importance for structure and content (method 3) rather than considering only either of them. Hence, in this work, we have developed a XML document classifier considering a combination of both structure and content based on k-nearest neighbors. The rest of the content is organized as follows: Section 2 highlights the research work that is carried out in the related field. Our methodology is discussed in section 3 and experimentation and results are discussed in section 4. Finally the paper concludes in section 5.

Related work

With the increase in XML documents, researchers are exploring many methods for their classification. While some approaches extend the traditional information retrieval methods [7, 10], some other are based on tree edit distance method where XML documents are represented as labeled trees and the distance between the documents is defined as the edit distance between the labeled trees [14]. Some of the research works on classification on XML documents are briefly summarized. The approach proposed by Abdelhamid [1] discovers the structural and content characteristics shared by XML documents of the same class. This approach based on k-nearest neighborhood algorithm relies on edit distance measures which consider both the content and structure of XML trees with structure bearing more weight than content. A bottom up approach for XML classification introduced by Junwei Wu [9] gives more weights to content of the XML documents. It is a similarity based method which begins with the content represented as leaf nodes and then the structural information is embedded. A methodology for indexing and retrieval from XML document is proposed in [12] where the structure information is represented using attributes so that the structure of the document can be expanded further. During indexing or retrieval, the attributes are converted into elements for which the existing systems can be used and indexing is made hierarchically. Mohammed J. Zaki and etal., [14] have proposed a rule based classifier XRules that uses Bayesian rule for decision making where XML documents are modeled as ordered rooted labeled trees. During classification, the rules relevant to a test document are identified and the statistics of the matched rules are used to predict the category of that document.

Saptarshi Ghosh and Pabitra Mitra [16] have proposed a combination of structure and content information using composite support vector machine (SVM) kernels for supervised XML classification. They have used kernels which individually measure the structure and content similarities. Both Boolean and Cosine similarity models are used to measure the structural similarity between XML documents. Jianwu Yang and Fudong Zhang [7] described a classification approach for XML documents based on Structured Link Vector Model (SLVM) which is an extension of conventional Vector Space Model (VSM) and Support Vector Machine. SLVM incorporates document structures which are represented as term by element matrices, referencing links that are extracted based on IDREF attributes as well as element similarity (represented as an element similarity matrix). This approach takes into account term semantics, element similarity, as well as elements' relative importance for a given set of documents.

A system that classifies XML documents based on their content and structure is presented by Swathy Giri and etal [17]. Here, XML documents are classified by a weighted combination of fieldwise content similarities. Here the algorithm automatically determines the field weights for an XML document based on features extracted from the field contents. The characteristics used for identifying useful fields for classification are the fields that have a large number of tokens and the fields with higher variability in their content across documents. In their work, Hui-I Hsiao [6] proposes a categorization technology which supports basic folder like operations and provides a set of advanced functionalities such as multipath navigation and traversal across multiple document collections for organizing and categorizing XML documents. This technology takes full advantage of the rich information and semantics embedded in XML documents. A report on the XML mining Track at INEX 2005 and 2006 [11] discusses the nine different models used for clustering and classification of XML documents using structure only and structure and content. This report highlights the fact that the structure only task is quite easy among other methods and simple models work very well with this task. In the formal model based on Bayesian classification developed by P.F. Marteau and etal., [17], a structural context of occurrence for unstructured data is defined and a recursive formulation is derived in which parameters are used to weigh the contribution of structural element relatively to the others. The tree structure of the XML document is approximated as a set of nodes from which the path to the root is attached.

A supervised machine learning system introduced by Georges Gardarin [5] classifies documents into predefined categories and tags them accordingly before storing them to database. A new hybrid algorithm CKNN is proposed by combining centroid based and knn algorithms. Zhang Na and etal., [18] presents a method called NM-Similarity computing similarity measure which is used by kNN for classification. The structure similarity between XML documents is computed by using Euclidean distance and the content similarity is computed by using cosine measure. They claim that when XML documents are similar in structure but different in content, MN-similarity provides a significant improvement in classification accuracy. Joe Tekli and etal [8] presents the background, current trends and future directions related to XML similarity. Some possible future research directions, covering XML structural and semantic similarity as well as the exploitation of XML grammars in developing improved XML comparison methods are discussed in their paper.

Methodology

In our approach, we assume that XML documents belonging to a particular category will adhere to a single DTD and there may be any number of categories /classes. However, we do not consider the DTDs for classifying new unlabeled XML documents. While content plays a major role in XML documents and without content XML documents are incomplete, structure component enforces the inherent hierarchical structure making them different from the unstructured documents which are represented as flat files. In view of these issues, we consider both the content and the structure components of the XML documents in designing a classifier. Here, each XML document is represented as a rooted ordered labeled tree, where the ordering is from left to right. The interior nodes of this tree represent the tags and leaf nodes represent the contents. Content component of an XML document may represent a single word or even a paragraph. The preprocessing methods used in the traditional Information Retrieval [4] are applied for the content part of the XML documents to obtain keywords. These methods include removal of stop words, punctuation, numeric information and words of length less than or equal to two. Further stemming is applied to reduce the words to their morphological root and the frequency count of each distinct word is obtained. Each of these keywords is then prefixed with a keypath to define a term, where keypath is the path that includes all the tags that appear from the root node to the immediate parent of each leaf node. This process is applied to all the documents in the training set and the documents are represented as term vectors where the terms are weighed using 'tf x idf'. These term vectors together represent a term x document matrix of the training documents. Example of a XML document, the corresponding tree structure and the terms are shown in Fig (1), (2) and (3) respectively.

```
<Academy>
   <Department>
        <Faculty>
           <Professor>John  Cramer </Professor>
      </Faculty>
   </Department>
  <Student>Harry </Student>
</Academy>
```

Fig. 1. Sample XML document

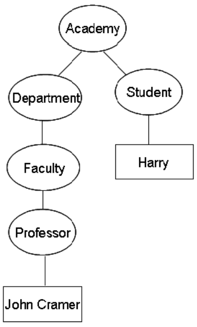

Fig. 2. Document representation using both structure and content

1. Academy Department Faculty Professor John
2. Academy Department Faculty Professor Cramer
3. Academy Student Harry

Fig. 3. Terms representing the XML document obtained from the corresponding tree representation

The new unlabeled XML (test) documents are classified based on 'k' nearest neighbors obtained by the cosine similarity (Eq. 1) between the test documents and each of the documents in the training set. Similarity between two documents di and dj is computed using cosine function as given below:

$$sim(d_i, d_j) = \cos(d_i, d_j) = \frac{\sum_{i=1}^{n} A_i * B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} * \sqrt{\sum_{i=1}^{n} B_i^2}} \quad ....(1)$$

where A and B are the vectors representing the term frequency vectors of the documents d_i and d_j respectively. 'k' Nearest Neighbor (kNN) [3] is a well known, simple, but effective and powerful lazy learning algorithm used for the classification. Decision in kNN is based on the entire training data set and upon the 'k' nearest neighbors, where neighbors are defined based on the similarity/dissimilarity measure. The value of 'k' is usually an odd number greater than 1. If k=1, then the decision is based on only one neighbor and the algorithm is simply called the nearest neighbor algorithm. However, choosing an optimal value of 'k' is still a challenge, which straightforwardly affects the performance of kNN.

Experiments and Results

The effectiveness of our methods is tested on MEDLINE data set [15] which is available as a set of text files formatted in XML at no cost to the licensee. A brief description of MEDLINE data set is given in section 4.1.

MEDLINE data set

MEDLINE is a rich source of biomedical text that lends itself well to research on text mining, information extraction, and natural language processing in biomedical domains. It is the largest component of PubMed (http://pubmed.gov/), the freely accessible online database of biomedical journal citations and abstracts created by the U.S. National Library of Medicine (NLM®).

Approximately 5,400 journals published in the United States and more than 80 other countries have been selected and are currently indexed for MEDLINE. A distinctive feature of MEDLINE is that the records are indexed with NLM's controlled vocabulary, the Medical Subject Headings (MeSH®). Coverage includes bibliographic information for articles from academic journals letters, editorials, and case studies covering medicine, nursing, pharmacy, dentistry, veterinary medicine, and health care.

Results

All documents belonging to a specific category are available as a single large XML file in MEDLINE data set. This large file is split into smaller but complete XML documents. We have considered three categories A, B and C where each category consists of 3000 XML documents. Out of these 3000 documents, 2400 documents are used for training and remaining 600 documents are used as test set to check the performance of the algorithm. For the sake of simplicity, the attributes, for example, DOCID = 123, are not used for the purpose of classification. 'k' nearest neighbors are selected by measuring the cosine similarity between each unlabeled XML document in the test set and each of the document in the training set.

To check the effect of 'k' on the performance of the classifier, values 1, 3 and 5 were used as 'k' values in our experiments and it is observed that it shows better results when k=5. The measures viz., Precison, Recall and F-score (Eq 2) are used to evaluate the performance of the classifier developed.

$$precision=\frac{TP}{TP+FN};recall=\frac{TP}{TP+TN};F=2.\frac{precisionrecall}{precision+recall} \quad ....(2)$$

where, TP - true positives - documents that are assigned to the right category, TN - true negatives - documents that are wrongly assigned to a category, FN - false negatives – documents should have been assigned to the category being considered. F-Measure is a performance metric for classification model that is based on the weighted average of the precision and recall, where an F score reaches its best value at 100% and worst score at 0%.

The results obtained are tabulated in the form of confusion matrix and evaluation metrics as shown in Table 1 and Table 2 respectively. Fig 4 gives a comparison of F-scores for 'k' values 1, 3 and 5. To study the effect of the training set on the performance of kNN algorithm, we conducted experiments by varying the size of the training set keeping the same test set and the Fscores obtained for k = 1, 3 and 5 are tabulated as shown in Table 3. Initially, we started with 900 documents in the training set with each category consisting of 300 documents and increased the number of documents by 100 in each category. The comparison of Fscores obtained for varying training set is illustrated in Fig 5. It can be observed that increasing the training set need not necessarily improve the performance of the classifier. However, if the training set is selected carefully in a supervised manner such that it encompasses the representative set of documents describing a particular category, the performance of the classifier can boost up with the increase in the size of the training set. But, in our experiments, the documents in the training set were selected in an unsupervised manner.

Conclusion

In this paper, we have presented an algorithm based on content and structure for the classification of XML documents. XML documents are represented as vectors, where each term in the vector is obtained by prefixing the key path to each of the content bearing word in the leaf nodes. Experiments were carried out on a subset of XML documents belonging to MEDLINE data set for different values of 'k' and also by varying the size of the training set and the results are tabulated.

References

[1] Abdelhamid Bouchachia, Marcus Hassler (2007) *Proceedings of the 2007 IEEE Symposium on Computational Intelligence and Data Mining, CIDM*, pp. 390-396.

[2] Buneman(1997) *In Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pp.117-121, Tucson, Arizona, United States.

[3] Cover T. M., Hart P. E. (1967) *IEEE Transactions on Information Theory* 13 (1), pp. 21–27.

[4] Salton G., Wong A., Yang C. S. (1975) *In Communications of the ACM*, Vol.18 (11) pp.613-620.

[5] Georges Gardarin, Huaizhong K. O. U., Alain d'HEYGERES (2002)

[6] Hui-I Hsiao(2005) *Emerging Information Technology Conference*.

[7] Jianwu Yang and Fudong Zhang(2007) *INitiative for the Evaluation of XML Retrieval* pp. 234-244.

[8] Joe Tekli, Richard Chbeir, Kokou Yétongnon(2009) *Computer Science Review* 3(3), pp. 151-173.

[9] Junwei Wu, Jian Tang, Bipin C. Desai (Ed.), (2008)*12th International Database Engineering and Applications Symposium (IDEAS 2008), ACM International Conference Proceeding Series* 299 ACM 2008, pp. 131-137.

[10] Kjersti Aas, Line Eikvil(1999)*Technical report*, *Norwegian Computing Center.*

[11] Ludovic Denoyer, Patrick Gallianari, Anne Marie Vercoustre(2006) Report on the XML Mining Track at INEX *INEX 2006*, *ACM SIGIR Forum*, Vol. 41(1), pp. 79 – 90, June 2007.

[12] Shoaib M., Shazia Arshad, Shah A., Amjad Ali, (2006) *18th National Computer Conference, Saudi Computer Society*, pp. 147-150.

[13] Megha Gupta, Naveen Aggarwal(2010) *NCCI 2010 -National Conference on Computational Instrumentation CSIO*, pp. 128-131, Chandigarh, INDIA.

[14] Mohammed J. Zaki, Charu C. Aggarwal, (2003) *Proceedings of the 9th ACM SIGKDD*

international conference on Knowledge discovery and data mining SIGKDD '03, pp. 316 – 325.

[15] *NLM: Leasing data from the National Library of Medicine.*

[16] http://www.nlm.nih.gov/databases/leased.html

[17] Saptarshi Ghosh and Pabitra Mitra(2008) *IEEE ICPR*, pp.1-4.

[18] Swathy Giri, Aravind Chandramouli, and Susan Gauch(2004) Information and Telecommunication Technology Center, *Technical Report:* ITTC-FY2004-TR-8646-37, University of Kansas.

[19] Zhang Na, Zhang Dongzhan, Yu Ye and Duan Jiangjiao(2010) *Proceedings of the Third International Symposium on Computer Science and Computational Technology (ISCSCT'10)* Jiaozuo, P. R. China, pp. 426-430, August 2010.

Table- 1- Confusion matrix

Known Class	Predicted Class													
	k=1					k=3					k=5			
		A	B	C			A	B	C			A	B	C
	A	150	32	18		A	176	16	8		A	179	11	10
	B	60	109	31		B	55	115	30		B	53	113	34
	C	10	71	119		C	10	61	129		C	7	47	146

Table-2- Evaluation metrics

Class	k=1			k=3			k=5		
	Precsion	Recall	Fscore	Precsion	Recall	Fscore	Precsion	Recall	Fscore
A	68	75	71	73	0.88	80	75	90	82
B	51	55	70	60	58	59	66	57	61
C	71	60	65	77	65	70	77	73	75

Table- 3- Fscores obtained for by varying the training set

Sl. No.	# of documents in the Training set	k=1				k=3				k=5		
		A	B	C		A	B	C		A	B	C
1	900	68	60	86		77	62	91		80	65	92
2	1200	69	63	87		79	68	92		80	70	93
3	1500	68	60	86		79	69	93		81	70	93
4	1800	69	60	87		80	71	93		79	71	93
5	2100	72	62	87		81	69	90		81	71	90
6	2400	71	70	65		79	58	70		82	61	75

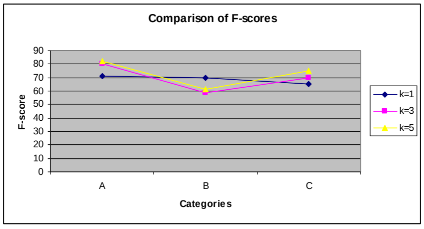

Fig. 4. Comparison of F-scores for k=1, 3 and 5

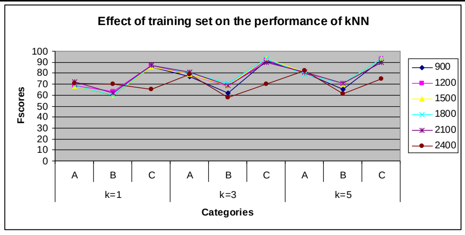

Fig. 5. Comparison of Fscores for varying training set