ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 2, May 2014

International Conference On Advances in Computer & Communication Engineering (ACCE - 2014)

on 21st & 22nd April 2014, Organized by

Department of CSE & ISE, Vemana Institute of Technology, Bengaluru, India

Classification of XML Document by Extracting Structural and Textual Features

Gnana Vardhini. H[1],Anju Abraham[2]

Student, Dept. of CSE, T. John Institute of Technology, Bengaluru, India[1]

Assistant Professor, Dept. of CSE, T. John Institute of Technology, Bengaluru, India[2]

ABSTRACT: In this paper the XML document classification is done by both structural and content based features. By this classification informative feature vectors are represented. In structural extraction, the tree-mining algorithm is used. For textual extraction, the algorithm is developed by using fuzzy c-means clustering algorithm. Once the classification is done the supervised classification algorithm is used which combines both structural and textual feature vectors. From which we get the classifier model. In this classification we can obtain 85% to 87% classification accuracy, which is more than the previously achieved classification accuracy.

KEYWORDS: XML document, textual information, tree-mining algorithm, structural information, soft clustering.

I. INTRODUCTION

The digital libraries, online news feeds and weblogs are stored as XML documents. As the size of documents grow faster and larger, Data Mining techniques become necessary to facilitate the organization of the documents for browsing. In the past many have done the research for this classification [6]. This automatic classification has been identified as one of the requirements of future information systems [10], [17], [19].

The goal of XML document classification is to build a classifier model that automatically assigns XML document to the existing category. Most of the existing classifier models either go for only structural classification or textual classification. The feature reduction has been used for preprocessing phase to reduce the complexity of classification model and to improve the accuracy of the classification process.

In this paper, the attempt has been done to capture both textual and structural information of a XML document and integrate them into the process of learning and predicting the class labels. The main target is to build a new dataset from XML documents by extracting different features that represent different aspects so that the classical classification techniques can be applied to classify XML documents. For this we need feature extraction and feature reduction to reduce sparsity and represent documents in a compact and informative feature space, any suitable model could be used in order to build the classifier. The process of feature extraction and reduction can improve the classification accuracy regardless of the final classification method being used in the context of XML document classification.

The details of the feature extraction would be different types of documents. Images. links and tables are considered feature extraction of the HTML document. Here the focus is done only on the XML document to get the accuracy in the document classification.

In the fig 1 the feature extraction and feature-based XML classification is depicted. Different types of features with range of values have been assigned hypothetically. As mentioned before both feature vectors have been extracted and then concatenated to obtain single vector.

In textual-based feature extractor component clusters the words through the distribution over the set of class labels and then the degree of association between documents and clusters is computed. The accuracy of classification can be affected as there are different clusters for different features and different clustering methods. Hard clustering algorithms assigns one object to only one cluster, while soft clustering algorithm assigns one object to many clusters with different membership degrees. The hard clustering or soft clustering algorithm is chosen based upon the average number of words in the document. In corpus as the number of words are more hard clustering algorithm is preferred where as in the case of the document soft clustering algorithm is used.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 2, May 2014

International Conference On Advances in Computer & Communication Engineering (ACCE - 2014)

on 21st & 22nd April 2014, Organized by

Department of CSE & ISE, Vemana Institute of Technology, Bengaluru, India

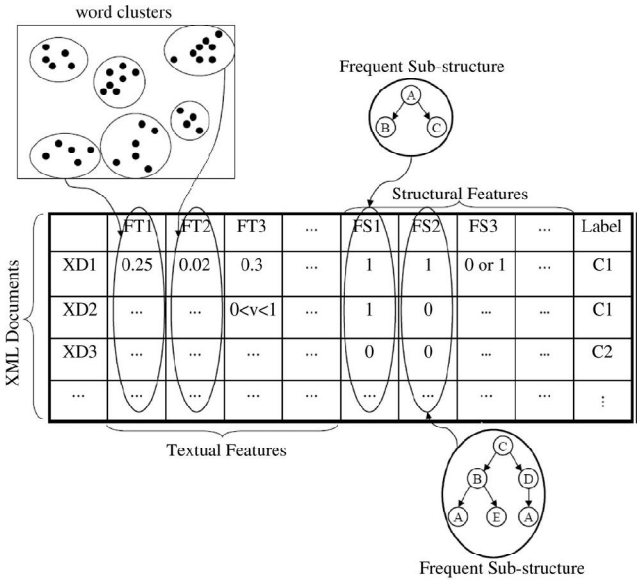

Fig 1: Using clusters as textual and frequent substructures as structural features.

In structural feature extraction the tag structure is converted to tree structure as parent-child relationship. Here we get frequent subtree, which is mined and assigned as a feature. Frequent pattern mining is used for frequent tree mining [18]. In this paper all the XML documents are assumed to be in tree structure.

In rest of the paper we have, Section II the existing techniques. Section III has the approach for XML classifier model and Section IV has expected result.

II. LITERATURE SURVEY

As mentioned in the introduction, the XML document classification was conducted in two techniques: textual-based and structural-based. Many approaches have been made to improve the performance, efficiency and accuracy of the classification.

Textual-based feature extraction: L.D. Baker and A. McCallum had proposed this technique using word clustering [5]. But this technique failed to address the sparse data and hence there was loss in the percentage of accuracy.

Text applications perform well but text classification approaches do not perform effectively for XML classification because most of the information is buried in the structure of XML documents. This type of information is ignored in textual-based text classification algorithms [19]. Structure will give the reliable information for classification. Hence both textual and structural features of XML documents are considered for classification.

Structural-based feature extraction: M. Theobald, R. Schenkel and G.Weikum [17] developed a technique which used a schema-less XML document for structural classification. But they failed to show the improvement in efficiency as they considered the tag length of 2. Antonellis [2] proposed XEdge where XML document is represented as LevelEdge. Then based on the LevelEdge the documents are grouped. M. J. Zaki and C. Aggarwal proposed XRules [19] where rules are generated for classification. But the disadvantage in this technique is that many rules are generated for using the existing rules. And this made difficult in storing rules, retrieve the rules and sort the rules. By using many rules the classification accuracy is achieved.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 2, May 2014

International Conference On Advances in Computer & Communication Engineering (ACCE - 2014)

on 21st & 22nd April 2014, Organized by

Department of CSE & ISE, Vemana Institute of Technology, Bengaluru, India

Greco [11] proposed a hybrid model where the collaboration technique is used for both textual and structural extractions. Here the peer to peer network is considered in a distributed framework. The problem was faced because the clustering has been done in a collaborative way and each peer is considered to be one cluster.

III. PROPOSED SYSTEM

The characterization of both content and structure of XML document for XML classification, every XML document is represented as a feature vector, features represent data item attributes in the form f1= v1Λ f2= v2 Λf3= v3Λ ..... Λ fn= vn, where fi is feature (attribute), and vi is its value.

In XML classification the feature set has textual features and structural features. Frequent structural patterns are captured as frequent subelements by using frequent tree algorithm. Frequent patterns are chosen carefully and take advantage of the class label information where the rule of the type X→ Ci is considered, where X is the frequent subtree and Ci is the class label.

The structural rule mining algorithm extracts all structural rules with support and confidence greater than the threshold value. Support is the percentage of documents that contain the structure in the left-hand side of the rule, and confidence is the percentage of the documents belonging to the class variable. From this we can choose those subelements that are both frequent and are also able to discriminate between different classes. The first one is obtained by support threshold and the later from confidence threshold.

Textual features are constructed based on XML document after document preprocessing. As we use individual words as document features with their frequencies as feature values, it is easy and straight forward for preprocessing. But it has some disadvantages. First, as every single word is feature, the feature vector obtained is of high dimension because the number of words in a document will be large. This increases the complexity of problem, processing time and memory requirement. Second, when the online document classification is done, everytime the feature vectors change as new words from new documents should be considered. Third, the documents having small words will lead to sparsity for large portion of the feature values of feature vectors. The words are filtered in the process of the classification such that the developed classifier is better than naive classifier.

To overcome these problems the soft clustering is performed on words where each cluster is used as a feature. The fuzzy c-means algorithm is used as it is easy and fast to work with [7].

The fig 2 depicts the system architecture for the XML document classifier model with the data flow in the system. To perform the mapping between vectors and class labels the weights are assigned to all the keywords.

A. Structural Feature Construction

As mentioned earlier that support and confidence threshold values are used for structural rules, the structural feature vectors are generated. Every feature corresponds to an individual frequent subelement. The feature values are assigned irrespective of the class label matching to the consequent of the rule or not. The main goal is to use frequent subelement as features. Class labels are considered only in structural rule mining step to discriminate the power of the frequent patterns besides their frequencies.

Structural Rule Mining: The structural rule is defined as X⇒ Class with support and confidence threshold values, where X is a frequent subelement in the given set of XML documents and Class is one of the classes. The problem in structural rule mining is the structural rules which satisfy the threshold values of support and confidence are considered.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 2, May 2014

International Conference On Advances in Computer & Communication Engineering (ACCE - 2014)

on 21st & 22nd April 2014, Organized by

Department of CSE & ISE, Vemana Institute of Technology, Bengaluru, India

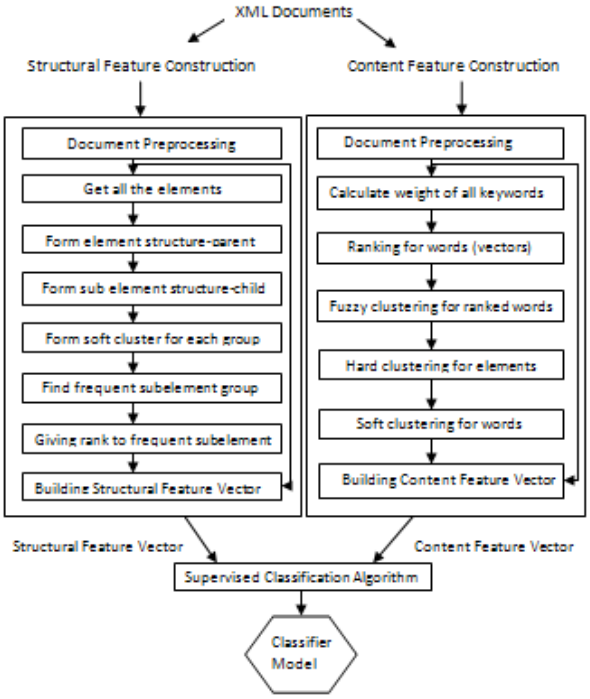

Fig 2: Architecture of the XML document classifier model.

The problem in structural rule mining is that the information extracted will be the information which satisfies the user defined threshold support and confidence. Here the information extracted is through structural rules. It can be divided into two subproblems. The first subproblem is finding frequent subelements from the given dataset. But the support of the frequent subelement will be more than threshold support value. The second subproblem is to generate the structural rule from frequent subelements which satisfy the minimum threshold confidence. The issue in the XML classification is because the XML document contains both tags and text. The general approach used for this is the use of document preprocessing, which is done in the first step. Here, first the document content is removed. Second, the tag structure of the XML document is transformed to the tree structure to preserve the hierarchical structure of the document. Finally, the structural rule mining is applied to these tree structures which represent the XML document. Here each tree structure is one XML document.

The algorithm here used is the XMINER [19] for structural rule mining as shown in the fig 3. This will find all the structural rules which have support and confidence more than threshold value. Then the XML documents are modeled as ordered, labeled and rooted trees. After this the tree structures are converted as transactions to preserve the hierarchical structure of the trees. It is known that if the subelements are not frequent then none of its super elements are not frequent. By this the efficiency is obtained for the structural feature extraction.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 2, May 2014

International Conference On Advances in Computer & Communication Engineering (ACCE - 2014)

on 21st & 22nd April 2014, Organized by

Department of CSE & ISE, Vemana Institute of Technology, Bengaluru, India

XMINER $(\mathcal{D}, \pi_i^{\min} \forall i = 1 \cdots k)$:
  $[P]_0 = \{$ frequent 1-subtrees for any class$\}$;
Enumerate-Xrules$([P]_0)$;

Enumerate-Xrules$([P])$:
  for all elements $x \in [P]$ **do**
    $[P_x] = \emptyset$;
    for all elements $y \in [P]$ **do**
      $\mathbf{R} = x \otimes y$;
      $\mathcal{L}(\mathbf{R}) = \mathcal{L}(x) \cap_{\otimes} \mathcal{L}(y)$;
      if for any $R \in \mathbf{R}$, R is frequent for any class
        then $[P_x] = [P_x] \cup \{R\}$;
    Enumerate-Xrules$([P_x])$;

Fig 3: XMINER- Tree mining for classification

B. Textual Feature Construction

After removing all the structural tags from the documents, to form the textual feature vectors from bag of words (BOW), we have steps to follow. 1) Document Preprocessing. 2) Calculate weight of all the keywords. 3)Ranking the words. 4) Fuzzy clustering the ranked words. 5) Hard clustering for elements. 6) Soft clustering for words. 7) Building Textual Feature Vectors.

Document Preprocessing: To find the occurrence of the given word in the document the linear search is used on all the documents. Every document D in the set can be represented as D = { (wi, f (wi)) | wi ∈ w }, where wi is a distinct word in the complete word set w and f (wi, D) is the normalized frequency of the word wi in the document D. Here the sum of all word frequencies for every document is 1. By using this representation every document can be modeled as a vector of < word, word frequency > pairs.

Calculate weight of all the keywords: All the keywords that are taken from the document D will be given a weight, from which the rank to those words can be assigned easily in the next step.

Ranking the words: Once the weight is calculated, ranks are assigned to the words. The words that occur frequently will have the highest rank. By this we can select the words or give the threshold value to the rank to select the words. For example we can select the frequently occurring words which are within top 5 ranks.

Fuzzy clustering of the ranked words: Clustering technique is used to estimate the closeness among the objects. As we are using fuzzy c-means algorithm the objects (words) are represented as vectors. Every word is mapped to a vector of length of n, where n is the number of distinct class labels. The vector is normalized to the sum of all values associated with the coordinates to 1.

Hard clustering and Soft clustering: In hard clustering one word can be assigned to one cluster. Whereas in soft clustering one word is assigned to more than one cluster. Fuzzy c-means algorithm will improve the accuracy of the classification as explained in the literature survey.

Building Textual Feature Vector: Feature vector is constructed using training and testing samples. The normalized feature vector can be passed to any classification model.

C. Building Classifier Model

In order to incorporate both structural and Textual features into the learning process, structural and textual feature vectors associated with any particular XML document are first merged into one feature vector. This feature vector represents both structural and textual characteristics of XML document. Then these feature vectors are used to train classifier model. The classifier model is built using support vector machines and decision tree algorithm. Validation techniques are used to get the accuracy of the classifier model.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 2, May 2014

International Conference On Advances in Computer & Communication Engineering (ACCE - 2014)

on 21st & 22nd April 2014, Organized by

Department of CSE & ISE, Vemana Institute of Technology, Bengaluru, India

IV. EXPECTED RESULT AND CONCLUSION

Expected Outcome: By applying support vector machines and decision tree algorithms using feature vector representation of XML document datasets, outcome can be achieved 85% of classification accuracy, which are higher than accuracy achieved by other XML document classifier.

Table 1
Characteristics of XML document

Dataset	# of Documents	Distribution in classes	
		edu	other
LOG1	8074	1962	6112
LOG2	7047	1686	5721
LOG3	7628	1798	5830

Conclusion: In this paper the XML document classification is done using both structural and textual features of the document. For structural feature extraction the tree is build and textual feature extraction the clusters are formed using the appropriate algorithms.

REFERENCES

[1] Mohammad Khabbaz, Keivan Kianmehr, and Reda Alhaji, " Employing Structural and Textual Feature Extraction for Semistructured Document Classification" IEEE sytems, man and cybernetics, Vol. 42, pp. 1566-1577, Nov 2012.

[2] P. Antonellis, C. Makris, and N. Tsirakis, " XEdge: Clustering homogenous and heterogeneous XML documents using edge summaries" ACM Symp.Appl.Comput., Ceara, Brazil, 2008.

[3] C. Aggarwal, S. Gates, P. Yu, " On the merits of using supervised clustering to build categorization systems" ACM SIGKDD, pp.352-356, 1999.

[4] R. Agarwal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications" ACM SIGMOD, pp. 94-105, 1998.

[5] L. D. Baker and A. McCallum, " Distributional clustering of words for text classification" ACM SIGIR, pp. 96-103, 1998.

[6] M. Berry, "Survey of text mining: Clustering, Classification, and Retrieval" Springer, 2004.

[7] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The Fuzzy C-means Clustering algorithm" Comp.Geosci., Vol.10, no. 2-3, pp. 191-203, 2003

[8] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation" J. Mach. Learn. Res, pp. 993-1022, 2003.

[9] I. Dhillon, S. Mallela, and R. Kumar, "Enhanced word clustering for hierarchical text classification" ACM KDD, pp. 191-200, 2002.

[10] C. Garboni, F. Massegalia, and B. Trousse, " Sequential pattern mining for structure-based XML document classification" Workshop Initiative Eval. XML Retrieval, pp. 458-468, 2005.

[11] S. Greco, F. Gullo, G. Ponti, and A Tagarelli, "Collaborative clustering of XML documents" J. Comput. Syst. Sci., Vol.77, no.6, pp.988-1008, 2011.

[12] M. A. Hall, "Correlation-based feature selection for discrete and numeric class machine learning" ICML, pp. 359-366, 2000.

[13] T. Hofmann, "Probabilistic latent semantic indexing" ACM SIGIR Conf. Res. Development Inf. Retrieval, pp. 50-57, 1999.

[14] Z. Huang, H. Chen, and D. Zeng, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering" ACM TOIS, pp.116-142, 2004.

[15] D. A. Hull, "Improving text retrieval for the routing problem using latent semantic indexing" ACM SIGIR, pp. 282-289, 1994.

[16] T. Joachims, " Text categorization with support vector machines: learning with many relevant features" ECML, pp. 137-142, 1998.

[17] M. Theobald, R. Schenkel, and G. Weikum, "Exploiting structure, annotation, and ontological knowledge for automatic classification of XML data" WebDB, pp.1-6, 2003.

[18] M. J. Zaki, "Efficiency mining frequent trees in a forest" ACM SIGKDD, pp. 71-80, 2002.

[19] M. J. Zaki and C. Aggarwal, "XRules" An effective structural classifier for XML data" ACM SIGKDD, pp. 316-325, 2003.

[20] S. Kutty, T. Tran, R. Nayak. and Y. Li, "Clustering XML document using closed frequent subtree: A structural similarity approach" Springer, pp. 183-194, 2008.