Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 46 (2015) 194 – 201

Procedia Computer Science

International Conference on Information and Communication Technologies (ICICT 2014)

Enhanced Associative Classification of XML Documents Supported by Semantic Concepts

Thasleena N.T.[a,]*, Varghese S.C.[a]

[a]*Dept. Of Computer Science and Engineering, Rajagiri School of engineering and Technology, Cochin- 682039, Kerala, India*

Abstract

A novel approach based on supervised classification has been proposed to classify a given collection of XML documents based on rule based classifier by semantically enriched structure and content features. The proposed methodology overcomes the drawbacks of the existing technologies by accomplishing the classification by utilizing not only the structure and content features but also context. It applies ontological information into structural and content based features from the XML documents and transforms it into transaction formats onto which FP-growth algorithm is executed to generate association rules. An associative classifier is constructed by eliminating irrelevant rules from the generated association rule.

Peer-review under responsibility of organizing committee of the International Conference on Information and Communication Technologies (ICICT 2014)

Keywords: XML documents;Classification;Ontology;FP_growth;Wordnet;Extended Lesk;Feature extraction.

1. Introduction

Analyzing and mining of XML data has gained much popularity from the knowledge discovery and data mining community in the recent years. This profound interest has led to the storing of large data repositories such as digital libraries, online news feeds, and weblogs in XML format. The effective browsing and searching of this enormous collection of documents substantiate the need of data mining techniques for the organization of documents. The growing amount of semi-structured data and the widespread adoption and use of XML as the standard for semi-structured data are the main reasons for this. XML is a standard model to store and transport data. The flexible and

* Corresponding author. Tel.: +91-9995561287.
E-mail address: thalu555@gmail.com

Peer-review under responsibility of organizing committee of the International Conference on Information and Communication Technologies (ICICT 2014)
doi:10.1016/j.procs.2015.02.011

expressive nature of XML allows to organize textual contents into hierarchical structures. The complex data structure of XML format, its structured dimension and the content (especially text) dimension and the possible heterogeneity of the documents raise a new challenge for document mining. The structure information alone or both the structure and the content of the documents may be considered depending on the application or the mining objective.

In classification problem, the training data (input dataset) consists of a set of multi-attribute record (features) and class variable. This class variable taken its value from a distinct set of classes. The training dataset is used to build a learning model (classifier) which associate the features extracted from training data to the class variable. The test documents in the prediction phase consist of a set of records with the features, but its class variable is unknown. With the help of training model test document can predict its class. Supervised classification of XML data construct a learning a model by extracting discriminating features from prelabeled XML documents. The generated classifier will acquire the ability to predict the classes of an unknown XML documents by examining its features. Every XML document includes both logical and physical structures. So based on these two information XML can be classified in two ways. One approach uses only the structural information of XML data in classification. Another one performs the classification, by considering both the content and structural information of XML data.

This paper proposes a new approach by incorporating semantic information on the structure and content information extracted from the XML document. Instead of using tags and terms as features of an XML document, the main idea is to map these into an ontological concept space. This method applies a FP growth algorithm[9] for generating association rules which not only diminishes time, but also the memory usage compared to existing techniques like Apriori algorithm used for the same. This paper first mines features of an XML document by capturing the structure and content features (i.e., attaching tag path with terms (or noun phrases) in leaf nodes) ,then ontological information (i.e., mapping words to word senses) is applied to this feature. So we will get more expressive features for automatic classification. Wordnet[7] is used as an underlying ontology and extended lesk [4] algorithm is used for word sense disambiguation. These context based features are given into FP-growth algorithm [9] for constructing association rules and compact classifiers are built from this rule. We have analyzed the performance of proposed methodology using Wikipedia XML dataset.

2. Related works

In[2] a novel methodology (XRULES) for the classification of XML documents based on structural information has been proposed. It has focused on the use of rule based classifiers as an effective tool for XML classification. XRULES extracts frequent embedded sub trees using XMINER algorithm and associate it with each known class of the XML documents under consideration. Structural rules for prediction are generated from these sub trees. These rules determine the likelihood for an XML document to come under a particular class. The limitation of this method is a large number of rules are produced by training phase, and it is very difficult to store the rules, retrieve the rules. Furthermore, contents of XML documents are completely avoided by XRULE.

In Paper[6] XML documents are classified by mining frequent attribute trees from a group of labeled XML documents. An attribute tree is a subtree with attributes associated with the elements of original XML documents. In training phase it computes frequent patterns based on support value from the training data set. From these frequent patterns it selects emerging patterns. Frequent patterns become emerging pattern if it frequently occurs in the XML tree of one class and rarely occurs in another class. Finally, these features are used as a binary feature in a decision tree algorithm for constructing a classifier. The Major drawback of this work is that the search for redundant subtrees within each class is highly time consuming, which intern make, model induction next to impossible when the number of XML documents exceeds a limit. It also ignores the content features in the classification.

In[5] a methodology for classifying XML documents without schema has been proposed by constructing expressive feature spaces. Feature space was filled with ontological, structural and content related information. In addition to the conventional term frequency vectors, it used XML twigs and tag paths as an enhanced feature that can be attached with text term occurrences in XML elements. And also it used ontological information using Wordnet for the purpose of more expressive feature space by mapping word into word sense. For creating ontological information word senses are identified for each word in tags/terms from Wordnet. It has used word sense

disambiguation method to find a correct sense for each word. The original tag or terms of the structural features are replaced by word sense ids. The final feature vectors are represented using this ids. The Synonyms are also recognized in the test phase where all tags/terms are converted along with their disambiguated synset ids. The test vector is expanded in the prediction phase by correlating concepts in training feature space with unknown concepts of the test document. Support Vector Machine was recruited to perform the classification of XML documents using the generated feature space.

In paper[1] an approach called XCSS is used as an associative rule based classifier for classification of XML documents. Here the key path is taken as structure and content features. The path from the root element to each term in the content of an XML document becomes a unique key path. It uses CAR (class association rule) to model association between subset of features and distinguished classes. Classification is divided into model learning and prediction. The model learning phase learns associative classifier from a database of labeled XML tree. Latter use tree to predict the classes for unlabeled one. Here XML data are represented by transitional form. It uses MINECAR procedure which is an enhancement of aprioiri alogorithm (Aggarwal) by using minimum support and minimum complement class support to produce meaningful CAR from training data. After that PRUNE method is used to build a compact classifier using these rule set R. Finally prediction is performed on unlabeled document tree based on the classifier obtained from model learning.

3. Proposed methodology

The proposed system for the classification of XML documents involves two main phases, learning phase followed by the prediction phase. The Learning phase constructs an associative classifier C from labeled XML trees database. The test phase exploits classifier *C* to predict the class of unlabeled XML trees. The proposed architecture is shown in Fig. 1.

3.1. Learning phase: feature extraction

Structural features are extracted by tracing the path from the root tag to leaf tag in each XML tree. In XCCS[1], to the tag path individual terms are appended to generate final structure and content features. Features extracted from following XML data based on XCCS are: [‘article/body^video’,’article/body^game’].

```
<article>
  <body>
    video game
  </body>
</article>
```

In proposed method instead of terms, noun phrases are appended to the tag path. In order to extract noun phrase, noun phrase chunking is used. Before doing chunk operation, the first step is to tag the leaf content using any of the tagger. To create NP-chunker, first define a chunk grammar, consisting of rules that indicate how sentences should be chunked. Using this grammar, create a chunk parser, and test it on our XML contents to be chunked. Based on our method features of same XML data will be [‘article/body^video game’].

3.2. Learning phase: feature selection

We include all tags and leaf contents in this feature extraction process, but for tractability and also noise reduction only a subset of these features is selected for the training process. Chi square method [11] is used for feature selection. Select the n_features from features with the highest values for the χ^2 (chi-square) statistic.

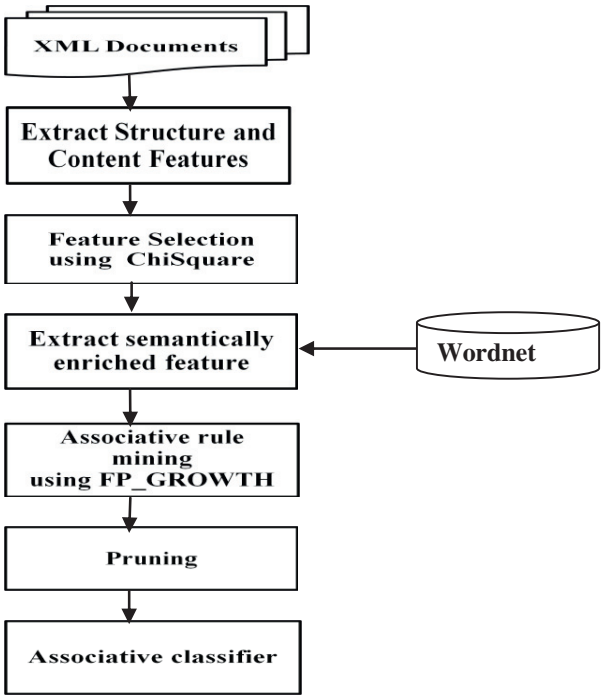

Fig. 1. Overall architecture of proposed system

3.3. Learning phase: semantic concept on features

In XCCS[1] ambiguities in the semantics of content and structural features may prevent effective XML classification. This drawback is resolved in the proposed system by exploiting improved semantic information and thereby eliminating ambiguities along with their side effects. Rather than using tags and terms directly as features of an XML document, a fascinating idea is to map these into an ontological concept space. So we can enrich the structure and content feature using semantic information taken from Wordnet. Here each tag name/noun phrases are mapped into its disambiguated sense ids. The following explanation describes how we extract ontological concept in structure and content features of XML documents.

3.3.1. Semantic on structural features

Tag paths are the regular way for extracting structural features from XML documents. It is to be highlighted that the context free tag names are mapped to the apt semantic concept without extracting context free tag name for XML document. The concept extracted from the ontological space is associated with their tag names. Thus features extracted from XML tag names can be appended into the semantic concept by exploiting lexical ontological knowledge obtained.

In this work we have used WordNet as ontological knowledge base due to its increasing scope and public availability. Wordnet combines words with the same meaning into one group and named it as synsets (sets of synonyms). Synsets are connected in the Wordnet hierarchy through a set of predefined relations ((hypernymy/hyponymy, (meronymy/holonymy)). At the time of mapping of tag name into semantic concepts, word sense disambiguation is essential to decide most appropriate sense for each tag name.

The method we have used for the semantic concept on structure follows the method [3]. Here the disambiguation of senses for tag names is done by selecting the most suitable senses based on a path-context. A path context is

represented by a semantic network built on all the possible senses associated to the tags of a specific path. Synset graph is created for each tag in specific path and it is used to disambiguate the senses of tags in that path. Details about construction of semantic network are given in[3]. Nodes of synset graphs are pairs of tag name and its sense. Edge weight between any nodes reflects the depth of semantic relationship between the concepts of the corresponding nodes on which the edge is incident. The disambiguation of the tag names in given path is accomplished by finding the maximum-weight path in synset graph of given path. Further details are given in[3].

3.3.2. Semantic on content features

We have used Extended lesk algorithm[4] for finding correct sense for content features (i.e. leaf elements attached to the tag path). In this method, a window is constructed for target word and set of words around it. Then extract set of candidate senses for each word in the window. Let windows contains 2n+1 words and each word is represented as w_i where –n<=i<=n and target word is placed at w_0. Also assume $|w_i|$ represent the total number of sense of word w_i. Next step is to assign to each sense of the target word is assigned with a SenseScore that is computed by adding together the relatedness scores obtained by comparing the sense of the target word with every sense of every other words in the window. The sense score for each sense of target word is computed as follows.

$$SenseScore_k = \sum_{i=-n}^{n} \sum_{j=1}^{|w_i|} relatedness(s_{0,k}, s_{i,j}), i \neq 0 \quad (1)$$

The sense with the highest SenseScore is taken to be the most appropriate sense for the target word. The details about relatedness function are given in[4].

3.4. Learning phase: rule mining

The learning process in proposed method is sketched in model learning algorithm shown below. It receives four input parameters: a dataset D of XML trees, a set *F* of semantically enriched features, a set *L* of class labels in database *D* and one global threshold *τ*. We have to create transactional representation of each tree of a XML database D (lines 3-6). FP growth algorithm[9] is applied to semantically enriched features and frequent items are generated based on minimum threshold value. FP growth algorithm is applied class wise to the set of features from same class to retrieve the frequent features. After discovering frequent itemset, next step is to construct class association rule from frequent item. A CAR associates the occurrence of a certain combination of features in a transactional representation of XML tree with one particular class. Only rules with confident value greater than minimum threshold is taken as final rule. Finally, the rule set **R** is distilled into associative classifier *C* through the pruning method.

MODEL-LEARNING (*D, L, F, τ)*

1. R ← *∅;*
2. *D'* ← *∅;*
3. for each t ∈ *D do*
4. s← *{F'| F' ∈ F }*
5. *D'* ←*D' ∪ {s};*
6. end for
7. *FI*← *FP-TREE(F,D',τ) // Frequent Item creation*
8. for each frequent itemset I ∈ *FI do*
9. create rule of the form r : I → *class(I);*
10. if *confidence (r) > minimum_confidence*
11. R ← *R ∪ {r}*
12. *C=PRUNE(R)*
13. RETURN *C*

3.5. Learning phase: pruning

Rule mining step yields a large number of Class association rules, which over fit the training data and will provide contrasting predictions. This issue is overcome by constructing an accurate classifier from the rule set using

PRUNE method as in XCCS.

3.6. Testing phase

The XML documents in the test set should be classified by the final Associative classifier induced by final step of learning phase. These documents undergo feature extraction step as in learning phase. A set of semantically enriched feature will be obtained from each test XML document. These test features are compared with each antecedent of rules in associative classifier. Test document will be classified into the class whose rules are satisfied by the features of the XML documents to the maximum degree.

4. Results

The performance of the proposed methodology for XML document classification is established by exploiting Wikipedia data sets. *Wikipedia* is an XML corpus proposed in the INEX contest 2007[8]. It is used as a major benchmark for XML classification and clustering. The corpus contains 96, 000 XML documents representing very long articles from the digital encyclopedia. The XML documents are organized into 60 portals of Wikipedia. The potential of an XML document classifier is expressed with average precision (**P**), average recall (**R**), average F-measure (**F**) [10]. All tests are performed on a Linux machine; with an Intel dual core processor with 4 GB RAM 3.3 GHz in Python (version 2. 7) language. Table. 1. summarizes the effectiveness of the chosen proposed method and XCCS across the Wikipedia data sets. Fig. 2. Shows ROC curve of proposed system and Fig. 3. shows ROC of XCCS.

Table 1. Performance comparison: Wikipedia

Method	Dataset	Precision	Recall	F-measure
Proposed system	Wikipedia	0.88	0.83	0.84
XCCS	Wikipedia	0.77	0.78	0.78

5. Conclusion and future work

In this paper, we proposed a new framework for XML document classification based on rule based associative classifier Rule using semantically enriched features. Fp-growth algorithm is used for associative rule mining and Wordnet for finding semantic concepts on features. Experiments have shown that application of ontological knowledge information in textual and structural features improves the accuracy mainly because it provides context based classification with an enhanced feature sets.

Ongoing research aims to improve the scalability of an algorithm by incorporating parallel implementation method using map reduce program mode in hadoop frame work. Also aims to incorporate other methods of ontological knowledge into our method to go beyond the information provided by WordNet.

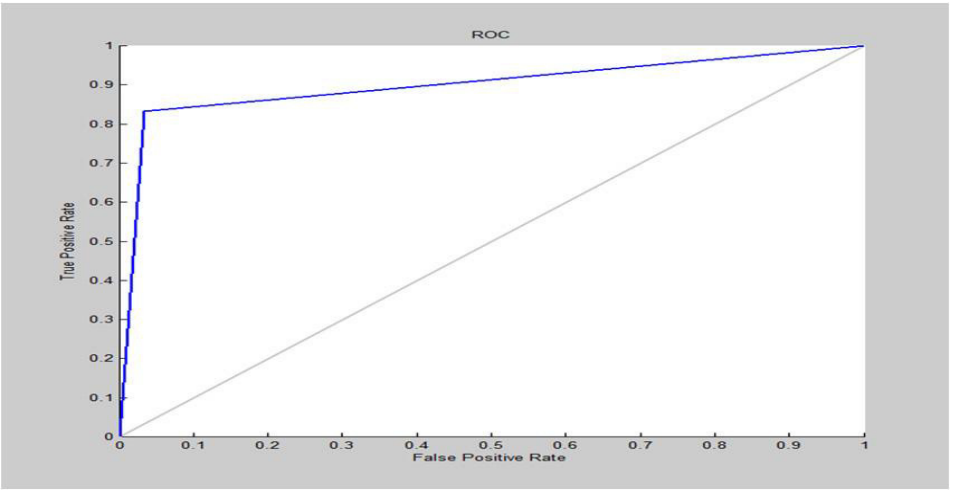

Fig. 2. ROC curve of Proposed System

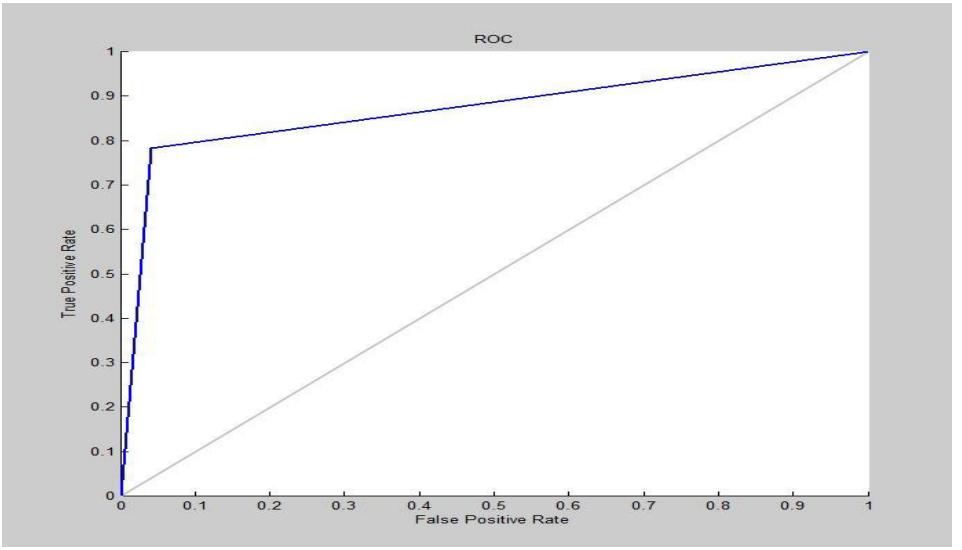

Fig. 3. ROC curve of XCCS[9]

References

1. Costa, G., Ortale, R., and Ritacco, E. *Learning* Effective XML Classifiers Based on Discriminatory Structures and Nested Content. *Knowledge Discovery, Knowledge Engineering and Knowledge Management Communications in Computer and Information Science* , vol. 348, p. 156-171, 2013.
2. Mohammed J. Zaki , Charu C. Aggarwal. XRules: An effective algorithm for structural classification of XML data, Machine Learning, v.62, p.137-170, 2006.
3. TAGARELLI, A. AND GRECO, S. Toward semantic XML clustering. In *Proceedings of the 6th Society for Industrial and applied Mathematics (SIAM) International Conference on Data Mining (SDM). p. 188–199, 2006.*

4. S. Banerjee and T. Pedersen. Extended Gloss Overlaps as a Measure of Semantic Relatedness. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI),* p. 805–810, 2003.
5. M. Theobald, R. Schenkel, and G. Weikum.. Exploiting structure, annotation, and ontological knowledge for automatic classification of xml data. *In Proceedings of Web and Database (WebDB) Workshop*, p. 1 – 6, 2003.
6. J. De Knijf. Fat-cat: Frequent attributes tree based classification. In *Proceedings of the Initiative for the Evaluation of XML Retrieval*, p. 485–496, 2007.
7. C. Fellbaum. *WordNet: An* Electronic Lexical Database. Massachusetts Institute of Technology (MIT) Press, 1998.
8. Denoyer, L., Gallinari. P. Report on the XML Mining Track at Inex 2007. *In ACM Special Interest Group on Information Retrieval (ACM SIGIR) Forum* 42 (1), p. 22–28, 2008.
9. J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. *In Special Interest Group on Management Data (SIGMOD)'00, Dallas, TX,* 2000.
10. Manning, C., Raghavan, P., Schutze, H. Introduction to Information Retrieval. Cambridge University Press, 2008.
11. Yang, Y. and Pedersen, J. O. A comparative study on feature selection in text categorization. In *Proceedings of 14th International Conference on Machine Learning (ICML-97),*Nashville, US, p. 412–420, 1997.