A Framework for Processing XML data Using Eclat Algorithm

Rohini L. Damahe¹, Veena Kulkarni²

PG Student, Computer Engineering, Thakur College of Engineering and Technology, Mumbai
Assistant Professor, Computer Engineering, Thakur College of Engineering and Technology, Mumbai

Abstract— since a huge amount of XML documents are available in digital form which represents a semi structured data. And it is one of difficult problems for users to obtain really necessary information efficiently from these XML files. In order to deal with such problems, many researchers are engaged in studies on Information Retrieval (IR) and Text Clustering. Association rules provide a better result and they can predict the mining patterns too. There are several mining algorithms of association rules. One of the most popular algorithms is Apriori that is used to extract frequent itemsets from large database and getting the association rule for discovering the knowledge. But Apriori rules are using horizontal frequent patterns, due to this many times vertical features may lost in mining. So our research proposed an idea of improving mining pattern by using Eclat algorithm for the XML data in vertical pattern and then cluster them efficiently.

Keywords— Frequent patterns, itemsets, rules, Eclat, Apriori, information retrieval, XQuery.

I. INTRODUCTION

In Data mining if the summarization system can make a effective summary, which can be a substitute of the original documents, the user can save the time to read. In this paper, therefore, we will discuss a summarization system under the following conditions:

- Target is a set of documents that are originally retrieved by an IR system on a certain topic and narrowed down to some small number of related documents.

- The output is an informative summary, which can be a substitute of the original (target) documents. In order to satisfy the second condition, the following points are important.

Comprehensibility The summary should include main content of target documents exhaustively.

Readability The summary should be a self-contained document and should be readable.

Many of recent researches put emphasis on the comprehensibility while keeping readability of summaries to some extent.

Classification is a well-known task in data mining that aims to predict the class of an unseen instance as accurately as possible. While single label classification, which assigns each rule in the classifier the most obvious label, has been widely studied [1], Another important task in data mining is the discovery of all association rules in data. Classification and association rule discovery are similar, except that there is only one target to predict in classification, i.e., the class, while association rule can predict any attribute in the data. In recent years, a new approach that integrates association rule with classification, named associative classification, has been proposed [1]. A few accurate classifiers that use associative classification have been presented in the past few years, such as CMAR [9], and CPAR [2].

In existing associative classification techniques, only one class label is associated with each rule derived, and thus rules are not suitable for the prediction of multiple labels. However, multi-label classification may often be useful in practice. Consider for example, a document which has two class labels “Health” and “Government”, and assume that the document is associated 50 times with the “Health” label and 48 times with the “Government” label, and the number of times the document appears in the training data is 98. A traditional associative technique like CBA generates the rule associated with the “Health” label simply because it has a larger representation, and discards the other rule. However, it is very useful to generate the other rule, since it brings up useful knowledge having a large representation in the training data, and thus could take a role in classification.

Feature selection forms an important subset within the much larger area of text classification. Correctly identifying the relevant features in a text is of vital importance to the task of text classification. Additionally, other methods for reducing dimensionality, such as pruning and clustering, can improve performance of text classification.

In this paper we are proposing a novel approach of mining association rules by Apriori and Eclat algorithm to enhance the properties of vertical and horizontal frequent pattern itemsets.
Before applying these mining techniques system first extracts the important words of the documents by applying Shannon info gain theory to the top words of the documents.

System Improved association rule mining technique of [3] which uses the XML data and XQuery more efficiently to identifies the frequent patterns.

The rest of the paper is organized as follows: Section 2 discusses some related work and section 3 presents the design of our approach. The details of the results and some discussions on this approach are presented in section 4 as Results and Discussions. Section 5 elaborates hint of some extension of the approach as future work and conclusion.

II. RELATED WORK

Text classification is an area within pattern recognition and classification that has been studied with increasing frequency as Internet usage becomes more commonplace. The goal is generally to assign a text to one or more classes based on some method that takes into account the contents of the text. There are many varied practical applications of text classification. The most well known of these applications is likely improved spam filtering techniques. Search engines may also take advantage of text classification techniques to return more accurate results to the user [4].

Many types of text representations have been proposed in the past. A well-known one is the bag of words that uses keywords (terms) as elements in the vector of the feature space. In [5], the tf*idf weighting scheme is used for text representation in Rocchio classifiers. In addition to Tfidf, the global IDF and entropy weighting scheme is proposed in [6] and improves performance by an average of 30 percent. Various weighting schemes for the bag of words representation approach were given in [7] [8]. The problem of the bag of words approach is how to select a limited number of features among an enormous set of words or terms in order to increase the system’s efficiency and avoid over fitting [9]. In order to reduce the number of features, many dimensionality reduction approaches have been conducted by the use of feature selection techniques, such as Information Gain, Mutual Information, Chi-Square, Odds ratio, and so on. Details of these selection functions were stated in [10], [9].

Pattern mining has been extensively studied in data mining communities for many years. A variety of efficient algorithms such as Apriori-like algorithms [11], Prefix Span, FP-tree, SPADE, SLP Miner, and GST [12] have been proposed.

These research works have mainly focused on developing efficient mining algorithms for discovering patterns from a large data collection.

However, searching for useful and interesting patterns and rules was still an open problem [13], [14], [15]. In the field of text mining, pattern mining techniques can be used to find various text patterns, such as sequential patterns, frequent itemsets, co-occurring terms and multiple grams, for building up a representation with these new types of features. Nevertheless, the challenging issue is how to effectively deal with the large amount of discovered patterns.

III. PROPOSED METHOD

In this section, we describe the approach of enriching process of rule mining for XML data using Eclat. The step followed by our proposed system is described as shown in Fig. 1.

Step 1: In this step, an XML file data is been extracted using java XQuery process and then it is been saved in the Database.

Step2: This is the step where all the XML data stored in DB are preprocessing by the following four main activities: Sentence Segmentation, Tokenization, Removing Stop Word, and Word Stemming. Sentence segmentation is boundary detection and separating source text into sentence. Tokenization is separating the input text into individual words. Next, Removing Stop Words, stop words are the words which appear frequently in the text but provide less meaning in identifying the important content of the document such as ‘a’, ‘an’, ‘the’, etc.. The last step for preprocessing is Word Stemming; Word stemming is the process of removing prefixes and suffixes of each word.

Step 3: Term Weight

The frequency of term occurrences within a document has often been used for calculating the importance of sentence. The score of a sentence can be calculated as the sum of the score of words in the sentence. The score of important score \( w_i \) of word \( i \) can be calculated by the traditional tf.idf method as follows [16](Inverse document frequency).

\[
W_i = \text{tf}_i \times \text{idf}_i = \text{tf}_i \times \log (N / n_i)
\]

Where \( tf_i \) is the term frequency of word \( i \) in the document, \( N \) is the total number of documents, and \( n_i \) is number of documents in which word \( i \) occurs.
Step 4: In order to summarize each of documents in an IR result, we use Shannon’s term weighting based on formation Gain Ratio (IGR).

This method extracts the similarity structure among a set of documents through a hierarchical clustering, then gives higher weights to words that contribute to forming the structure. Important words are calculated based on IGR as follows:

\[
\text{IGR}(C) = \sum (|C_i| / |C|) \log (|C_i| / |C|)
\]

Where \(C_i\) is the frequency of the word \(w\) in Cluster \(C\).

Step 5: Then after fetching the important words from all the documents our system will perform association rule using Apriori Algorithm with the step stated below.

Let \(T\) be the training data with \(n\) attributes \(A_1, A_2, ..., A_n\) and \(C\) is a list of class labels. A particular value for attribute \(A_i\) will be denoted \(a_i\), and the class labels of \(C\) are denoted \(c_j\).

Definition 1: An item is defined by the association of an attribute and its value \((A_i, a_i)\), or a combination of between 1 and \(n\) different attributes values, e.g. \(< (A_1, a_1)>, < (A_1, a_1), (A_2, a_2)>, < (A_1, a_1), (A_2, a_2), (A_2, a_2)>, < (A_3, a_3)>, ... \) etc.

Definition 2: A rule \(r\) for multi-label classification is represented in the form:

\((A_{i1}, a_{i1}) \land (A_{i2}, a_{i2}) \ldots \land (A_{im}, a_{im}) \rightarrow c_{i1}, ..., c_{im}\)

where the condition of the rule is an item and the consequent is a list of ranked class labels.

Definition 3: The actual occurrence (ActOccr) of a rule \(r\) in \(T\) is the number of cases in \(T\) that match \(r\)'s condition.

Definition 4: The support count (SuppCount) of \(r\) is the number of cases in \(T\) that matches \(r\)'s condition, and belong to a class \(c_i\). When the item is associated with multiple labels, there should be a different SuppCount for each label.

Definition 5: A rule \(r\) passes the minimum support threshold (MinSupp) if for \(r\), the \(\text{SuppCount}(r) / |T| \geq \text{MinSupp}\), where \(|T|\) is the number of instances in \(T\).

Definition 6: A rule \(r\) passes the minimum confidence threshold (MinConf) if \(\text{SuppCount}(r) / \text{ActOccr}(r) \geq \text{MinConf}\).

Definition 7: Any item in \(T\) that passes the MinSupp is said to be a frequent item.

Step 6: In the final step our system we perform vertical frequent pattern mining for eclat using below algorithm:

---

**Algorithm 1 Eclat Algorithm**

**Input:** Alphabet A with ordering \(\leq\) multiset \(T \subseteq P(A)\) of sets of Items, Minimum support value \(\text{minsup} \in \mathbb{N}\).

**Output:** Set \(F\) of frequent Itemsets and their support counts.

\[F:=\{(\emptyset, |T|)\}.
\]

\[C\emptyset:= \{(x,T\{\{x\}\})|x \in A\}.
\]

\[C'\emptyset:= \text{freq}(C_\emptyset):= \{(x,T_x)|(x,T_x) \in C_\emptyset, |T_x| \geq \text{minsup}\}
\]

\[F:=\{\emptyset\}.
\]

Add frequent supersets \((\emptyset, C'\emptyset)\).

**function** add Frequent Supersets():

**Input:** frequent Itemsets \(p \in P(A)\) called prefix, incidence matrix \(C\) of frequent 1-item-extensions of \(p\).

**Output:** add all frequent extensions of \(p\) to global variable \(F\).
for \((x, T_x) \in C\) do
\[ q := p \cup \{X\}. \]
\[ C_q := \{(y, T_y) \mid (y, T_y) \in C, y > x\}. \]
\[ C'q := \text{freq}(C_q) := \{(y, T_y) \mid (y, T_y) \in C_q, |T_y| \geq \text{minsup}\} \]

If \(C'q \neq \emptyset\) then
Add frequent supersets \((q, C'q)\).

End if
\[ F := F \cup \{(q, |T_x|)\} \]
End for

IV. RESULTS AND DISCUSSIONS

To show the effectiveness of the proposed system, some experiments are reported. Selecting a suitable dataset is a critical and important step in designing of any association rule based system.

For experiment, we used Reuters XML Dataset and then extract the important XML tags like Document ID, body and label names and then store in the Database. We used Java as the implementation language and MySQL as Database.

Time Comparison of Apriori and Eclat algorithms

This shows that we are effectively applied the techniques mining association rules for huge datasets also.

V. CONCLUSION AND FUTURE WORK

This paper has presented a na"ive multi label-oriented framework in vertical association mining technique using Eclat. In our approach we are taken Reuters XML dataset and then extract the required data from XML very efficiently to store in database. In Apriori a large amount of candidate are produced so require large memory space due to this more time is wasted in producing candidates. Where as in Eclat it works on vertical pattern so it produces less amount of candidates so it takes less amount of time, this can be seen in figure no2. And we also proved in our system that éclat produces more quality rules compared to Apriori more efficiently.

For the future enhancement of the system this proposed work can be taken to a another level of selective pattern mining technique to reduce much time. This can be achieving by considering the important features of the documents like noun, verb and numerical data to exponentially reduce the time taken for the rule mining.

REFERENCES


