International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Pattern discovery on the World Wide Web by Using Web Mining Methods: A Review

Anu Bala[1], Amardeep Singh[2]

[1]Department of Computer Engineering, Master of Engineering, UCOE, Punjabi University Patiala

[2]Professor, Department of Computer Engineering, UCOE, Punjabi University Patiala

Abstract: ***Web mining is a very popular technique that helps users find useful and required information from a huge amount of digital text documents available on the Web, knowledge or databases. Instead of using the simplest keyword-based approach which is normally used in this field for data extraction, the pattern based model in which we are able to find common sequential patterns can be used to perform the same concept of tasks in a more. However, how to effectively use these discovered patterns is still a big challenge. In this study, we propose different approaches based on the use of pattern deploying strategies.***

Keywords: Text mining, D-Pattern (DP), Sequential pattern (SP), Term frequency–inverse document frequency (TFIDF), Frequent pattern (FP), Pattern Deploy method (PDM).

1. Introduction

In the past years there are some techniques for data mining which helps the person to perform different tasks relating to education. These techniques includes association rule mining, sequential pattern mining, frequent item set mining, maximum pattern mining, and closed pattern mining. There is a huge advancement in the fields of digital data in such a years, knowledge discovery and data mining have work together a great deal of attention with an need for meaningful data into useful information and knowledge. [1]

Web mining is the technique that helps users finds useful information of a large number of digital text documents Web or databases. Therefore, it is crucial that good model for text mining to retrieve information meets the needs of users in a relatively efficient time. Many text mining methods have been developed to achieve the objective of recovering useful information for users. Most of them adopt Keyword approach, while others choose the syntagmatic technique for the construction of a representative of a set of text documents [2].

Web mining is used to finding relevant & interesting information from huge database. Web mining is to exploit information contained in textual documents in various ways including discovery of patterns, association among entities, etc. In this paper, I want to compare different systems so that it is possible to focus on the development of a knowledge discovery model to effectively use and update the discovered patterns and apply it to the field of web mining. The advantages of term based methods include efficient computational performance as well as theories for term weighting.

2. Deploying Approaches for Pattern Refinement in Text Mining

Data pre-processing is applied before the documents can be interpreted by the deploying methods. The fields we chose in a document are title and text. The content in title is viewed as a paragraph as the one in text which consists of paragraphs. For dimensionality reduction, stop words removal is applied and Porter algorithm [2] is selected for suffix stripping. Terms with frequency equalling to one are discarded. To evaluate experimental results, the author had used several standard measures such as the precision of first k returned documents (top-K). [3]

For evaluating the proposed algorithm, it is required to apply the pattern deploy method, PDR, to the information filtering task. For each topic, the system extracts the concept and aims to filter out the non-relevant incoming documents according to the user profiles. Concept generating is based on the Rocchio algorithm which is used to build the profile for representing a concept of a topic which consists of a set of relevant and irrelevant documents. The Centroid ~C of a topic can be generated by using the following Rocchio equation.

$$\vec{c} = \alpha \frac{1}{|D^+|} \sum_{\vec{d} \in D^+} \frac{\vec{d}}{\|\vec{d}\|} - \beta \frac{1}{|D^-|} \sum_{\vec{d} \in D^-} \frac{\vec{d}}{\|\vec{d}\|} \quad (1)$$

Where “alpha” and “beta” are empirical parameters; D+ and D− are the set of relevant documents and the set of irrelevant documents respectively; “d” denotes a document. [4]

3. D-Pattern Mining Algorithm

In this module to improve the efficiency of the pattern taxonomy mining, an algorithm, D-Pattern Mining is used to find all closed sequential patterns, which used the well-known. A priority property is used in order to reduce the searching space. Algorithm is used to describe the training process of finding the set of d-patterns. For every positive document, the D-Pattern Mining algorithm is first called giving rise to a set of closed sequential patterns SP.[5] The main focus of this project is the deploying process, which consists of the d-pattern discovery and term support evaluation. In Algorithm all discovered patterns in a positive document are composed into a “d-pattern” giving rise to a set of d-patterns DP. Thereafter, term supports are calculated

based on the normal forms for all terms in “d-patterns”. Let “m” be the number of terms in “T”, “n” be the number of positive documents in a training set, “K” be the average number of discovered patterns in a positive document, and “k” be the average number of terms in a discovered pattern.

The technique will be useful to reduce the side effects of noisy patterns because of the low-frequency problem. This technique is called inner pattern evolution here, because it only changes a pattern’s term supports within the pattern. A threshold is usually used to classify documents into relevant or irrelevant categories. In order to reduce the noise, we need to track which d-patterns have been used to give rise to such an error. We call these patterns offenders of d. An offender of d is a d-pattern that has at least one term in n. There are two types of offenders, a complete conflict offender which is a subset of d; and a partial conflict offender which contains part of terms of d. The basic idea of updating patterns is explained as follows: complete conflict offenders are removed from d-patterns first. For partial conflict offenders, their term supports are reshuffled in order to reduce the effects of noise documents. The main process of inner pattern evolution is implemented by the algorithm IP Evolving. The inputs of this algorithm are a set of d-patterns DP, a training set D. The output is a composed of d-pattern. The algorithm is used to estimate the threshold for finding the noise negative documents. It revises term supports by using all noise negative documents. It also finds noise documents and the corresponding offenders. Shuffling is used to update NDP according to noise documents. The task of algorithm Shuffling is to tune the support distribution of terms within a d-pattern. A different strategy is dedicated in this algorithm for each type of offender. In the algorithm Shuffling, complete conflict offenders are removed since all elements within the d-patterns are held by the negative documents indicating that they can be discarded for preventing interference from these possible “noises.”.[6]

4. Features Selection Method

In this method, documents are considered as an input and the features for the set of documents are collected. Features are selected based on the TFIDF method. Information retrieval has been developed based on many mature techniques which demonstrate the terms which are important features in the text documents. However, many terms with larger weights FP-tree generation algorithm works given an input database *D* that has five item sets (*t*1, *t*2 , *t*5) with items *I* = {1, 2, 3, 4, 5, 6}, First, it scans *D* to find the number of times that each item occurs in the various item sets and uses this information to build the Item Support Table. This table consists of a set of (item-ID, support) pairs. For example, item 1 occurs twice in the item set database (in item sets *t*1 and *t*5); therefore its support is 2⁄5 = 0.4. Then, any items whose support is smaller than the minimum support are eliminated and the remaining items are sorted in non-increasing order according to their support. The resulting ordering is stored in an array called the Node-Link header table or NL for short. Finally, the FP-tree is generated by reading the item sets from the database and inserting them one-by-one in the tree. Initially the FP-tree has only a root node called the null node. Each (non-root) node of the FP-tree contains three fields. [9] The first field corresponds to the item-ID of the item for which are general terms because they can be frequently used in both relevant and irrelevant information. The features selection approach is used to improve the accuracy of evaluating term weights because the discovered patterns are more specific than whole documents. In order to reduce the irrelevant features, much dimensionality reduction conducted by the use of feature selection techniques. Patterns can be structured into taxonomy by using the subset relation. Smaller patterns in the taxonomy are usually more general because they could be used frequently in both positive and negative documents and larger patterns are usually more specific since they may be used only in positive documents.[7] The semantic information will be used in the pattern taxonomy to improve the performance of using closed patterns in text mining.

5. FP-growth Algorithm

The key idea behind the FP-growth algorithm is to use a data structure called FP-tree to compactly store the database so that it can fit in the main memory. As a result, any subsequent operations that are required to find the frequent item set patterns can be performed quickly, without accessing the disks.

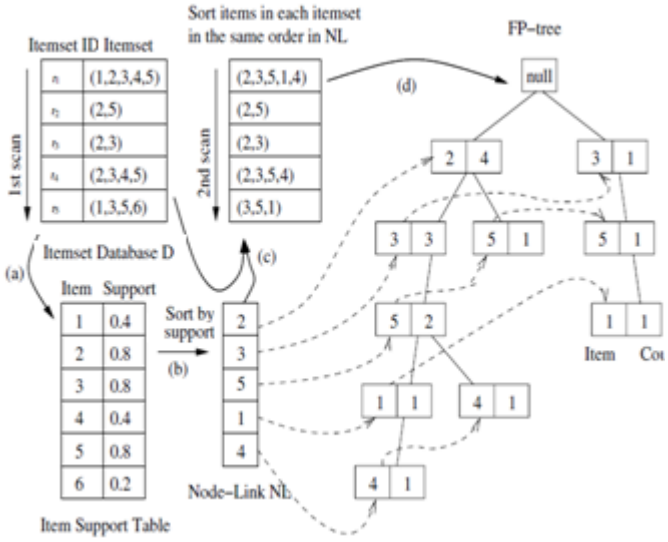

Figure 1: Tree of FP growth algorithm

The FP-tree itself can be built efficiently by requiring only two passes over the input database. Figure 3 shows how the

this node was created, the second field represents a counter that is set to one the moment a node is created, and the third field is used to form a link-list of all the nodes corresponding to the same item. Note that the FP-tree of Figure 1 uses a two-element array to represent each node in which the first element corresponds to the item-ID and the second element corresponds to the counter [8].

6. Comparison

Table 1: Comparison of different techniques studied.

S. no.	*Technique*	*Design*	*Design Complexity level*	*Efficiency level*	*Advantages*
1	Pattern Refinement in Text Mining	Pattern deploy method (PDM)	Simple	Normal	Easy to use

2	D-Pattern Mining	Finding closed sequential patterns	High	High	Reduction in the side effects of noisy patterns
3	Features Selection Method	Term frequency–inverse document frequency	Normal	High	Accuracy
4	FP-growth Algorithm	Storing the database	High	Normal	Efficiency

Here, we show the comparison of different techniques that we have studied in terms of efficiency and design complexity. From this, we can conclude that D pattern mining and FP Growth algorithm have high complexity where as efficiency is higher for D-pattern mining and feature selection method.

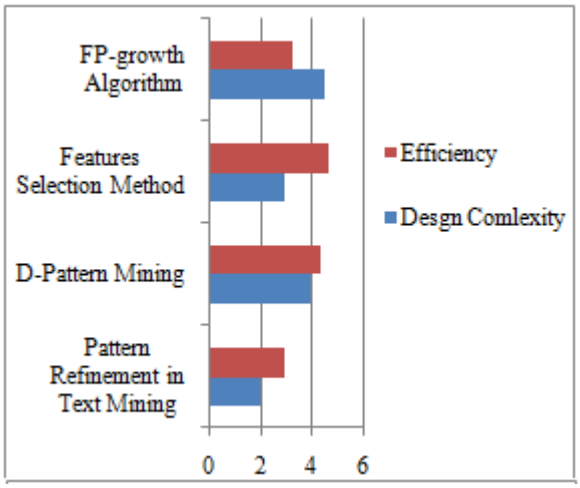

Figure 2: Comparison of different techniques.

7. Conclusion

Different web data mining techniques had been studied in this paper. These techniques include association rule mining, frequent item set mining, sequential pattern mining, maximum pattern mining, and closed pattern mining. However, using other discovered knowledge, patterns in the field of web mining is difficult and ineffective. The reason is that some useful long patterns with high specificity lack in support. We had observed that not all frequent short patterns are useful. So misinterpretations of patterns derived from web mining techniques lead to in effective performance. In this review work, a few effective pattern discovery techniques have been proposed to overcome the low frequency.

8. Future Scope

The work proposed in this project is quiet significant because we will concentrate on the different techniques of web content mining methods which can be used for pattern discovery. To use the advantages of web mining, pattern taxonomy models have been used for tracing closed sequential patterns in text classification. These pattern mining based approaches have shown a certain extent improvement on the effectiveness. However, fewer significant improvements are made compared with term-based methods. There are many challenging issues for introducing pattern mining techniques to find relevance features in both positive and negative documents.

The pattern discovery is also significant because using these discovered knowledge (or patterns) in the field of web mining is difficult but effective. The reason is that some useful long patterns with high specificity lack in support due to the low-frequency problem if some methods are used. So we can say that all frequent short patterns are useful for the overall success of these methods. Hence, misinterpretations of patterns derived from data mining techniques lead to the effective performance. In this research work, we are going to implement effective pattern discovery technique that can overcome the low-frequency and misinterpretation problems for web mining. The proposed technique is significant as it will be based on two processes which are pattern deploying and pattern evolving and these will refine the discovered patterns in text documents on World Wide Web.

References

[1] Rupali Bhaisare , T. Raju Rao ; REVIEW ON TEXT MINING WITH PATTERN DISCOVERY; International Journal of Innovative Research in Computer and Communication Engineering, 2010.
[2] Sheng-Tang Wu Yuefeng Li Yue Xu; Deploying Approaches for Pattern Refinement in Text Mining; School of Software Engineering and Data Communications, 2009.
[3] Ning Zhong, Yuefeng Li, and Sheng-Tang Wu; Effective Pattern Discovery for Text Mining; IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 24, NO. 1, JANUARY 2012;
[4] Mrs.K. Mythili and Mrs. K. Yasodha; A Pattern Taxonomy Model with New Pattern Discovery Model for Text Mining; International Journal of Science and Applied Information Technology Volume 1, No.3, July – August 2012;
[5] Masakazu Seno and George Karypis;Finding Frequent Patterns Using Length-Decreasing Support Constraints; Department of Computer Science and Engineering University of Minnesota, Minneapolis, MN 55455 Technical Report 03–0004;
[6] Yuefeng Li Abdulmohsen Algarni Ning Zhong; Mining Positive and Negative Patterns for Relevance Feature Discovery, 2012.
[7] M.Suganthy , K.Rupika ,J. Sharmiliya Fransuva; TEXT MINING FOR PATTERN IDENTIFICATION; International Journal of Futuristic Science Engineering and Technology Vol 1 Issue 3 March 2013 ISSN 2320 – 4486;
[8] Jian Pei Jiawei Han Behzad Mortazavi-Asl Helen Pinto; Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth; Intelligent Database Systems Research Lab. , 2010
[9] N. Phanikiran, Vuppu Shankar, Information Mining Using Evolving Patterns; International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 9, September 2013.