Middle-East Journal of Scientific Research
24 (Recent Innovations in Engineering, Technology, Management & Applications): 167-170, 2016
ISSN 1990-9233; © IDOSI Publications, 2016
DOI: 10.5829/idosi.mejsr.2016.24.RIETMA127

Effective Pattern Discovery for Text Mining Using Parallel Processing

R. Malathi, U. Jayalakshmi and S. Vijayakumar

Department of MCA, Priyadarshini Engineering College, Vaniyambadi, India

Abstract: Parallel processing can be applied in Data mining to speed up the processing of huge amount of data. Many data mining techniques have been proposed for mining useful patterns in text documents. However, how to effectively use and update discovered patterns is still an open research issue, especially in the domain of text mining. Applying parallel processing in the text mining will speed up the useful pattern discovery. This paper presents an innovative and effective pattern discovery technique using parallel processing which includes the processes of pattern deploying and pattern evolving, to improve the effectiveness of using and updating discovered patterns for finding relevant and interesting information.

Key words: Text mining · Parallel processing · Pattern mining · Pattern evolving

INTRODUCTION

Parallel processing [1] is the process of working with more than one processors simultaneously. Due to the rapid growth of digital data made available in recent years, knowledge discovery and data mining have attracted a great deal of attention with an imminent need for turning such data into useful information and knowledge within short period of time. Many applications, such as market analysis and business management, can benefit by the use of the information and knowledge extracted from a large amount of data using parallel processing concept. Knowledge discovery can be viewed as the process of nontrivial extraction of information from large databases, information that is implicitly presented in the data, previously unknown and potentially useful for users. Data mining is therefore an essential step in the process of knowledge discovery in databases.

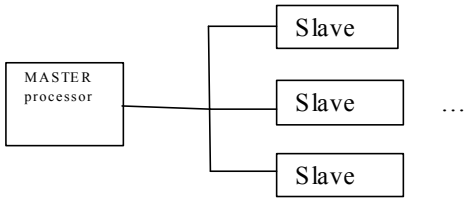

Fig. 1: Parallel processing concept.

Pattern Mining: Pattern mining has been extensively studied in data mining communities for many years. A variety of efficient algorithms such as Apriori-like algorithms [2, 3, 4], PrefixSpan [5, 6], FP-tree [7, 8], SPADE, SLPMiner [9] and GST [10] 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. In the field of text mining, pattern mining techniques can be used to find various text patterns, such as sequential patterns, frequent item sets, 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.

For the challenging issue, closed sequential patterns have been used for text mining in which proposed that the concept of closed patterns in text mining was useful and had the potential for improving the performance of text mining. Pattern taxonomy model was also developed in and to improve the effectiveness by effectively using closed patterns in text mining. In addition, a two-stage model that used both term-based methods and pattern based methods was introduced in to significantly improve the performance of information filtering.

Table 1: A Set of Paragraphs

PARAGRAPH	TERMS
Dp_1	t_1 t_2
Dp_2	t_3 t_4 t_6
Dp_3	t_3 t_4 t_5 t_6
Dp_4	t_3 t_4 t_5 t_6
Dp_5	t_1 t_2 t_6 t_7
Dp_6	t_1 t_2 t_6 t_7

Corresponding Author: R. Malathi, Department of MCA, Priyadarshini Engineering College, Vaniyambadi, India.

Natural language processing (NLP) is a modern computational technology that can help people to understand the meaning of text documents. For a long time, NLP was struggling for dealing with uncertainties in human languages. Recently, a new concept-based model [11, 12] was presented to bridge the gap between NLP and text mining, which analyzed terms on the sentence and document levels. This model included three components. The first component analyzed the semantic structure of sentences; the second component constructed a conceptual ontological graph (COG) to describe the semantic structures; and the last component extracted top concepts based on the first two components to build feature vectors using the standard vector space model. The advantage of the concept-based model is that it can effectively discriminate between non important terms and meaningful terms which describe a sentence meaning. Compared with the above methods, the concept-based model usually relies upon its employed NLP techniques.

Pattern Taxonomy Model: In this paper, we assume that all documents are split into paragraphs. So a given document d yields a set of paragraphs PS (d). Let D be a training set of documents, which consists of a set of positive documents, D^+; and a set of negative documents, D^-. Let T =$\{t_1,t_2. . . t_m\}$ be a set of terms (or keywords) which can be extracted from the set of positive documents, D^+.

Table 2: Frequent Pattern and Covering Sets

Frequent Pattern	*Covering Set*
$\{t_3,t_4,t_6\}$	$\{Dp_2,Dp_3,Dp_4\}$
$\{t_3,t_4\}$	$\{Dp_2,Dp_3,Dp_4\}$
$\{t_3,t_6\}$	$\{Dp_2,Dp_3,Dp_4\}$
$\{t_4,t_6\}$	$\{Dp_2,Dp_3,Dp_4\}$
$\{t_3\}$	$\{Dp_2,Dp_3,Dp_4\}$
$\{t_4\}$	$\{Dp_2,Dp_3,Dp_4\}$
$\{t_1,t_2\}$	$\{Dp_1,Dp_5,Dp_6\}$
$\{t_1\}$	$\{Dp_1,Dp_5,Dp_6\}$
$\{t_2\}$	$\{Dp_1,Dp_5,Dp_6\}$
$\{t_6\}$	$\{Dp_2,Dp_3,Dp_4,Dp_5,Dp_6\}$

Pattern Taxonomy: Patterns can be structured into a taxonomy by using the is-a (or subset) relation. For the example of Table 2.1, where we have illustrated a set of paragraphs of a document and the discovered 10 frequent patterns in Table 2.2 if assuming min_sup = 50%. There are, however, only three closed patterns in this example. They are $<t_3; t_4; t_6>$,$<t_1; t_2>$ and $<t_6>$.

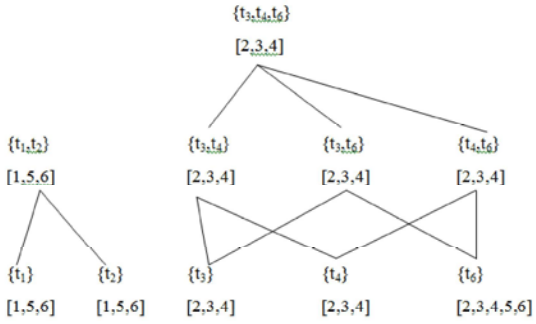

Fig. 2: Pattern taxonomy

Fig. 2 illustrates an example of the pattern taxonomy for the frequent patterns in Table 2.2, where the nodes represent frequent patterns and their covering sets; non closed patterns can be pruned; the edges are "is-a" relation. After pruning, some direct "is-a" retaliations may be changed, for example, pattern $\{t_6\}$ would become a direct sub pattern of $\{t_3, t_4, t_6\}$ after pruning non closed patterns.

Smaller patterns in the taxonomy, for example pattern $\{t_6\}$, (Fig. 2.1) are usually more general because they could be used frequently in both positive and negative documents; and larger patterns, for example pattern $\{t_3, t_4, t_6\}$, in the taxonomy are usually more specific since they may be used only in positive documents. The semantic information will be used in the pattern taxonomy to improve the performance of using closed patterns in text mining, which will be further discussed in the next section.

Closed Sequential Patterns: A sequential pattern $s= <t_1;. . . ; t_r>(t_i \in T)$ is an ordered list of terms. A sequence $s_1= <x_1,. . . , x_i>$ is a subsequence of another sequence $s_2= <y_1,. . . , y_j>$, denoted by $s_1 \subseteq s_2$, iff $\exists j_1,.. . , j_y$ such that $1 = j_1 < j_2. . . < j_y= j$ and $x_{1=} y_{j1}, x_2 = y_{j2}, . . . , x_{i=} y_{jy}$. Given $s_1 \subseteq s_2$, we usually say s_1 is a subpattern of s_2 and s_2 is a superpattern of s_1. In the following, we simply say patterns for sequential patterns.

Given a pattern (an ordered termset) X in document d, $^+X^+$ is still used to denote the covering set of X, which includes all paragraphs $ps \in PS(d)$ such that $X \subseteq ps$, i.e.,

$$^+X^+ = \{ps \backslash ps \in PS(d), X \subseteq ps\}.$$

Its absolute support is the number of occurrences of X in PS(d), that is

$$sup_a(X) = \in^+ \lceil X \rceil^+ \in.$$

Its relative support is the fraction of the paragraphs that contain the pattern, that is,

$sup_r(X) = \frac{\lceil X \rceil}{PS(d)}$

A sequential pattern X is called frequent pattern if its relative support (or absolute support) = min_sup, a minimum support. The property of closed patterns $sup_a(X_1) < sup_a(X)$,can be used to define closed sequential patterns. A frequent sequential pattern X is called closed if not 9 any super pattern X1 of X such that $sup_a(X_1) = sup_a(X)$.

Related Work: This frequent pattern mining can be solved using parallel processing by splitting the document by n partition [13].

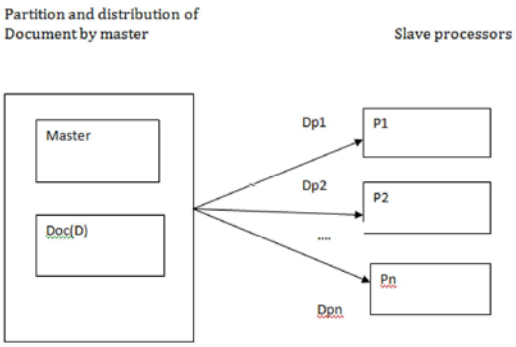

Fig. 3: Master and slave system

The parallel processing system uses n+1 processors [13]. In that one act as the master others act as slaves, in which master partition and distributes the document D^+ to all the slave processors as$\{Dp_1,Dp_2,Dp_3,\ldots..,Dp_n\}$.

The termset X will place in the master based on the termset the slave processors will generate the Boolean bit(bbit) the total count will be send to masters count.

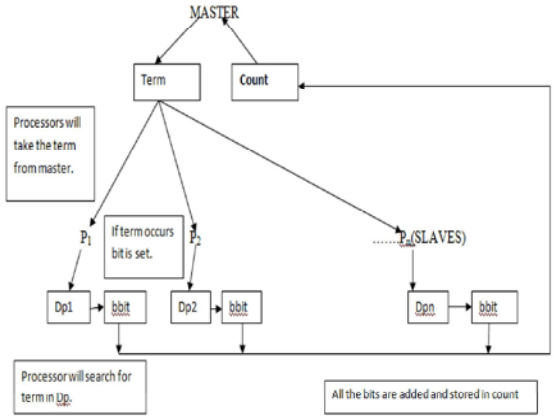

Fig. 4: Count of termset

$PS(d)=\{Dp_1,Dp_2,\ldots,Dp_n\}$.

Absolute support sup_a is the number of occurance of X in PS(d).

Relative support sup_r is fraction of the paragraph that contain the pattern.

Frequent pattern min_sup=50%

$$sup_r(X) = \frac{\lceil X \rceil}{PS(d)} = \frac{|\{Dp1,Dp5,Dp6\}|}{|\{DP1,Dp2,Dp3,Dp4,Dp5,Dp6\}|} = \frac{|3|}{|6|}$$

$sup_r(X)$ =0.5.

Proposed Algorithm:

Input: positive document D^+, minimum support, min_sup.

Output: d-patterns DP and supports of terms.

DP=0;
For each processor P_{i+1} in p do
For each document d in D^+ do
Master processor distributes each paragraph in PS(d) to each processor P_i
Let PS (d) be the set of paragraphs in d
SP=SP mining (PS(d), min_sup);
D^=◎
For each pattern p_i ◎SP do
p={(t,1) | t◎p_i},
d^=d^◎p
End
DP=DP◎{d^},
End
T={t|(t,f) ◎p, p◎DP},
For each term t◎T do
Support(t)=0
End
For each d-pattern p◎DP do
For each (t,w) ◎β(p) do
Support(t)=support(t)+w,
End
End
End
End

CONCLUSION

In this rearch work, an effective pattern discovery technique using parallel processing has been

proposed to improve the effectiveness of using and updating discovered patterns for finding relevant and interesting information within short period of time. The ratio of reduced time can be calculated as

$$T_p = \frac{\text{Time taken by the Sequential Algorithm}}{\text{Time taken by the Parallel Algorithm}}$$

where T_p is the ratio of time taken by the Parallel Algorithm.

REFERENCES

1. https://en.wikipedia.org/wiki/Parallel_computing.
2. Agrawal, R. and R. Srikant, 1994. Fast Algorithms for Mining Association Rules in Large Databases, Proc. 20th Int'l Conf. Very Large Data Bases (VLDB '94): 478-499.
3. Park, J.S., M.S. Chen and P.S. Yu, 1995. An Effective Hash-Based Algorithm for Mining Association Rules, Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '95), pp: 175-186.
4. Srikant, R. and R. Agrawal, 1995. Mining Generalized Association Rules, Proc. 21th Int'l Conf. Very Large Data Bases (VLDB '95), pp: 407-419.
5. Pei, J., J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal and M. Hsu, 2001. Prefixspan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth, Proc. 17th Int'l Conf. Data Eng. (ICDE '01), pp: 215-224.
6. Yan, X., J. Han and R. Afshar, 2003. Clospan: Mining Closed Sequential Patterns in Large Datasets, Proc. SIAM Int'l Conf. Data Mining (SDM '03), pp: 166-177.
7. Han, J. and K.C.C. Chang, 2002. Data Mining for Web Intelligence, Computer, 35(11): 64-70.
8. Han, J., J. Pei and Y. Yin, 2000. Mining Frequent Patterns without Candidate Generation, Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '00), pp: 1-12.
9. Seno, M. and G. Karypis, 2002. Slpminer: An Algorithm for Finding Frequent Sequential Patterns Using Length-Decreasing Support Constraint, Proc. IEEE Second Int'l Conf. Data Mining (ICDM '02): 418-425.
10. Huang, Y. and S. Lin, 2003. Mining Sequential Patterns Using Graph Search Techniques, Proc. 27th Ann. Int'l Computer Software and Applications Conf., pp: 4-9.
11. Shehata, S., F. Karray and M. Kamel, 2006. Enhancing Text Clustering Using Concept-Based Mining Model, Proc. IEEE Sixth Int'l Conf. Data Mining (ICDM '06): pp: 1043-1048.
12. Shehata, S., F. Karray and M. Kamel, 2007. A Concept-Based Model for Enhancing Text Categorization, Proc. 13th Int'l Conf. Knowledge Discovery and Data Mining (KDD '07): 629-637.
13. Rajaraman, V. and C. Siva Ram Murthy, XXXX. Parallel Computers: Architecture and Programming ".