Operational Pattern Revealing Technique in Text Mining

Bhushan Inje
Department of Computer Engineering
R C Patel Institute of Technology, Shirpur
Dist. Dhule, Maharashtra, India
bhushan.inje@gmail.com

Ujawla Patil
Department of Computer Engineering
R C Patel Institute of Technology, Shirpur
Dist. Dhule, Maharashtra, India
patil_ujwala2003@rediffmail.com

***Abstract*— In this digital era most of the information is made available in digital form. For many years, people have held the hypothesis that using phrases for a representation of document and topic should perform better than terms. In this paper we are examine and investigate this fact with considering several state of art datamining methods that gives satisfactory results to improve the effectiveness of the pattern. Here we implementing pattern detection method to solve problem of term-based methods and improved result which helpful in information retrieval systems. Our proposal is also evaluated for several well distinguish domain, offering in all cases, reliable taxonomies considering precision and recall along with F-measure. For the experiment, we use *Reuters (RCV1)* dataset and the results show that we improve the discovering pattern as compared to previous text mining methods. The results of the experiment setup show that the keyword-based methods not give better performance than pattern-based method. The results also indicate that removal of meaningless patterns not only reduces the cost of computation but also improves the effectiveness of the system.**

Keywords—Data mining; Sequential pattern mining; Pattern discovery; Text mining; Phrase based method; Information Retrieval.

I. INTRODUCTION

We observe that most of datamining techniques have been to perform different knowledge tasks. Because 85% of business information lives in the form of text [2]. Text mining is a variation on a field called data mining [2] that tries to find interesting patterns from large databases. Text mining, also known as Intelligent Text Analysis, Text Data Mining or Knowledge-Discovery in Text (KDT), refers generally to the process of extracting interesting and non-trivial information and knowledge from unstructured text [1].

Text mining is the technique that helps users finds useful information from a large amount of digital text data. Many text-mining methods have been developed in order to achieve the goal of retrieving useful information for users [7, 8, 9]. Most text mining methods use the keyword-based approaches, whereas others choose the phrase technique to construct a text representation for a set of documents. It is believed that the phrase-based approaches should perform better than the keyword-based ones as it is considered that more information is carried by a phrase than by a single term. Based on this hypothesis, Lewis [10] on ducted several experiments using phrasal indexing language on a text categorization task. Ironically, the results showed that the phrase-based indexing language was not superior to the word-based one. Although phrases carry less ambiguous and more concise meanings than individual words, the likely reasons for the depressing performance from the use of phrases are: (1) phrases have inferior statistical properties to words, (2) they have a low frequency of occurrence, and (3) there are a large number of redundant and noisy phrases among them [11].

The remainder of this paper is organized as follows. Section 2 gives a detailed overview of the PTM model and section 3 highlights the related problems and definitions with the PTM model. How to use discovered pattern presented in Section 4, in Section 5 deployed pattern evolution and Section 6 describes the experimental findings and discusses the results.

II. MOTIVATION AND RELATED WORK

Many data mining methods has been proposed to represent text in past. Sequential mining is also studied in that context since the first research work in [15]. Then Apriori like algorithm takes initiative for solving problem face in large dataset [16] but this algorithms perform well in case of short frequency of term. To solve disadvantage of Apriori like algorithm, number of new algorithms has been proposed i.e. PrefixSpan [16], in order to improve efficiency of patterns along with many other algorithms are functional proved that they have improve the efficiency, each of these algorithms work on different method of discovering frequent sequential pattern. However, the searching for useful and interesting patterns was still an open research problem.

The process of knowledge discovery may consist following steps: Data selection, Data preprocessing, Data transformation, Pattern discovery and Pattern evaluation [18]. In Data selection generating a target dataset and selecting a dataset or a subset of large data sources where discovery is to be performed. Then Pre-processing process involves data cleaning and noise removing. It also includes collecting required information from selected data fields, providing appropriate strategies for dealing with missing data and accounting for redundant data. For the case of web pages then transformation, preprocessed data needs to be transformed into a predefined format, depending on the data-mining task. This process needs to select an adequate type of features to represent data.

978-1-4799-2526-1/14/$31.00 ©2014 IEEE

In the presence of these setbacks, sequential patterns used in data mining community have turned out to be a promising alternative to phrases [1,19] because patterns enjoy good statistical properties like terms. To overcome the disadvantages of phrase-based approaches, pattern-based approaches (or Pattern Taxonomy Models (PTM) [1,19]) have been proposed for IF from within the data mining community. These pattern based approaches have shown encouraging improvements on effectiveness, but at the expense of computational efficiency. In regard to the aforementioned problem of redundancy and noise, PTM adopts the concept of closed patterns, or pruned non-closed patterns [1]. However, it is a still challenging issue for PTM to deal with low frequency patterns because the measures used from data mining (e.g., "support" and "confidences") to learn the profile turn out be not suitable in the filtering stage. By way of illustration, given a specified topic, a highly frequent pattern (normally a short pattern with large support) is usually a general pattern, or a specific pattern of low frequency. This parallels the situation in term indexing where words of high frequency (stop words) or very low frequency (highly information bearing uncommon words) are not considered useful.

III. Problem formulation

Due to limitation found [13] in the keyword-based concept used in the traditional document representation model, the pattern based model [12] containing frequent sequential patterns is used to perform the same concept of task. This section will define the basic problem of mining sequential pattern in text documents.

In this paper, we assume that all documents are split into paragraphs. So a given documents *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 $T= \{t_1, t_2,...,t_k \}$ be a set of terms or keywords which can be extracted from the set of positive documents D^+.

Definition 1 {sub sequence}

A sequence $S= \{s_1, s_2 ,...,s_n \}$ $(s_i \in T)$ is an ordered list of terms. A sequence $\alpha= \{a_1, a_2, ...,a_n\}$ is a sub-sequence of another sequence $\beta= \{b_1, b_2,...,b_m \}$, denoted by $\alpha \subseteq \beta$, if there exist integers $1\leq i_1< i_2< \dots < i_n \leq m$, such that $a_1= b_{i1}$, $a_2 =b_{i2}$,..., $a_n =b_{in}$. The sequence α is a proper sub-sequence of β if but $\alpha \neq \beta$ denoted by $\alpha \subset \beta$. For instance, sequence <A, C > is a sub-sequence of sequences <A, B, C >. However <B, A> is not a sub-sequence of <A, B, C> since the order of terms is considered. In addition, we also can say sequence < A, B, C > is a super-sequence of <A, C>. The problem of mining sequential patterns is to find the complete set of sub-sequences from a set of sequences whose support is greater than a user predefined threshold*;* *min_sup* Table 1 shows sequence of the ordered list words.

Definition 2 { Frequent Sequential Pattern }

A sequential pattern *P* is called frequent sequential pattern if $supp_r(P)$ is greater than or equal to a minimum support (*min_sup* for short) ξ.

For example, let *min_sup* be 0.75 for mining frequent sequential patterns from a sample document in Table 1. we can obtain four frequent sequential patterns: $(t_2,t_3),(t_1)$, (t_2) and (t_3) since their relative supports arc not less than ξ .

TABLE 1. Paragraph containing sequence of ordered list words

Transaction	Sequences
1	$S_1=(t_1,t_2,t_3,t_4)$
2	$S_2=(t_2,t_4,t_5,t_3)$
3	$S_3=(t_3,t_6,t_1)$
4	$S_4=(t_5,t_1,t_2,t_7,t_3)$

Definition 3 (n Closed Sequential Pattern)

A frequent sequential pattern *P* is a closed sequential pattern if there exist no frequent sequential patterns *P'* such that $P \subset P'$ and $supp_a(P) = supp_a(P')$. The relation $\subset$ represents the strict part of the subsequence relation.

In SPMining [12] algorithm is use to pruning patterns for remove redundant pattern or ambiguous reference in the pattern. As mentioned before, the algorithm 1 SPMining is designed for discovering both *closed* and *non-closed* sequential patterns from a set of documents. The execution of the first line in the algorithm is the key for the *closed* patterns finding and removal of the other patterns. Moreover, it can be easily adjusted to find the *non-closed* sequential patterns by skipping the first line in the algorithm. For the previous document example in Table 1. After inputting this document into the algorithm and setting the minimum support to be (0.5), a list of all the *closed* or *non-closed* sequential patterns can be returned and their results are shown in Table 2.

IV. USING DISCOVERED PATTERNS

The algorithm SPMining uses the sequential data mining technique with a pruning scheme to find meaningful patterns from text documents. The next issue is then how to use these discovered patterns. There are various ways to utilize discovered patterns by using a weighting function to assign a value for each pattern according to its frequency. One strategy has been implemented and evaluated in [12], which proposed a pattern-mining method that treated each found sequential pattern as a whole item without breaking it into a set of individual terms, and its result found that using confidence as the pattern measure outperformed the use of support.

$$\mathrm{W}(p) = \frac{|\{\mathrm{da} \mid \mathrm{da}\} \in \mathrm{D}^+ \ , \mathrm{p} \subseteq da|}{|\{\mathrm{da} \mid \mathrm{da}\} \in \mathrm{D} \ , \mathrm{p} \subseteq da|} \quad (1)$$

Where *D* is the training set of documents and *D*+ indicates the set of positive documents in *D*.

Two problems arise when using the above-mentioned weighting function. One is the low pattern frequency problem, which is mainly because it is hard to match patterns in documents when the length of the pattern is long. The other problem is that those patterns, which are specific to a topic, may gain a lower score than those for general patterns. In other words, the information carried by the specific patterns cannot be estimated by the weighting function. That can be overcome with the following algorithm 1. The inputs of the algorithm PDM are a set of positive documents and a pre-specified minimum support. In line 4 of this algorithm, a set of sequential patterns is discovered by calling the algorithm SPMining for each document. So far, only positive documents are considered and used in this approach. The use of information from negative documents is another issue with reference to pattern evolution.

V. DEPLOYED PATTERN EVOLUTION

The algorithm DPEvolving (Algorithm 3) implements the evolution of deployed patterns. The inputs of this algorithm are a list of deployed patterns Ω, a list of positive and negative documents. D and D^- the output is a set of term weight pairs which can be used directly in the testing phase. Line 2 in DPEvolving is used to estimate the threshold for finding the interesting negative documents. Line 3 to 5 is the process of discovering the offenders of negative documents.

The major task of this algorithm is performed and completed in this phase. The involved patterns are evaluated before being deployed into a hypothesis space. The procedure is stated from line 7 to line 13 in the algorithm.

VI. EXPRIMENTAL SETUP AND RESULTS

Here we focused on experimental setup of our alternating approaches to the pattern taxonomy model to Enhanced performance of pattern deploying model (EPDM). To implement the method three aspects are discussed including experimental Datasets, Performance measures and Evaluation procedures. The latest version of Reuter's document collection is chosen among several versions as our benchmark dataset. Here we get RCV1 cd's from NIST, they made available data set for research purposes. Most of the standard performance measures in textmining (i.e. precision, recall, breakeven point, F_β Measures) are used for evaluating the experimental performance. The EPDM model comprises the methods including pattern discovery approaches (i.e. *Prob* and *tfidf*) pattern deploying methods and pattern evolution strategies. The process of executing EPDM consists of two major phases, concept learning and document evaluation. In the former phase, one of the proposed enhanced pattern discovery approaches is adopted to learn the concept (i.e. user profile) of documents in the training set, and then the various combinations of pattern deploying and evolving methods are taken in the latter phase to evaluate documents in the test set. Text preprocessing for each document is applied before both of the learning and evaluating phases. Term stemming and stop word removal techniques are also used in this stage for document indexing.

To evaluate the performance of EPDM, we implement EPDM for the task of information filtering (IF) in our experiments. By conducting IF tasks, we can examine the ability of the proposed pattern discovery approaches and test the effectiveness of refinement methods for discovered patterns. The experimental results are compared with other well known IF-related methods including Term Frequency Inverse Document Frequency (*tfidf*) method [21]. Probabilistic method (*Prob.)* [5]. We also compare the results from EPDM to those from data mining-based methods, such as frequent itemset mining, sequential pattern mining and closed pattern mining methods.

To evaluate the performance of EPDM, we implement EPDM for the task of information filtering (IF) in our experiments. By conducting IF tasks, we can examine the ability of the proposed pattern discovery approaches and test the effectiveness of refinement methods for discovered patterns. The experimental results are compared with other well known IF-related methods including Term Frequency Inverse Document Frequency (*tfidf*) method [21]. Probabilistic method (*Prob.)* [5]. We also compare the results from EPDM to those from data mining-based methods, such as frequent itemset mining, sequential pattern mining and closed pattern mining methods.

Algorithm 1. SPMining (PL, min_sup)
Input: A set of *nTerms* frequent sequential patterns. *PL*;
Minimum support (*min_sup)*.
Output: A set of frequent sequential patterns (*SP)*.
Method:

```
1. $SP \leftarrow SP - \{\mathrm{Pa} \in \mathrm{SP} \mid \exists\, \mathrm{Pa} \in \mathrm{PL}\}$ such that
   $\mathrm{len(Pa)} = \mathrm{len(Pa)} - 1 \wedge Pa \subset Pb \wedge \mathrm{supp}_{\mathrm{a}}(P_a) = \mathrm{suppa}(P_{\mathrm{b}})$
   $//Here\ Pruning\ of\ Patterns$
2. $SP \leftarrow SP \cup PL\ //Storing\ Found\ Patterns\ (nTerms)$
3. $PL' \leftarrow \{\varnothing\}\ //PL':\ set\ of\ Frequent\ Sequential$
                 $//Pattern\ (n+1)\ Term$
4. foreach $pattern\ p\ in\ PL$ do begin
5.       $generate\ p - projected\ database\ PD$
6.       foreach $frequent\ termt\ in\ PD$ do begin
7.       $P' \leftarrow p \triangleright\triangleleft t\ //P': set\ of\ (n+1)$
                                   $//\ Terms\ sequential\ candidates$
8.      if $\mathrm{supp}_r(P') \geq \min\_\sup$ then
9.         $PL' \leftarrow PL' \cup P'$
10.     end if
11.    end for
12. end for
13. if $|PL'| = 0$ then
14. return $//\ no\ more\ patterns\ found$
15. else
16. call $SPMining\ (PL', min\_sup)$
17. end if
18. $output\ frequent\ sequential\ patterns\ inSP$
```

The measures used for evaluating experimental results are precision / recall (P/R), breakeven points and the precision of top-20 returned documents. The precision is the fraction of retrieved documents that are relevant to the topic, and the recall is the fraction of relevant documents that have been retrieved. These two measures are denoted by the following formulas:

$$Precision = TP/(TP+FP) \tag{2}$$

$$Recall = TP/(TP+FN) \tag{3}$$

$$Threshold(D) = \arg\min \vec{d} \in \sum_{(t_j, n_k) \in \vec{d}} n_k \qquad (4)$$

TABLE II. DISCOVERED FREQUENT *CLOSED* AND *NON-CLOSED* SEQUENTIAL PATTERN

Frequent Patterns	Non-Close	Closed
1Term	(t_2),(t_4)	(t_1),(t_3),(t_5)
2Terms	(t1,t2),(t1,t3)	(t2,t3),(t2,t4),(t5,t3)
3Terms	Non	(t1,t2,t3)

Algorithm 2. *Deploying with Relevance Method* [1]

Input: a set of positive documents (D^+), minimum Support (*min_sup)*;

Output: New set of terms, a set of vectors (Δ).

Method:

```
$\Delta \leftarrow \varnothing$
Foreach document d in $D^+$ do begin
    Extract 1Term frequent pattern PL from d
    SP=SPMining (PL, min_sup) // call Algorithm SPMining
    $\vec{d} \leftarrow \varnothing$
    Foreach pattern p in SP do begin
        $\vec{d} \leftarrow \vec{d} \oplus p$ // p' is the expanded form of p
    End for
    $\Delta \leftarrow \Delta \cup \{\vec{d}\}\ p$
End for
```

Whereas the figures of pattern-based models (PTM and EPDM) are obviously increased to be over 0.5. This implies that both EPDM models improve the precision of the top returned documents. We compare the Effective pattern deploying method *EPDM* and *PTM* with the other three methods and illustrate in Figure 1 with results of precision all standard recall points on the first 50 topics. It can be seen that the EPDM method yields 0.78 of precision all the first recall point *(recall = 0)* and 0.67 at the second point *(recall = 0.1)*. The scores produced by the *PDM* method all the first few points are slightly less than those for the *EPDM* method with 0.8 and 0.67 at the first and second point respectively. Comparing these scores to those generated by the other methods, we find *PTM* are much superior to *tifdf* and *Prob (Probability)* methods but not quite so to the *EPDM* method. It can be seen that the *EPDM* method gives a similar score to that for the *PTM* method at the first point. This behavior corresponds to the previous finding that a data mining method is able to keep the high relevant documents in the ranked list as front as possible compared to the *tfidf* and *Prob* methods.

Algorithm 3. Evolving of pattern (Ω, D^+, D^-) [1]

Input: A list of deployed patterns Ω ; a list of positive and negative documents, D^+ and D^-

Output: A set of term weight pairs $\vec{d}$.

Method:

```
$\vec{d} \leftarrow \varnothing$ // It gives minimum threshold Value
$\tau \leftarrow threshold(D^+)$
foreach negative documents "nd" in $D^-$ do begin
    If $Threshold(\{nd\}) > \tau$ Then
        $\Delta p \leftarrow \{dp \in \Omega \mid termset(dp) \cap nd \neq \varnothing\}$
        $Shuffling(nd, \Delta p)$
    end if
    foreach deployed pattern dp in $\Omega$ do begin
        $\vec{d} \leftarrow \vec{d} \oplus p$
    end for
end for
```

Figure 1 illustrates the effect of introducing the pattern-based methods on the P/R curves on the topic 220, which can represent the trend of all topics. In this P/R curves lift up and indicate the improvement of performance made by the PTM models.

TABLE III. NUMBER OF PATTERNS FOR TEN TOPICS WITH DIFFERENT SUPPORT APPLIED IN THE TEST SET

Topic	# docs	# frequent sequential patterns		
		min_sup = 0	**min_sup=0.2**	**Min_sup=0.2 & pruning**
110	491	9977	5784	5252
120	415	5395	3933	2959
130	307	4128	1948	1845
140	432	16688	4007	3227
150	371	8492	5022	3646
160	199	4032	2060	1929
170	507	12239	6649	4745
180	426	26098	2023	1794
190	337	4382	2780	2085
200	277	3227	1996	1251
Total	3762	94658	36202	28733
Avg. P/R		**0.409**	**0.406**	**0.443**

TABLE IV. PRECISION/ RECALL, BREAKEVEN POINT FOR THE TEN TOPICS

Topic	PTM	*tfidf*	*Prob*	EPDM
110	0.419	0.161	0.226	0.548
120	0.570	0.519	0.494	0.551
130	0.063	0.063	0.063	0.313
140	0.373	0.328	0.388	0.284
150	0.111	0.130	0.222	0.167
160	0.759	0.796	0.815	0.778
170	0.493	0.411	0.384	0.397
180	0.472	0.569	0.611	0.486
190	0.529	0.494	0.553	0.588
200	0.267	0.314	0.372	0.314
Avg.	**0.406**	**0.379**	**0.413**	**0.443**

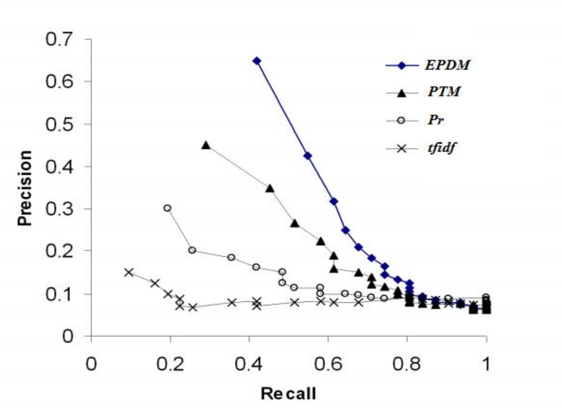

Fig. 1. P/R curve on the r110~r200 on Top 10 Topic.

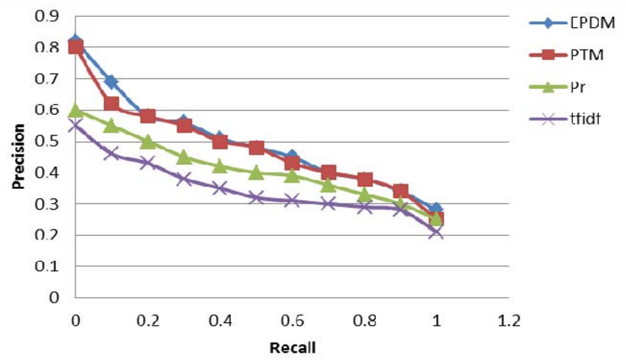

Fig. 2. Comparison of all methods in precision at standard recall point on first 50 topics.

CONCLUSION

In this paper, we have investigated the existing datamining methods with respect to the alternating approach for finding relevant pattern in large documents collection; some research works have been used phrases rather than individual words. However, the effectiveness of the text mining systems was not improved very much. The likely reason is that, a phrase-based method has "lower consistency of assignment and lower document frequency for terms". Hence, in this paper, we present a concept for mining text documents for sequential patterns. Instead of using single words, we use pattern-based taxonomy (is-a) relation to represent documents. By pruning meaningless (negative) patterns, which have been proven the source of the 'noise' in this study, the problem of over fitting is solved and the experimental results, which show the encouraging outcomes, are achieved. The results of the experiment show that the keyword-based methods not gives better performance compare to pattern-based method. The results also indicate that removal of meaningless patterns not only reduces the cost of computation but also improves the effectiveness of the system.

REFERENCES

[1] Ning Zhong, Yuefeng Li "Effective pattern discovery in text mining" *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, VOL., NO.1, January 2012.

[2] Andreas Hotho , Andreas Nurnberger "A Brief Survey of Text Mining" May 13, 2005.

[3] R. Feldman and I. Dagan. " KDT - knowledge discovery in texts". *In Proc. of the First Int. Conf. on Knowledge Discovery (KDD)* , pages 112–117, 1995.

[4] Weiguo Fan, Linda Wallace, Stephanie Rich, and Zhong ju Zhang,"Tapping into the Power of Text Mining", *Journal of ACM, Blacksburg* , (2005).

[5] D. A. Grossman and O. Frieder "Information Retrieval Algorithms and Heuristics". Kluwer Academic, 1998.

[6] Y. Li and N. Zhong. "Mining ontology for automatically acquiring web user information needs." *IEEE Transactions on Knowledge and Data Engineering*, 18(4):554-568, 2006.

[7] K. Aas and L. Eikvil. "Text categorization: A survey. Technical report.", *Norwegian Computing Center, Raport NR* 941, 1999.

[8] L. Edda and K. Jorg. "Text categorization with support vector machines. how to represent exits in input space?" *Machine Learning,* 46:423-444,2002.

[9] W. Lam. M. E. Ruiz. and P. Srinivasan. "Automatic text categorization and its application to text retrieval." *IEEE Transactions on Knowledge and Data Engineering.* 1 l (6):865-879, 1999.

[10] D. D. Lewis. "An evaluation of phrasal and clustered representations on a text categorization ask." *In Proceedings of SIGIR.* pages 37-50, 1992.

[11] F. Sebastiani. "Machine learning in automated text categorization". *ACM Computing Surveys*, 34(1): 1-47, 2002.

[12] S-T. Wu. Y. Li, Y. Xu, B. Pham. and P. Chen. "Automatic pattern-taxonomy extraction for web mining". *In Proceedings of the IEEEJWI/AM International conference on Web Intelligence* (W104), pages 242-248, 2004.

[13] S-T. Wu, Y. Li, and Y Xu. "Deploying approaches for pattern refinement in text mining". *In Proceedings of IC'DM, pages* 1157-1161, 2006.

[14] S-T. Wu, Y. Li, and Y. Xu. "An effective deploying algorithm for using pattern-taxonomy". *In Proceedings of the 7th international Conference on information Integration and Web-based Applications & Services* (1iWASO5), pages 1013-1022, 2005

[15] R. Agrawal, and R. Srikant, "Mining sequential patterns," *Proceedings of Int. Conf. on Data engineering (ICDE'95),Taipei, Taiwan*, pp. 3-14, 1995.

[16] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu, "PrefixSpan: Mining sequential patterns efficiently by prefix projected pattern growth," *Proceedings of Int. Conf. on Data Engineering (ICDE'02), Heidelberg, Germany,* 2001, pp. 215-224.

[17] M. Zaki, "SPADE: An efficient algorithm for mining frequent sequences," *Machine Learning vol.* 40, 2001, pp. 31-60.

[18] Y. J. Fu. "Data mining: Tasks, techniques and applications". *IEEE Potentials*.16(4):18-20, 1997.

[19] S. T. Wu, Y. Li, and Y. Xu. " Deploying approaches for pattern refinement in text mining". *In Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006),* pages 1157-1161, 2006.

[20] K. Sparck Jones. "Experiments in relevance weighting of search terms." *Inf. Process. Manage*., 15(3):133-144. 1979.

[21] *S. Scott and S. Matwin. "*Feature engineering for text classification*". In Proceedings of* ICML. pages 379-388. 1999.