ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Normalized Pattern Taxonomy Model for Effective Pattern Discovery

N.Rama Devi
M.Tech (IT),
S.R.K.R. Engineering College.

Abstract:
Text mining is the technique that helps users find useful information from a large amount of digital text documents on the Web or databases. Instead of the keyword-based approach which is typically used in this field, the pattern based model containing frequent sequential patterns is employed to perform the same concept of tasks. However, how to effectively use and update discovered patterns is still an open research issue, especially in the domain of text mining. Since most existing text mining methods adopted term-based approaches, they all suffer from the problems of polyse my and synonymy. Over the years, people have often held the hypothesis that pattern (or phrase)-based approaches should perform better than the term-based ones, but many experiments do not support this hypothesis. Here we present an innovative and normalized pattern taxonomy model for effective pattern discovery technique 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. The effective pattern discovery technique has been introduced to overcome the low-frequency and misinterpretation problems for text mining, which first calculates discovered specificities of patterns and then evaluates term weights according to the distribution of terms in the discovered patterns rather than the distribution in documents for solving the misinterpretation problem. It also considers the influence of patterns from the negative training examples to find ambiguous (noisy) patterns and try to reduce their influence for the low-frequency problem. The process of updating ambiguous patterns can be referred as pattern evolution. The proposed approach can improve the accuracy of evaluating term weights because discovered patterns are more specific than whole documents. Substantial experiments on RCV1 data collection and TREC topics demonstrate that the proposed solution achieves encouraging performance.

1. INTRODUCTION:

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. 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. 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. In the past decade, a significant number of data mining techniques have been presented in order to perform different knowledge tasks. These techniques include association rule mining, frequent item set mining, sequential pattern mining, maximum pattern mining, and closed pattern mining. Most of them are proposed for the purpose of developing efficient mining algorithms to find particular patterns within a reasonable and acceptable time frame. With a large number of patterns generated by using data mining approaches, how to effectively use and update these patterns is still an open research issue.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Therefore, we focus on the development of a knowledge discovery model to effectively use and update the discovered patterns and apply it to the field of text mining. Text mining is the discovery of interesting knowledge in text documents. It is a challenging issue to find accurate knowledge (or features) in text documents to help users to find what they want. In the beginning, Information Retrieval (IR) provided many term-based methods to solve this challenge, such as Rocchio and probabilistic models, rough set models, BM25 and support vector machine (SVM) based filtering models. The advantages of term based methods include efficient computational performance as well as mature theories for term weighting, which have emerged over the last couple of decades from the IR and machine learning communities. However, term based methods suffer from the problems of polysemy and synonymy, where polysemy means a word has multiple meanings, and synonymy is multiple words having the same meaning.

The semantic meaning of many discovered terms is uncertain for answering what users want. There are two fundamental issues regarding the effectiveness of pattern-based approaches: low frequency and misinterpretation. 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. If we decrease the minimum support, a lot of noisy patterns would be discovered. Misinterpretation means the measures used in pattern mining (e.g., "support" and "confidence") turn out to be not suitable in using discovered patterns to answer what users want. The difficult problem hence is how to use discovered patterns to accurately evaluate the weights of useful features (knowledge) in text documents. The main objective of "Pattern taxonomy model for effective pattern discovery" is that phrase-based approaches could perform better than the term based ones, as phrases may carry more "semantics" like information. Although phrases are less ambiguous and more discriminative than individual terms. In the presence of these setbacks, sequential patterns used in data mining community have turned out to be a promising alternative to phrases because sequential patterns enjoy good statistical properties like terms. To overcome the disadvantages of phrase-based approaches, pattern mining-based approaches (or pattern taxonomy models (PTM)) have been proposed, which adopted the concept of closed sequential patterns, and pruned non closed patterns. These pattern mining-based approaches have shown certain extent improvements on the effectiveness. However, the paradox is that people think pattern-based approaches could be a significant alternative, but consequently less significant improvements are made for the effectiveness compared with term-based methods. The paper is organized as follows. Chapter 2 discusses literature review, Chapter 3 discusses Pattern Taxonomy Model, Chapter 4 System Architecture, Chapter 5 discusses about Result Analysis, Chapter 6 discusses about Conclusion, and Chapter 7 discusses about future work and Chapter 8 consists of References.

2. RELATED WORK:

Literature Survey is the most important step in software development process. 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 [1] 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 [2] and improves performance by an average of 30 percent. Various weighting schemes for the bag of words representation approach were given in [3], [4], [5]. 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 [6]. 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. The choice of a representation depended on what one regards as the meaningful units of text and the meaningful natural

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

language rules for the combination of these units [6]. With respect to the representation of the content of documents, some research works have used phrases rather than individual words. In [7] the combination of unigram and bigrams was chosen for document indexing in text categorization (TC) and evaluated on a variety of feature evaluation functions (FEF). A phrase-based text representation for Web document management was also proposed. In [8], Data mining techniques have been used for text analysis by extracting co-occurring terms as descriptive phrases from document collections. However, the effectiveness of the text mining systems using phrases as text representation showed no significant improvement. The likely reason was that a phrase-based method had "lower consistency of assignment and lower document frequency for terms". Term-based ontology mining methods also provided some thoughts for text representations. For example, hierarchical clustering [9] was used to determine synonymy and hyponymy relations between keywords. Also, the pattern evolution technique was introduced in order to improve the performance of term-based ontology mining. Pattern mining has been extensively studied in data mining communities for many years.

A variety of efficient algorithms such as Apriori-like algorithms [10], FP-tree [11] has 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 [12], 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 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 to significantly improve the performance of information filtering. 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 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

3. PATTERN TAXONOMY MODEL:

In this paper, we assume that all 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= {t1, t2,……..tm}g be a set of terms (or keywords) which can be extracted from the set of positive documents, D+.

3.1 Frequent and Closed patterns:

Given a termset X in document d, [X] is used to denote the covering set of X for d, which includes all paragraphs dp ∈ PS (d) such that X⊆ dp, i.e., X={dp | dp ∈PS(d),X ⊆dp). Its absolute support is the number of occurrences of X in PS(d), that is supa (X) = |X| .

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Its relative support is the fraction of the paragraphs that contain the pattern, that is, supr (X) =|X|/PS(d)| .
A term set X is called frequent pattern if its supr (or supa)>= min _sup, a minimum support. Table 1 lists a set of paragraphs for a given document d, where PS(d) ={dp1, dp2, . . . ,dp6}, and duplicate terms were removed .Let min_sup = 50%,we can obtain ten frequent patterns in Table 1 using above definitions. Table 2 illustrates the ten frequent patterns and heir covering sets

TABLE 1
A Set of Paragraphs.

Frequent Pattern	*Covering Set*
$\{t_3, t_4, t_6\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_3, t_4\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_3, t_6\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_4, t_6\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_3\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_4\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_1, t_2\}$	$\{dp_1, dp_5, dp_6\}$
$\{t_1\}$	$\{dp_1, dp_5, dp_6\}$
$\{t_2\}$	$\{dp_1, dp_5, dp_6\}$
$\{t_6\}$	$\{dp_2, dp_3, dp_4, dp_5, dp_6\}$

TABLE 2
Frequent patterns and covering sets

Frequent Pattern	*Covering Set*
$\{t_3, t_4, t_6\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_3, t_4\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_3, t_6\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_4, t_6\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_3\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_4\}$	$\{dp_2, dp_3, dp_4\}$
$\{t_1, t_2\}$	$\{dp_1, dp_5, dp_6\}$
$\{t_1\}$	$\{dp_1, dp_5, dp_6\}$
$\{t_2\}$	$\{dp_1, dp_5, dp_6\}$
$\{t_6\}$	$\{dp_2, dp_3, dp_4, dp_5, dp_6\}$

Not all frequent patterns in Table 2 are useful. For example, pattern {t3, t4} always occurs with term t6 in paragraphs, i.e., the shorter pattern, {t3; t4}, is always a part of the larger pattern, {t3; t4; t6}, in all of the paragraphs. Hence, we believe that the shorter one, {t3; t4}, is a noise pattern and expect to keep the larger pattern, {t3; t4; t6}, only. Given a term set X, its covering set X is a subset of paragraphs.

3.2 Pattern Taxonomy:

Patterns can be into a taxonomy by using the is-a (or subset) relation. For the example of Table 1, where we have illustrated a set of paragraphs of a document, and the discovered 10 frequent patterns in Table 2 if assuming min_sup=50%. There are, however, only three

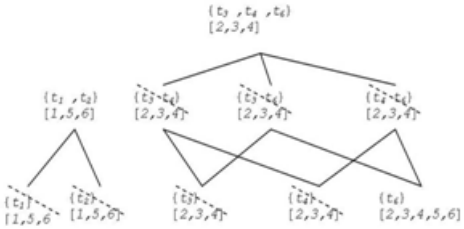

Fig 1.Pattern Taxonomy

Closed patterns in this example they are <t3; t4; t6 >, <t1; t2 >, and <t6 >. Fig. 1 illustrates an example of the pattern taxonomy for the frequent patterns in Table 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 {t6} would become a direct sub pattern of {t3, t4, t6} after pruning non closed patterns. Smaller patterns in the taxonomy, for example pattern {t6} (see Fig. 1) are usually more general because they could be used frequently in both positive and negative documents; and larger patterns, for example pattern ft3 ; t4 ; t6 g, 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.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

4. SYSTEM ARCHITECTURE

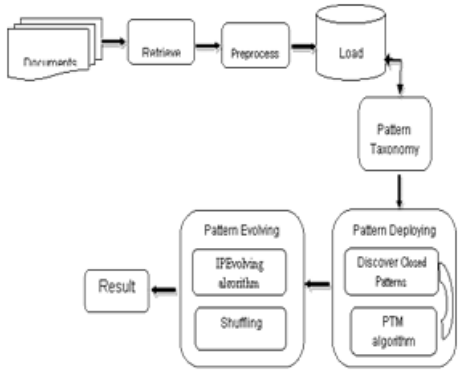

Fig 2: Architecture of pattern taxonomy model for effective pattern discovery.

In Fig 2 the architecture of pattern taxonomy model for effective pattern discovery referred to as text data mining roughly equal to text analysts which refers to the process of high quality information from the text. High quality information is typically derived through the devising of patterns and runs through means such as statistical pattern learning. Text mining involves the process of structuring the input text deriving patterns within the structure data and finally evaluating and interpretation of the output. High quality in text mining usually refers to some combination of relevance normality. In proposed system the effective pattern discovery technique is applied, it evaluates patterns and then evaluates term weights according to the distribution of the terms in the discover patterns. The pattern deploying which is used to improve the efficiency of pattern taxonomy in mining and the pattern evolving technique is used for updating ambiguous patterns are used to improve the effectiveness of the patterns. There are two fundamental issues regarding the effectiveness of the pattern based approaches: low frequency and misinterpretation. To solve the paradox to use effective pattern discovery technique which first calculates discovered specifications of patterns and then evaluates term weights according to the distribution of terms in the discovered patterns rather than distribution in documents for solving the misinterpretation problem.

It also consider the influence of patterns from the negative training to find ambiguous patterns and try to reduce their influence for the low frequency problem.

4.1 Pattern Deploying:

In order to use the semantic information in the pattern taxonomy to improve the performance of closed patterns in text mining, we need to interpret discovered patterns by summarizing them as d-patterns. The PTM algorithm is used to discover all patterns in positive documents are composed. The term supports are calculated by all terms in d-pattern. Term support means weight of the term is evaluated. The rational behind this motivation is that d-patterns include more semantic meaning than terms that are selected based on a term-based technique (e.g., tf*idf). As a result, a term with a higher tf*idf value could be meaningless if it has not cited by some d-patterns (some important parts in documents). The evaluation of term weights (supports) is different to the normal term-based approaches. In the term-based approaches, the evaluation of term weights are based on the distribution of terms in documents. In this research, terms are weighted according to their appearances in discovered closed patterns.

Input: Positive documents D^+; min_sup
Output: d-patterns DP and term support

Step 1: DP=0;
Step 2: for each document d ∈ D+
Let ps (d) set of paragraphs in d;
SP= SPMining (PS (d), min_sup);
d'=0;
Step 3: for each pattern pi ∈ sp do
$p = \{(t, 1)/t \in pi\}$;
$d' = d' \oplus p$;
End
End
Step 4: DP=DP U {d'};
End
Step 5: $T = \{t/(t, f) \in p, p \in DP\}$;
Step 6: for each term t ∈ T
Support (t) =0;
End
Step 7: for each d-pattern p ∈ DP do
Step 8: for each (t, w) ∈ β (p) do
Support (t) = Support (t) + w;
End
End

Algorithm 1: PTM (D+, min_sup)

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Before applying the algorithm we have to assume that document patterns are null i.e. DP={0}.The documents which we are taking should belong to the main set. Let PS (d) represents the set of paragraphs in the document. For every positive document, the SPMining algorithm is first called to give rise to a set of closed sequential patterns SP and the discovered patterns in the positive document are composed into a d-pattern giving rise to a set of d-patterns DP. We need to take a term and perform pattern reorganization with the single pattern and we need to add the document point to data patterns. Then the term supports are calculated based on the normal forms for all terms in d-patterns.

4.2 Pattern Evolving:

In this section, described how to reshuffle supports of terms within normal forms of d-patterns based on negative documents in the training set. 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. Using the d-patterns, the threshold can be defined naturally as follows:

$$Threshold(DP) = \min_{p \in DP}\left(\sum_{(t,w)\in\beta(p)} support(t)\right)$$

A noise negative document nd in D_ is a negative document that the system falsely identified as a positive, that is weight (nd) >= Threshold (DP). 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 nd. An offender of nd is a d-pattern that has at least one term in nd. The set of offenders of nd is defined by:

$$\Delta(nd) = \{p \in DP / termset(p) \cap nd \neq 0\}$$

There are two types of offenders: 1) a complete conflict offender which is a subset of nd; and 2) a partial conflict offender which contains part of terms of nd. 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.

Input: A training set $D=D^{+}UD^{-}$; a set of d-patterns DP; and an experimental coefficient μ

Output: a set of term –support pairs np.

Step 1: np→0;

Step 2: threshold= Threshold (DP).

Step 3: for each noise negative document nd ∈ D- do

Algorithm 2: IPEvolving(D+, D-, DP, μ)

The inputs of this algorithm are a set of d-patterns DP, a training set $D=D^{+}UD^{-}$. The output is a composed of d-pattern. Step 2 in IP Evolving is used to estimate the threshold for finding the noise negative documents. Steps 3 to 10 revise term supports by using all noise negative documents. Step 4 is to find noise documents and the corresponding offenders. Step 5 gets normal forms of d-patterns NDP. Step 6 calls algorithm shuffling to update NDP according to noise documents. Steps 7 to 9 compose updated normal forms together.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Input: a noise document nd. Its offenders Δ (nd), normal forms of d-patterns NDP, and an experimental coefficient μ.

Output: updated normal forms of d-patterns NDP.

Step 1: for each p in Δ (nd) do

Step 2: if termset(p) ⊆ nd then

NDP=NDP-{β (p)};

Step 3: else

offering= $\left(1-\frac{1}{\mu}\right)\times\sum_{t\in(\text{termset}\ (p)\ \text{and}\ )}\text{support}(t)$;

base= $\sum_{t\in(\text{termset}\ (p)-\text{nd})}\text{support}(t)$;

Step 4: for each term t in termset (p) do

if t ∈ nd then

Algorithm 3: Shuffling (nd, Δ (nd), NDP, μ)

The inputs of this algorithm are a noise document nd, its offenders Δ (nd), normal forms of d-patterns NDP, and an experimental coefficient μ.The task of this algorithm is to tune the support distribution of terms within a d-pattern. The complete conflict offenders are removed if all the elements within the d-patterns are held by the negative documents and the parameter offering is used for the purpose of temporarily storing the reduced supports of some terms in partial conflict offenders. It calculates base which is certainly not zero and updates the support distribution of terms.

5. RESULT ANALYSIS:

The results are observing in the two modules. In first module, we preprocess the text document so that we can easily classify the document. By preprocessing the text data, the commonality words and stop words are removed.

Table 3.Comparison of file size in preprocessing tasks

Preprocessing task	File Size (%)
Original file	100%
Stop Words Removal	99.5%
Commonality Words Removal	99.3%

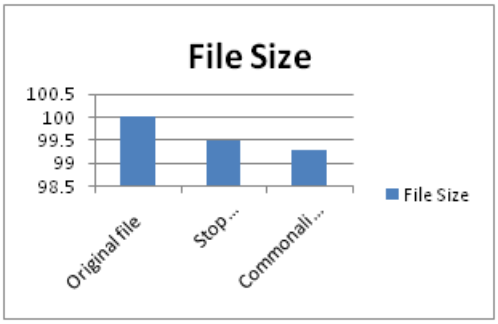

Fig 3 .Comparison of file size in preprocessing tasks.

The above Fig represents the file size in each preprocessing task. In each preprocessing task, we remove the unrelated words or symbols, so the file size is decreased. In this example, after preprocessing, the file size is reduced. In second module, we apply the Pattern Taxonomy process. In which the document is divided into paragraphs and each paragraph is consider as each document, from each document set of terms have been extracted. The proposed technique uses two processes, pattern deploying and pattern evolving.

Table 4.Comparison of different datasets for calculating the threshold value.

Pattern Taxonomy Process	D-patterns term weight	Term Support	Threshold Value
Dataset 1	8	3	12
Dataset 2	7	2	14
Dataset 3	9	2	18

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

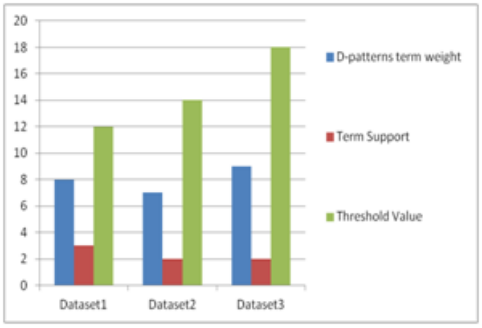

Fig 4. Comparison of different datasets for calculating the threshold value.

The above Fig 4 represents the comparison of different datasets for calculating the threshold value. The process contains the pattern deploying and the pattern evolving techniques which are used for finding the effectiveness of the patterns. The pattern deploying is used to improve the efficiency of pattern taxonomy in mining in which the closed sequential patterns which are greater than the minimum support are extracted then the patterns are structured into taxonomy by using subset relation. It evaluates the terms that could be used frequently in both positive and negative documents. Term supports are calculated by all terms in d-pattern. Based on the terms and their term supports the threshold value is calculated. If the weight of the negative document is greater than the threshold value then we apply shuffling process which is used to remove the noise data from the documents in order to slove the low frequency problem.

6. CONCLUSION:

The Pattern taxonomy model technique has been introduced to overcome the low-frequency and misinterpretation problems for text mining, which first calculates discovered specificities of patterns and then evaluates term weights according to the distribution of terms in the discovered patterns rather than the distribution in documents for solving the misinterpretation problem.

It also considers the influence of patterns from the negative training examples to find ambiguous (noisy) patterns and try to reduce their influence for the low-frequency problem. Hence, misinterpretations of patterns derived from data mining techniques lead to the effective performance.

7. FUTURE ENHANCEMENT:

In this paper work, there are two phases, the training phase and the testing phase. In training Phase the proposed model finds d-patterns in positive documents based on minimum support and evaluates term supports by deploying d-patterns to terms and it revise term supports using noise negative documents based on experimental coefficient. In testing phase it evaluates weights of all incoming documents. The incoming documents then can be sorted based on these weights.

8. REFERENCES:

[1]X. Li and B. Liu, "Learning to Classify Texts Using Positive and Unlabeled Data," Proc. Int'l Joint Conf. Artificial Intelligence (IJCAI '03), pp. 587-594, 2003.

[2]S.T. Dumais, "Improving the Retrieval of Information from External Sources," Behavior Research Methods, Instruments, and Computers, vol. 23, no. 2, pp. 229-236, 1991.

[3]K. Aas and L. Eikvil, "Text Categorisation: A Survey," Technical Report Raport NR 941, Norwegian Computing Center, 1999.

[4]T. Joachims, "A Probabilistic Analysis of the Rocchio Algorithm with tfidf for Text Categorization," Proc. 14th Int'l Conf. Machine Learning (ICML '97), pp. 143-151, 1997.

[5]G. Salton and C. Buckley, "Term-Weighting Approaches in Automatic Text Retrieval," Information Processing and Management: An Int'l J., vol. 24, no. 5, pp. 513-523, 1988.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[6]F. Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, vol. 34, no. 1, pp. 1-47, 2002.

[7]M.F. Caropreso, S. Matwin, and F. Sebastiani, "Statistical Phrases in Automated Text Categorization," Technical Report IEI-B4-07-2000, Instituto di Elaborazione dell'Informazione, 2000.

[8]H. Ahonen, O. Heinonen, M. Klemettinen, and A.I. Verkamo, "Applying Data Mining Techniques for Descriptive Phrase Extraction in Digital Document Collections," Proc. IEEE Int'l Forum on Research and Technology Advances in Digital Libraries (ADL '98), pp. 2-11,1998.

[9]A. Maedche, Ontology Learning for the Semantic Web. Kluwer Academic, 2003.

[10]R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," Proc. 20th Int'l Conf. Very Large Data Bases (VLDB '94), pp. 478-499, 1994.

[11] J. Han and K.C.-C. Chang, "Data Mining for Web Intelligence," Computer, vol. 35, no. 11, pp. 64-70, Nov. 2002.

[12]S.-T. Wu, Y. Li, Y. Xu, B. Pham, and P. Chen, "Automatic Pattern- Taxonomy Extraction for Web Mining," Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence (WI '04), pp. 242-248, 2004.