Mining Specific Features for Acquiring User Information Needs

Abdulmohsen Algarni[1] and Yuefeng Li[2,⋆]

[1] College of Computer Science, King Khalid University
Saudi Arabia, B.O.Box 394, ABHA 61411
a.algarni@kku.edu.sa
[2] School of Electrical Engineering and Computer Science
Queensland University of Technology, Brisbane, QLD 4001, Australia
y2.li@qut.edu.au

Abstract. Term-based approaches can extract many features in text documents, but most include noise. Many popular text-mining strategies have been adapted to reduce noisy information from extracted features; however, text-mining techniques suffer from low frequency. The key issue is how to discover relevance features in text documents to fulfil user information needs. To address this issue, we propose a new method to extract specific features from user relevance feedback. The proposed approach includes two stages. The first stage extracts topics (or patterns) from text documents to focus on interesting topics. In the second stage, topics are deployed to lower level terms to address the low-frequency problem and find specific terms. The specific terms are determined based on their appearances in relevance feedback and their distribution in topics or high-level patterns. We test our proposed method with extensive experiments in the Reuters Corpus Volume 1 dataset and TREC topics. Results show that our proposed approach significantly outperforms the state-of-the-art models.

Keywords: Feature extraction, Pattern mining, Relevance feedback, Text classification.

1 Introduction

One of the objectives of knowledge extraction is to build user profiles by finding a set of features from feedback documents to describe user information needs. This is a particularly challenging task in modern information analysis, empirically and theoretically [11,13]. This problem has received much attention from the data mining, web intelligence, and information retrieval communities.

Information retrieval has deployed many effective term-based methods to find popular terms [14]. The advantages of term-based methods include efficient computational performance and mature theories for term weighting. However, many noisy terms can be extracted from the large-scale feedback documents. Words

⋆ Corresponding author.

J. Pei et al. (Eds.): PAKDD 2013, Part I, LNAI 7818, pp. 532–543, 2013.

and phrases have also been used as terms in many models. Many researchers believe phrases are more useful and crucial than words for query expansion in building effective ranking functions [14,4,22]. However, there are usually many redundant and noisy phrases [18,19].

Popular terms are useful for describing documents; however, they do not focus on the interesting topics in these documents. We argue that patterns (itemsets, or sets of terms) can be a good alternative form of terms for describing interesting topics in documents.

Data-mining techniques have been developed–e.g., maximal, closed, and master patterns–for removing redundant and noisy patterns [27,25]. By using the advantages of data-mining techniques, pattern taxonomy models (PTM) [24,23,12] have been proposed for using closed sequential patterns in text classification. These pattern mining-based approaches have improved the effectiveness for relevant (positive) feedback documents, but offer fewer significant improvements compared with term-based methods for using both relevant and irrelevant feedback.

Existing approaches focus more on extracting general or popular topics from feedback documents rather than what users really want. Several attempts have been made to determine terms specificity regarding to term distribution in documents. For example, Inverse Document Frequency (*IDF*) measures terms specificity in a set of documents, but much noisy information in text documents affects *IDF* for finding specific features.

This research proposes a specificity definition for mining specific features for user information needs. The proposed approach includes two stages. The first stage extracts high-level patterns (or topics) from text documents to focus on interesting topics in order to reduce the noise. The second stage deploys these topics (high-level patterns) to lower level terms to address the low-frequency problem in order to find specific features. The proposed approach can determine specific terms based on both their appearances in relevance feedback and their distribution in interesting topics (or high-level patterns).

The remainder of this paper is organized as follows. Section 2 introduces a detailed overview of related works. Section 3 reviews concepts of patterns in text documents. Section 4 proposes a method of mining specific features from positive feedback documents. Section 5 shows empirical results; Section 6 reports related discussions, followed by the final sections concluding remarks.

2 Related Work

Scientists have proposed many types of text representation. A well-known one is the bag-of-words model that uses keywords, or terms, in the vector of the feature space. In [9], the *tf*idf* weighting scheme was used for text representation in Rocchio classifiers. Enhanced from *tf*idf*, the global IDF and entropy weighting scheme was proposed in [5]. Various weighting schemes for the bag-of-words representation were given in [1,7].

Bag of words problem is how to select a limited number of feature terms to increase the system's efficiency and avoid *overfitting* [19]. To reduce the number

of features, many dimensionality reduction approaches have been conducted using feature selection techniques like information gain, mutual information, chi-square, and odds ratio [19].

In [2], data-mining techniques analyzed text by extracting co-occurring terms as descriptive phrases from document collections. However, the effectiveness of the text classification systems using phrases as text representation showed no significant improvement. The likely reason is that a phrase-based method has lower consistency of assignment and lower document frequency for terms [8].

The data-mining community has extensively studied pattern mining for many years. Usually, existing data-mining techniques discover numerous patterns (e.g., sets of terms) from a training set, but many patterns may be redundant [25]. Nevertheless, the challenge is dealing effectively with the many discovered patterns and terms with much noise.

Regarding to these setbacks, closed patterns present a promising alternative to phrases [23,6]. Patterns, like terms, enjoy good statistical properties. To use closed patterns effectively in text mining, patterns have been evaluated by being deployed into a vector with a set of terms and term-weight distributions. The pattern-deploying method encouragingly improves effectiveness compared with traditional probabilistic models and Rocchio-based methods [23,12].

In summary, we can group the existing methods for finding relevance features into three approaches. The first one is to revise feature terms in both positive and negative samples, like Rocchio-based models [15]. The second approach is based on how often terms appear or do not appear in positive and negative samples like probabilistic-based models [26]. The third approach is to describe specific features based on their appearances in both patterns or/and documents [23,10]. In this paper, we further develop the third approach to utilize high-level patterns (or topics) extracted from only positive samples (relevant documents) for finding specific terms. The major research issue is how to determine the topics specificity of terms according their distributions in both documents and topics.

3 Definition

In this paper, we assume that all documents are split in 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 $T = \{t_1, t_2, \ldots, t_m\}$ be a set of terms (or keywords) which are extracted from the set of positive documents, D^+.

3.1 Frequent and Closed Patterns

Let $T = \{t_1, t_2, \ldots, t_m\}$ be a set of terms which are extracted from D^+. Given a *termset* X, a set of terms, in document d, $coverset(X) = \{dp | dp \in PS(d), X \subseteq dp\}$. Its *absolute support* $sup_a(X) = |coverset(X)|$; and its *relative support* $sup_r(X) = \frac{|coverset(X)|}{|PS(d)|}$. A termset X is called *frequent pattern* if its sup_a (or sup_r) $\geq min\_sup$, a minimum support.

Table 1. A set of paragraphs

Parapgraph	*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_7

Table 2. Frequent patterns and covering sets

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

Table 1 lists a set of paragraphs for a given document d, where $PS(d) = \{dp_1, dp_2, \ldots, dp_6\}$, and duplicate terms are removed. Let $min\_sup = 3$ giving rise to ten frequent patterns which are illustrated in Table 2. Normally not all frequent patterns are useful [24,25]. For example, pattern $\{t_3, t_4\}$ always occurs with term t_6 in paragraphs (see Table 1); therefore, we want to keep the larger pattern only.

Given a set of paragraphs $Y \subseteq PS(d)$, we can define its *termset*, which satisfies

$$termset(Y) = \{t | \forall dp \in Y \Rightarrow t \in dp\}.$$

Let $Cls(X) = termset(coverset(X))$ be the closure of X. We call X *closed* if and only if $X = Cls(X)$.

Let X be a closed pattern. We have

$$sup_a(X_1) < sup_a(X) \tag{1}$$

for all pattern $X_1 \supset X$.

3.2 Closed Sequential Patterns

A sequential pattern $s =< t_1, \ldots, t_r >$ $(t_i \in T)$ is an ordered list of terms. A sequence $s_1 =< x_1, \ldots, x_i >$ is a sub-sequence of another sequence $s_2 =< y_1, \ldots, y_j >$, denoted by $s_1 \sqsubseteq s_2$, iff $\exists j_1, \ldots, j_i$ such that $1 \leq j_1 < j_2 \ldots < j_i \leq j$ and $x_1 = y_{j_1}, x_2 = y_{j_2}, \ldots, x_i = y_{j_i}$. Given $s_1 \sqsubseteq s_2$, we usually say s_1 is a sub-pattern of s_2, and s_2 is a super-pattern of s_1. In the following, we simply say patterns for sequential patterns.

Given a pattern (an ordered *termset*) X in document d, $\ulcorner X \urcorner$ is still used to denote the covering set of X, which includes all paragraphs $ps \in PS(d)$ such that $X \sqsubseteq ps$, i.e., $\ulcorner X \urcorner = \{ps | ps \in PS(d), X \sqsubseteq ps\}$. Its *absolute support* and *relative support* are defined as the same as for the normal patterns.

A sequential pattern X is called *frequent pattern* if its relative support $\geq$ *min_sup*, a minimum support. The property of closed patterns can be used to define closed sequential patterns. A frequent sequential pattern X is called *closed* if not $\exists$ any super-pattern X_1 of X such that $sup_a(X_1) = sup_a(X)$.

4 Acquiring User Information Needs

Extracting high-level patterns from text documents would help us to focus on interesting topics in positive (relevant) feedback documents. In this paper, a feature's specificity describes the extent to which the feature focuses on interesting topics.

4.1 Two Levels of Features

Term-based user profiles are considered the most mature theories for term weighting to emerge over the last couple decades in information retrieval. A term-based model is based on the bag of words, which uses terms as elements and evaluates term weights based on terms' appearance or distribution in documents. This main drawback is that the relationship among words cannot be depicted [20]. Another problem in considering single words as features is semantic ambiguities, such as synonyms and polysemy. To overcome the limitations of term-based approaches, pattern mining-based techniques are used for information filtering systems, as patterns are less ambiguous and more discriminative than individual terms; but, pattern-based approaches suffer from low frequency problem.

To improve the efficiency of pattern taxonomy mining, an algorithm, *SPMining*(D^+, min_sup) [24], was proposed to find closed sequential patterns in paragraphs for all documents $\in D^+$ that used the well-known *Apriori* property to reduce searching space. For all positive documents $d \in D^+$, the *SPMining* algorithm discovered all closed sequential patterns based on a given *min_sup*.

Let SP_1, SP_2, ..., SP_n be the sets of discovered closed sequential patterns for all document $d_i \in D^+(i = 1, \cdots, n)$, where $n = |D^+|$). All possible candidates of specific terms can be obtained from all SP_i $(i = 1, \cdots, n)$ as follows:

$$T = \bigcup_{i=1}^{n} \{t | t \in p, p \in SP_i\}$$

4.2 Specificity of Low-level Features

We assume a topic is a set of terms. We call a topic interesting if it is a closed sequential pattern. Usually large number of topics can be extracted from feedback documents, and there are overlaps among many topics. It is very difficult to determine which topic are useful to describe user information needs. Moreover, the user can be interested in one or many topics. Thus, in this paper we define the specificity of a term t based on the specificity of topics that contain t (ST); its distribution in topics (or term's frequency in patterns, TFP) and its appearance in documents (or called document frequency, DF)

The specificity of any given term t to topics ST can be measured by the topics size and the terms distribution in topics that contain t. The topic that contains more terms is unlikely used by other irrelevant documents, and then it is more

likely a specific topic for user information needs. Thus, a large topic that contains term t is more important than a short topic. Moreover, the relative support of topics is significant for measure the term's specificity. Therefore, the specificity of a term to topics ST can be calculated as follows:

$$ST(t) = \sum_{i=1}^{n} \sum_{t \in p \subseteq SP_i} sup_r(p, d_i) \times |p|$$

where $sup_r(p, d_i)$ is the relative support of pattern p in document d_i and n is the number of positive feedback documents.

All positive documents in the user feedback describe user information needs. In other words, terms that appear in all positive documents are likely specific features. For example, if the user seeks information about Unicef, we expect the keyword Unicef appear in all positive feedback documents; however, many feedback documents contain noisy terms. To reduce noisy terms, the terms frequency will be calculated only according to their appearances in the extracted topics rather than in documents.

Based on the above analysis, in this paper, we propose the following equation to calculate the specificity of term t for all $t \in T$:

$$\begin{aligned} spe(t) &= TFP(t) \times DF(t) \times ST(t) \\ &= \left(\sum_{i=1}^{n} \sum_{t \in p \subseteq SP_i} sup_a(p, d_i) \right) \times \left(\frac{|coverage(t, D^+)|}{|D^+|} \right) \times \\ &\quad \left(\sum_{j=1}^{n} \sum_{t \in p \subseteq SP_j} \left(sup_r(p, d_j) \times |p| \right) \right) \\ &= \frac{1}{n} |coverage(t, D^+)| \times \left(\sum_{i=1}^{n} \sum_{t \in p \subseteq SP_i} sup_a(p, d_i) \right) \times \\ &\quad \left(\sum_{j=1}^{n} \sum_{t \in p \subseteq SP_j} \left(sup_r(p, d_j) \times |p| \right) \right) \\ &= \frac{r(t)}{n} \times \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\sum_{t \in p \subseteq SP_i} sup_a(p, d_i) \times \sum_{t \in q \subseteq SP_j} \left(sup_r(q, d_j) \times |q| \right) \right) \end{aligned}$$

where, $sup_a(p, d_i)$ is the frequency of patterns that contain term t in document d; $r(t) = |coverage(t, D^+)|$ is the number of positive documents that contain terms t; and n is the total number of positive documents.

For a term t, the higher of its *spe* score is, the more useful for describing the user information needs. The relevance of an incoming document d to the user information needs can be evaluated using the following ranking function:

$$rank(d) = \sum_{t \in T} spe(t) \tau(t, d)$$

where $\tau(t, d) = 1$ if $t \in d$; $\tau(t, d) = 0$ otherwise.

5 Evaluation

In this paper, we conduct binary text classification to test the proposed approach. We use routing filtering to avoid the need for threshold tuning, which is beyond our research scope. The proposed model in this paper is called Specific Feature Discovery (SFD). The SFD model uses positive relevance feedback to build user profiles. Unlike other models, it uses only positive feedback for selecting useful features.

According to Buckley and others [3], 50 topics are adequate to make a stable, high quality experiment. This evaluation used the 50 expert-designed topics in Reuters Corpus Volume 1 (RCV1) [21]. RCV1 corpus consists of 806,791 documents produced by Reuter's journalists. The document collection is divided into training sets and test sets. These topics were developed by human assessors of the National Institute of Standards and Technology (NIST). The documents are treated as plain text documents by preprocessing the documents. The tasks of removing stop-words according to a given stop-words list and stemming term by applying the Porter Stemming algorithm are conducted.

5.1 Baseline Models and Setting

The main baseline models were the well-known term-based methods: Rocchio, BM25 and SVM. The Rocchio algorithm [17] has been widely adopted in the areas of text categorization and information filtering. It can be used to build a profile for representing the concept of a topic which consists of a set of relevant (positive) and irrelevant (negative) documents. we set $\alpha = \beta = 1.0$ in this paper.

BM25 [16] is one of state-of-the-art term-based models. The values of k_1 and b are set as 1.2 and 0.75, respectively, in this paper.

Information filtering can also be regarded as a special form of text classification [19]. SVM is a statistical method that can be used to find a hyperplane that best separates two classes. SVM achieved the best performance on the Reuters-21578 data collection for document classification [28]. The decision function in SVM is defined as:

$$h(x) = sign(w \cdot x + b) = \begin{cases} +1 & \text{if } (w \cdot x + b) > 0 \\ -1 & \text{otherwise} \end{cases}$$

where x is the input object; $b \in \Re$ is a threshold and $w = \sum_{i=1}^{l} y_i \alpha_i x_i$ for the given training data: $(x_i, y_i), ..., (x_l, y_l)$, where $x_i \varepsilon \Re^n$ and $y_i = +1(-1)$, if document x_i is labelled positive (negative). $\alpha_i \in \Re$ is the weight of the sample x_i and satisfies the constraint:

$$\forall_i : \alpha_i \geq 0 \quad \text{and} \quad \sum_{i=1}^{l} \alpha_i y_i = 0 \tag{2}$$

To compare with other baseline models, SVM was used to rank documents rather than to make binary decisions. For this purpose, threshold b was ignored. For the documents in a training set, we knew only what were positive (or negative), but not which one was more important. To avoid this bias, we assigned the same α_i value (i.e., 1) to each positive document first, and then determined the same α_i (i.e., $\acute{\alpha}$) value to each negative document based on Eq.(2). Therefore, we used the following weighting function to estimate the similarity between a testing document and a given topic:

$$weight(d) = w \cdot d$$

where $\cdot$ means *inner product*; d is the term vector of the testing document; and

$$w = (\sum_{d_i \in D^+} d_i) + (\sum_{d_j \in D^-} d_j \acute{\alpha}).$$

For each topic, we also chose 150 terms in the positive documents, based on *tf*idf* values for all term-based baseline models.

5.2 Evaluation Measures

Precision p and recall r are suitable because the complete classification is based on the positive class. In order to evaluate the effectiveness of the proposed SFD method, we utilized a variety of existing methods; Mean Average Precision (*MAP*),*breakeven points(b/p)*, the precision of *top-20* returned documents, *F-scores* and recall at 11-points (*IAP*). These methods have been widely used to evaluate the performance of information filtering system.

A statistical method, t-test, was also used to analyse the experimental results. The t-test assesses whether the means of two groups are statistically different from each other. If the p-value associated with t is significantly low (<0.05), there is evidence to reject the null hypothesis, and the difference in means across the paired observations is significant.

In summary, the effectiveness is measured by five different means: the average precision of the top 20 documents, F_1 measure, Mean Average Precision (*MAP*), the break-even point (b/p), and Interpolated Average Precision (IAP) on *11-points*. The larger their values are, the better the system performs.

5.3 Results

We compared the proposed method, SFD, with baseline models, including Rocchio, BM25, and SVM. The experimental results for all 50 assessing topics are reported in Table 3, with the percentage changes %*chg*. The percentage changes %*chg* of the proposed SFD model were compared with the performance of the best baseline model (Rocchio). The SFD model outperforms all the baseline models, including the deployment of sequential closed patterns without using the specificity score (Seq. Cls). The average percentage of improvement over the

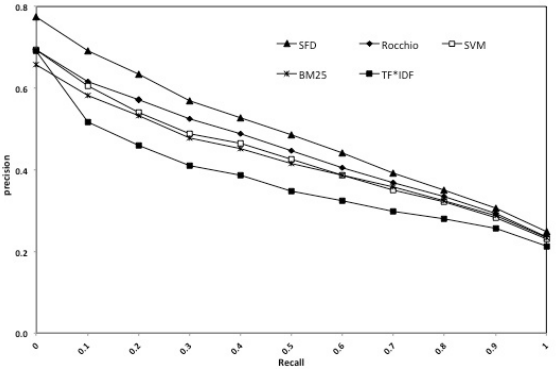

Fig. 1. Comparison of the results in all assessing topics

Table 3. Detailed comparisons of all models in all assessing topics

	top-20	*MAP*	$F_{\beta=1}$	*b/p*	*IAP*
SFD	**0.543**	**0.473**	**0.456**	**0.460**	**0.496**
Seq.Cls*	0.496	0.444	0.439	0.430	0.464
Rocchio	0.474	0.431	0.431	0.420	0.452
SVM	0.453	0.409	0.421	0.408	0.435
BM25	0.445	0.407	0.414	0.407	0.428
%*chg*	+12.71%	+9.05%	+ 5.70%	+8.59%	+8.84%

* Applying Deploying method to sequential closed patterns without weight revision.

standard measures is 8.98%, with a maximum of 12.71% and minimum 5.70% compared with the best results in Table 3.

The improvements are consistent and very significant on all five measures, as shown by *11-points* on all 50 assessing topics in Figure 1. The *t-test p* values in Table 4 indicate the significance of improvements in the SFD model statistically. Therefore, we conclude the SFD model is an exciting achievement in discovering high-quality features in text documents because it uses high-level patterns to get low-level terms and revises low-level terms based on specificity and distributions in positive relevance feedback.

5.4 Discussion

Generally, term-based approaches extract many terms from documents without considering terms relationships. The advantage of using patterns is that they carry more semantic information than single terms–but these suffer from low frequency [10]. Based on that observation, we used the patterns in this paper to consider the relationship among terms to reduce the extracted noise terms in features extracted from documents. As shown in Table 5, the number of extracted patterns is about 202 patterns with an average length of 2 terms in patterns.

Table 4. T-Test p-values for all models compared with the SFD model in all assessing topics

	top-20	*MAP*	$F_{\beta=1}$	*b/p*	*IAP*
Rocchio	0.02621	0.03650	0.05762	0.08629	0.02868
SVM	0.00269	0.00122	0.00658	0.02094	0.00135
BM25	0.00516	0.00424	0.00645	0.03155	0.00206

Table 5. Patterns statistical information for the proposed model

	$\|SP\|$	*Average length of P*	$\|T\|$	sup_r
50 Topics	202	2	156	86.393

Table 6. Statistical information for the proposed model

$\|D^+\|$	*Average No. terms*	*terms weight in topics*	*spe*
13	156	202.788	250.570

From that information, we expected about $404 = 202*2$ terms in all patterns. But the actual number of terms deployed from those patterns are 156 terms, which indicated that about $61.37 = \frac{404-156}{404}$ of the patterns overlap. This overlapping from the closed pattern indicates some terms importance in the documents.

As shown in Table 5, the average weight of patterns for each topic is 86.393 distributed into 202 patterns on average, which gave about $0.428 = \frac{86.393}{202}$. On the other hand, Table 6 shows a weight of about 202.788 distributed in 156 terms on average. That information indicates a weight of about $57.40\% = \frac{202.788-86.393}{202.788}$ is increased according to terms from the deploying method.

Using common sense, we know that positive terms with large *specificity* are more interesting than general terms with less *specificity* for a given topic. However, evaluating the specificity of a given term is challenging. The proposed model calculates the specificity of terms based on the specificity of each term to the topic ST, distribution of terms in topic TFP, and appearance of terms in the documents DF. Unlike other models, in this paper, specific terms appear in most positive patterns in positive documents. As shown in Table 6, before revision, 202.788 was distributed to all positive terms as weights; the *spe* increased the weight of terms to 250.570. The percentage of increase is $19.07\% = \frac{250.570-202.788}{250.570}$. However, the amount of increase differs for each term.

6 Conclusions

It has been proven that pattern-based approaches are useful for improving the quality of feature selection from text documents, although they suffer from low frequency. To solve that problem, deploying methods have been proposed to deploy high-level patterns into low-level terms. However, these deploying methods cause many low-level terms to have the same weight regardless of a terms

specificity. The proposed SFD approach utilizes term distribution in high-level patterns (topics) and terms document frequency to calculate terms specificity according to their appearances and distribution in topics. The experimental results on RCV1 demonstrate that the proposed method has performed excitingly, with an average 8.98% improvement over the state-of-the-art benchmarks.

Acknowledgments. This paper was partially supported by Grant DP0988007 from the Australian Research Council (ARC Discovery Project).

References

1. Aas, K., Eikvil, L.: Text categorisation: A survey. Technical report, Norwegian Computing Center (June 1999)
2. Ahonen, H., Heinonen, O., Klemettinen, M., Verkamo, A.I.: Applying data mining techniques for descriptive phrase extraction in digital document collections. In: Proceedings of the IEEE Forum on Research and Technology Advances in Digital Libraries (ADL 1998), pp. 2–11 (1998)
3. Buckley, C., Voorhees, E.M.: Evaluating evaluation measure stability. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 33–40 (2000)
4. Cao, G., Nie, J.-Y., Gao, J., Robertson, S.: Selecting good expansion terms for pseudo-relevance feedback. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 243–250 (2008)
5. Dumais, S.T.: Improving the retrieval of information from external sources. Behavior Research Methods, Instruments, & Computers 23(2), 229–236 (1991)
6. Jindal, N., Liu, B.: Identifying comparative sentences in text documents. In: Proceedings of SIGIR 2006, pp. 244–251 (2006)
7. Joachims, T.: A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning, pp. 143–151. Morgan Kaufmann Publishers Inc. (1997)
8. Lewis, D.D.: An evaluation of phrasal and clustered representations on a text categorization task. In: Proceedings of SIGIR 1992, pp. 37–50 (1992)
9. Li, X., Liu, B.: Learning to classify texts using positive and unlabelled data. In: Proceedings of IJCAI 2003, pp. 587–594 (2003)
10. Li, Y., Algarni, A., Zhong, N.: Mining positive and negative patterns for relevance feature discovery. Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 753–762 (2010)
11. Li, Y., Zhong, N.: Mining ontology for automatically acquiring web user information needs. IEEE Transactions on Knowledge and Data Engineering 18(4), 554–568 (2006)
12. Li, Y., Zhou, X., Bruza, P., Xu, Y., Lau, R.Y.: A two-stage text mining model for information filtering. In: Proceeding of the 17th ACM Conference on Information and Knowledge Management, pp. 1023–1032 (2008)
13. Ling, X., Mei, Q., Zhai, C., Schatz, B.: Mining multi-faceted overviews of arbitrary topics in a text collection. In: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 497–505 (2008)

14. Metzler, D., Croft, W.B.: Latent concept expansion using markov random fields. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 311–318 (2007)
15. Pon, R.K., Cardenas, A.F., Buttler, D., Critchlow, T.: Tracking multiple topics for finding interesting articles. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 560–569 (2007)
16. Robertson, S.E., Soboroff, I.: The trec 2002 filtering track report. In: Proceedings of TREC (2002)
17. Salton, G.: The SMART Retrieval System-Experiments in Automatic Document Processing. Prentice-Hall, Inc., Upper Saddle River (1971)
18. Scott, S., Matwin, S.: Feature engineering for text classification. In: The 16th International Conference on Machine Learning, pp. 379–388 (1999)
19. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. 34(1), 1–47 (2002)
20. Shen, D., Sun, J.-T., Yang, Q., Zhao, H., Chen, Z.: Text classification improved through automatically extracted sequences. In: Proceedings of the 22nd International Conference on Data Engineering, pp. 121–123. IEEE Computer Society (2006)
21. Soboroff, I., Robertson, S.: Building a filtering test collection for trec 2002. In: Proceedings of SIGIR 2003, pp. 243–250 (2003)
22. Wang, X., Fang, H., Zhai, C.: A study of methods for negative relevance feedback. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 219–226 (2008)
23. Wu, S.-T., Li, Y., Xu, Y.: Deploying approaches for pattern refinement in text mining. In: Proceedings of ICDM 2006, pp. 1157–1161 (2006)
24. Wu, S.-T., Li, Y., Xu, Y., Pham, B., Chen, P.: Automatic pattern-taxonomy extraction for web mining. In: Proceedings of WI 2004, pp. 242–248 (2004)
25. Xu, Y., Li, Y.: Generating concise association rules. In: Proceedings of CIKM 2007, pp. 781–790 (2007)
26. Xu, Z., Akella, R.: Active relevance feedback for difficult queries. In: Proceeding of the 17th ACM Conference on Information and Knowledge Management, pp. 459–468 (2008)
27. Yan, X., Cheng, H., Han, J., Xin, D.: Summarizing itemset patterns: a profile-based approach. In: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 314–323 (2005)
28. Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 42–49 (1999)