SAT-MOD: Moderate Itemset Fittest for Text Classification

Jianlin Feng
Dept. of Computer Science
Huazhong Univ. of Sci. & Tech.
Wuhan 430074, Hubei, China
86-27-87544375
fengjl@mail.hust.edu.cn

Huijun Liu
Dept. of Computer Science
Huazhong Univ. of Sci. & Tech.
Wuhan 430074, Hubei, China
86-27-87544375
HuijunLiu@mail.hust.edu.cn

Jing Zou
Dept. of Computer Science
Huazhong Univ. of Sci. & Tech.
Wuhan 430074, Hubei, China
86-27-87544375
zoujing1003@163.com

ABSTRACT
In this paper, we present a novel association-based method called *SAT-MOD* for text classification. SAT-MOD views a sentence rather than a document as a transaction, and uses a novel heuristic called *MODFIT* to select the most significant itemsets for constructing a category classifier. The effectiveness of SAT-MOD has been demonstrated comparable to well-known alternatives such as LinearSVM and much better than current document-level words association based methods on the Reuters corpus.

Categories and Subject Descriptors
I.7.M [**Computing Methodologies**]: Document and Text Processing – *Miscellaneous.*

General Terms
Algorithms, Performance.

Keywords
Text Classification, MODFIT (Moderate Itemset Fittest) Heuristic.

1. INTRODUCTION
Text classification (TC) is to realize the task of assigning one or more (*multi-labeled*) predefined category labels to unlabeled natural language text documents based on their content. TC has become more and more important due to the flourishing of digital documents over the Internet and intranets. It has extensive applications in online news classification, email filtering, and the like. Many methods have been proposed for TC, including Naïve Bayes, decision trees, *k*-NN, LinearSVM and *association rule* based (or simply *association* based) methods [1-5].

An association rule for TC is indeed similar to an IF-THEN rule manually defined by domain experts in the *knowledge engineering* method, which is the most popular real-world approach to TC in the 1980s. To the best of our knowledge, all the current association based text classification and clustering methods all exploit document-level co-occurring words (*itemsets*), which are a group of words co-occurring in the same document. Two ARC algorithms are proposed in [3], both viewed a document as a single transaction and used the traditional *database coverage* heuristic for selecting significant itemsets. Document-level frequent itemsets are also exploited for text clustering, e.g. the FIHC algorithm [8]. The very recently proposed eMailSift [4] also takes each mail instead of a sentence in the email as a transaction.

WWW 2005, May 10--14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

However, the basic semantic unit in a document is actually a sentence. Words co-occurring in the same sentence are usually associated in one way or the other, and are more meaningful than the same group of words spanning several sentences in a document. Hence we view a sentence rather than a document as the basic semantic unit and present a novel association-based TC method called *SAT-MOD*.

2. MINING CO-OCCURRING WORDS
2.1 Document Frequent Itemset
In daily life, usually people are liable to emphasize some core ideas by repeating some *representative words* in different sentences, thus frequently repeated words tend to represent a facet of the whole "*document subject*" of a document. Those *content* words are captured by *Document Frequent Itemsets* (abbr. DFIs) in our SAT-MOD method. A DFI is a group of words co-occurring in a minimum number (*document minsup*) of sentences in a (*supporting*) document. The supporting document is said to be *covered* by the DFI. With each word as an item, and each natural sentence as a transaction, we can use frequent itemsets mining algorithm such as the classical Apriori [6] to mine DFIs in a document, and represent each training document as a set of DFIs.

Naturally, document minsup should be set to guarantee that a DFI occurs in at least 2 sentences. Hence a document minsup of value 2 is called the *natural document minsup*.

2.2 Mining Contexts
As argued in [5], although most content words are much more likely to occur again in a document once they have occurred once, in many cases, the probability of reusing a content word immediately after its first occurrence is lower than general since we are taught to avoid repetitive writing. Usually authors may alternately use synonyms to avoid dull repeat. However we believe some content words, especially proper nouns such as our DFI, do not avoid direct repeat. Thus a compromise could be made that content words would repeat in near paragraphs, and we can use a sliding window to construct different mining contexts. For simplicity, a mining context is respectively called a *unit*, a *multi*, and a *full* mining context when the sliding window size p is accordingly set to be 1, k ($1< k <P$), and P, here P is the total number of paragraphs in the document. Given a mining context, content words can be captured by itemsets which are document frequent in that context.

3. SAT-MOD
DFIs are then used to generate *Category Frequent Itemsets* (abbr. CFIs). A CFI with respect to a pre-defined category is an itemset whose *category support* (the number of supporting documents in

that category, provided that the CFI is a DFI in each supporting document) is no less than a user-specified minimum number (*category minsup*). All the CFIs are collected using a *category prefix-tree*, and the tree is then pruned by our novel heuristic called MODFIT, i.e. the ***moderate itemset fittest*** intuition as follows: 1) *Intuitively an itemset is usually harder to appear than its proper subsets in a sentence and also harder to be document frequent in a document, hence an itemset tends to have more discriminating power than its proper subsets*; 2) *On the other hand, a too long itemset may cause overfitting and hence lose its discriminating power for unlabeled documents.*

Based on MODFIT, we should seek a definition which can make a moderate itemset have a classification *confidence* greater than any of its underfitting proper subsets or overfitting proper supersets.

Definition 1 (Confidence of a CFI w.r.t a category C_i) The confidence, denoted as $Conf(I \Longrightarrow C_i)$, is defined as the ratio of S_i to S_{tot}, i.e. $Conf(I \Longrightarrow C_i) = S_i / S_{tot}$, where S_i is the category support of I in C_i, and S_{tot} is the total number of *distinct* supporting documents covered by I in the whole training set.

Intuitively, the MODFIT heuristic equals to moderately extending a single word with other words along a natural sentence. Using MODFIT pruning, we will keep all *synonymic* itemsets that only partly share some items with each other. In addition, we do mot need the very expensive step of removing covered documents in the database coverage heuristic. The pruned tree is finally taken as the category classifier. Figure 1 is just an illustration of a category prefix-tree where each node contains three counters: I_c, I_{conf} and I_s. We use I_c and I_{conf} to respectively hold category support and confidence (i.e, $Conf(I \Longrightarrow C)$) of the itemset I which corresponds to the host node. The counter I_s is for holding document support of I in an unlabeled document D_u in subsequent classifying phase.

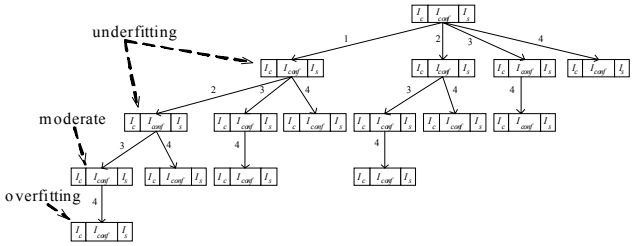

Figure 1. A category prefix-tree for items 1, 2, 3, and 4.

Given a set of predefined categories $\{C_1, C_2, \ldots, C_M\}$, and a document D_u to be classified, we first need to identify all the "common itemsets" shared between each category C_i ($1 <= i <= M$) and D_u (i.e. *category intersection* $C_i \bigcap D_u$), and count their supports in D_u. The category similarity is then derived as follows:

$$sim(C_i \leftarrow D_u) = \sum (I_{conf} \times I_s) \times [I \in C_i \bigcap D_u]$$

where $[I \in C_i \bigcap D_u]$ has a value of 1 if I is truly an itemset in the intersection $(C_i \bigcap D_u)$ and otherwise 0. We then realize the multi-labeled classification task using a classifying strategy as follows: First assign D_u with the category label of the category having the highest similarity score; Second specify a threshold called *label minsup* in percentage, and if any other category has a similarity score greater than that percentage of the highest score, then D_u is also assigned the label of that category.

We studied how crucial parameters would affect the effectiveness of SAT-MOD, and also made a comparison with other alternatives including *SAT-FOIL* (using the FOIL heuristic[7], i.e. database coverage, instead of MODFIT) in terms of *classification accuracy*. So far, we have only considered exploiting itemsets with a maximum size of 5 which seems already enough.

Our experimental study shows that the *natural document minsup* can be a quite reasonable choice to capture sentential word co-occurrence. The underlying reason is consistent with MODFIT pruning, because natural document minsup can keep more synonyms. Since natural document minsup is actually a bottom bound of document minsup, we can use it as a default and hence logically remove this parameter. The study of mining contexts provides a proof of previous assertions on the clumping of content words: usually content words are more liable to be repeated in different sentences distributed in different paragraphs, but the distribution is not very regular, hence it is better to use the whole document as the mining context. Currently we have got very encouraging classification results (referring to Table 1) on relatively short documents such as the Reuters, the measures of other well-known methods are obtained from [2, 3].

Table 1. BEP on 10 largest categories of Reuters

BEP	SAT-MOD labelMinsup		SAT-FOIL labelMinsup		ARC-BC δ=50		Bayes	Rocchio	C4.5	k-NN	LinearSVM
	65%	75%	70%	80%	10%	15%					
acq	93.7	**95.1**	91.8	**92.7**	90.9	89.9	91.5	92.1	85.3	92.0	93.6
corn	**77.1**	71.2	**70.7**	69.4	69.6	82.3	47.3	62.2	87.7	77.9	90.3
crude	**90.9**	90.6	90.1	**90.5**	77.9	77.0	81.0	81.5	75.5	85.7	88.9
earn	97.0	**97.4**	95.0	**96.2**	92.8	89.2	95.9	96.1	96.1	97.3	98.0
grain	90.6	**91.3**	**89.3**	86.4	68.8	72.1	72.5	79.5	89.1	82.2	94.6
interest	**75.1**	74.9	**78.9**	76.3	70.5	70.1	58.0	72.5	49.1	74.0	77.7
money-fx	85.3	**86.6**	**84.0**	82.7	70.5	72.4	62.9	67.6	69.4	78.2	74.5
ship	**86.9**	83.6	**83.1**	80.9	73.6	73.2	78.7	83.1	80.9	79.2	85.6
trade	82.6	**84.9**	**86.9**	86.7	68.0	69.7	50.0	77.4	59.2	77.4	75.9
wheat	**79.1**	75.2	**72.7**	70.4	84.8	86.5	60.6	79.4	85.5	76.6	91.8
micro-avg	91.7	**92.2**	90.1	**90.3**	82.1	81.8	72.0	79.9	79.4	82.3	92.0
macro-avg	**85.8**	85.1	**84.3**	83.2	76.74	78.24	65.21	79.14	77.78	82.05	87.10

4. CONCLUTIONS

We have proposed the SAT-MOD exploiting a novel MODFIT heuristic, which has very promising classification accuracy on relatively short documents such as the Reuters. In addition, it has inherent readability and refinability of acquired classification rules.

5. ACKNOWLEDGMENTS

This work was supported by China NSF grant No. 60303030 and Chongqing NSF research grant No. 8721.

6. REFERENCES

[1] F. Sebastiani. Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, 34(1): 1-47, 2002.

[2] S. Dumais, et al. Inductive Learning Algorithms and Representations for Text Categorization. *CIKM98*, 148-155.

[3] M. Antonie and O. R. Zaiane. Text Document Categorization by Term Association. *IEEE ICDM02*, 19-26.

[4] M. Aery, S. Chakravarthy eMailSift: Mining-based Approaches to Email Classification. *ACM SIGIR*04, 580-581.

[5] C. D. Manning and H. Schutze. Foundations of Statistical Natural Language Processing. The MIT Press, 1999.

[6] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules. *VLDB94*, 487-499.

[7] J. R. Quinlan ,et al. FOIL: A Midterm Report. *ECML93*, 3-20.

[8] B. C. M. Fung, et al. Hierarchical Document Clustering Using Frequent Itemsets. *SIAM ICDM*, 2003.