A Rough Set Approach to Classifying Web Page Without Negative Examples

Qiguo Duan, Duoqian Miao, and Kaimin Jin

Department of Computer Science and Technology, Tongji University, Shanghai, 201804,China
The Key Laboratory of "Embedded System and Service Computing", Ministry of Education, Shanghai, 201804,China
dqgcn@126.com,miaoduoqian@163.com,jinkaimin@163.com

Abstract. This paper studies the problem of building Web page classifiers using positive and unlabeled examples, and proposes a more principled technique to solving the problem based on tolerance rough set and Support Vector Machine (SVM). It uses tolerance classes to approximate concepts existed in Web pages and enrich the representation of Web pages, draws an initial approximation of negative example. It then iteratively runs SVM to build classifier which maximizes margins to progressively improve the approximation of negative example. Thus, the class boundary eventually converges to the true boundary of the positive class in the feature space. Experimental results show that the novel method outperforms existing methods significantly.

Keywords: Web page classification, rough set, Support Vector Machine.

1 Introduction

With the rapid growth of information on the World Wide Web, automatic classification of Web pages has become important for effective retrieval of Web documents. The common approach to building a Web page classifier is to manually label some set of Web page to pre-defined categories or classes, and then use a learning algorithm to produce a classifier. The main bottleneck of building such a classifier is that a large number of labeled training Web page is needed to build accurate classifiers. In most cases of automatic Web page classification, it is normally easy and inexpensive to collect positive and unlabeled examples, however, arduous and very time consuming to collect negative training examples and label them by user's own hands.

In this paper, we focus on the problem to classifying Web page with positive and unlabeled data and without labeled negative data. Recently, a few techniques for solving this problem were proposed in the literature. Liu et al. proposed a method (called S-EM) to solve the problem in the text domain [7]. In [8], Yu et al. proposed a technique (called PEBL) to classify Web pages given

Z.-H. Zhou, H. Li, and Q. Yang (Eds.): PAKDD 2007, LNAI 4426, pp. 481–488, 2007.

positive and unlabeled pages. This paper proposes a more effective and robust technique to solve the problem. Experimental results show that the new method outperforms existing methods significantly. Throughout the paper, we call the class of Web page that we are interested in positive and the complement set of samples negative.

The rest of the paper is organized as follows: Section 2 presents the concepts of the tolerance rough set briefly. Section 3 describes proposed technique. Section 4 reports and discusses the experimental results. Finally, Section 5 concludes the paper.

2 Tolerance Rough Set

Rough set theory is a formal mathematical tool to deal with incomplete or imprecise information [2]. The classical rough set theory is based on equivalence relation that divides the universe of objects into disjoint classes. By relaxing the equivalence relation to a tolerance relation, where transitivity property is not required, a generalized tolerance space is introduced below [3],[4],[5],[6].

Let $I : U \rightarrow P(U)$ to denote a tolerance relation, if and only if $x \in I(x)$ for $x \in U$ and $y \in I(x) \Leftrightarrow x \in I(y)$ for any $x, y \in U$, where $P(U)$ are sets of all subsets of U. Thus the relation $xIy \Leftrightarrow y \in I(x)$ is a tolerance relation (i.e. reflexive, symmetric) and $I(x)$ is a tolerance class of x. Define the tolerance rough membership function $\mu_{I,V}$, as $x \in U, X \subseteq U$,

$$\mu_{I,V}(x, X) = \nu(I(x), X) = \frac{|I(x) \bigcap X|}{|I(x)|} . \tag{1}$$

The tolerance rough set for any $X \subseteq U$ are then defined as

$$L_R(X) = \{x \in U | \nu(I(x), X) = 1\} . \tag{2}$$

$$U_R(X) = \{x \in U | \nu(I(x), X) > 1\} . \tag{3}$$

With its ability to deal with vagueness and fuzziness, tolerance rough set seems to be promising tool to model relations between terms and documents. The application of tolerance rough set in classifying Web page using positive and unlabeled examples was proposed as a way to enrich feature and document representation and extract reliable negative examples for improvement of classification.

2.1 Tolerance Space of Terms in Unlabeled Set

Let $U = \{d_1, ..., d_M\}$ be a set of unlabeled Web pages and $T = \{t_1, ..., t_N\}$ set of terms for U. The tolerance space is defined over a universe of all terms for U. The idea of terms expansion is to capture conceptually related terms into classes. For

this purpose, the tolerance relation is determined as the co-occurrence of terms in all Web pages from U.

2.2 Tolerance Class of Term

Let $f_U(t_i, t_j)$ denotes the number of Web pages in U in which both terms t_i and t_j occurs. The uncertainty function I with regards to co-occurrence threshold θ defined as

$$I_\theta(t_i) = \{t_j | f_U(t_i, t_j) \geq \theta\} \cup \{t_i\} \ . \tag{4}$$

Clearly, the above function satisfies conditions of being reflexive: $t_i \in I_\theta(t_j)$ and symmetric: $t_j \in I_\theta(t_i) \Leftrightarrow t_i \in I_\theta(t_j)$ for any $t_i, t_j \in T$. Thus, $I_\theta(t_i)$ is the tolerance class of term t_i.Tolerance class of terms is generated to capture conceptually related terms into classes. The degree of correlation of terms in tolerance classes can be controlled by varying the threshold θ.The membership function μ for $t_i \in T, X \subseteq T$ is then defined as:

$$\mu(t_i, X) = \nu(I_\theta(t_i), X) = \frac{|I_\theta(t_i) \cap X|}{|I_\theta(t_i)|} \ . \tag{5}$$

Finally, the lower and upper approximations of any subset $X \subseteq T$ can be determined with the obtained tolerance relation respectively as [5],[6]:

$$L_R(X) = \{t_i \in T | \nu(I_\theta, X) = 1\} \ . \tag{6}$$

$$U_R(X) = \{t_i \in T | \nu(I_\theta, X) > 0\} \ . \tag{7}$$

2.3 Expansion the Web Pages on Tolerance Class of Term

In tolerance space of term, an expanded representation of Web document can be acquired by representing Web document as set of tolerance classes of terms it contains. This can be achieved by simply representing Web document with its upper approximation, e.g., the Web page $d_i \in U$ is represented by:

$$U_R(d_i) = \{t_i \in T | \nu(I_\theta(t_i), d_i) > 0\} \ . \tag{8}$$

The usage of tolerance space and upper approximation to enrich Web page and term relation allows the proposed technique to discover subtle similarities between positive examples in positive set and latent positive examples in unlabeled set.

3 The TRS-SVM Algorithm

We use TRS-SVM to denote the proposed classification techniques that employ the method based on tolerance rough set to extract reliable negative set and SVM to build classifier. The TRS-SVM algorithm is composed by following steps:

Step1: Preprocessing of Web page in set P and U.

A preprocessing procedure is done as follows: Remove the HTML tag and extract plain text from each Web page. All the extracted words are stemmed. Use a stop list to omit the most common words. Finally, extract term set from positive set P and unlabeled set U respectively, let PT be a term set for P and UT a term set for U.

Step2: Positive feature selection.

This step builds a positive feature set PF which contains terms that occur in the term set PT more frequently than in the term set UT. The decision threshold σ is normally set to 1 but can be adjusted. Here $freq(t_i, X)$ denotes the number of occurrence of term t_i in set X and $|X|$ denotes the total number of Web pages in set X.The detail algorithm is given as follows.

1. Generating the set $\{t_1, \cdots, t_n\}, t_i \in UT \cup PT$;
2. $PF = \emptyset$;
3. For $i = 0$ to n
4. $\quad f_p^i = freq(t_i, P)/|P|, f_u^i = freq(t_i, U)/|U|$;
5. $\quad$ If $f_p^i / f_u^i > \sigma$ then $PF = PF \cup \{t_i\}$;
6. $\quad$ End If
7. End For

Step3: Generating tolerance class of term in unlabeled set and enriching Web page representation.

The goal of this step is to determine for each term in UT, the tolerance class which contains its related terms with regards to the tolerance relation. In our experiment we set $\theta = 7$ for good result. Then, the Web page in unlabeled set is represented with its upper approximation, e.g. the Web page $d \in U$ is represented by $U_R(d)$.

Step4: Expansion the positive feature set on tolerance class of term.

The tolerance class of term in unlabeled set which contains the positive feature term in PF will be merged with PF.The algorithm is given as follows.

1. For each $t_i \in PF \cap UT$;
2. $\quad PF = PF \cup I_\theta(t_i)$;
3. End For

Step5: Generating reliable negative set.

This step tries to filter out possible positive Web pages from U. A Web page in U which upper approximation does not have any positive feature in PF is regarded as a reliable negative example. The algorithm is given as follows.

1. $RN = U$;
2. For each Web page $d \in U$;
3. $\quad$ If $\exists x_j freq(x_j, U_R(d)) > 0$ and $x_j \in PF$ then $RN = RN - d$;
4. $\quad$ End If
5. End For

Step6: building classifier.

This step builds the final classifier by running SVM iteratively with the sets P and RN. The basic idea is to use each iteration of SVM to extract more possible negative data from $U - RN$ and put them in RN. Let Q be the set of remaining unlabeled Web pages, $Q = U - RN$. The algorithm for this step is given as follows.

1. Every Web page in P is assigned the class label +1;
2. Every Web page in RN is assigned the label -1;
3. $i = 1, Pr_0 = 0$;
4. Loop
5. Use P and RN to train a SVM classifier C_i;
6. Classify Q using C_i;
   Let the set of Web pages in Q that are classified as negative be W;
7. Classify positive set P with C_i;
   Set Pr_i as classification precision of P;
8. If $(|W| = 0 || Pr_i < Pr_{i-1})$
   then store the final SVM classifier, exit loop;
9. else $Q = Q - W$;
   $RN = RN \cup W$;
   $i = i + 1$;
10. End If
11. End Loop

The reason that we run SVM iteratively is that the reliable negative set RN extracted by the method based on tolerance rough set may not be sufficiently large to build the best classifier. SVM classifiers can be used to iteratively extract more negative Web pages from Q.There is, however, a danger in running SVM iteratively. Since SVM is very sensitive to noise, if some iteration of SVM goes wrong and extracts many positive Web pages from Q and put them in the negative set RN, then the last SVM classifier will be extremely poor. This is the problem with PEBL, which also runs SVM iteratively. In our algorithm, the iteration stops when there is no negative Web page that can be extracted from Q or the classification precision decreases which indicates that SVM has gone wrong.

4 Experimental Evaluation

4.1 Experiment Datasets

To evaluate the proposed techniques, we use the WebKB data set[1], which contains 8282 Web pages collected from computer science departments of various universities. The pages were manually classified into the following categories: student, faculty, staff, department, course, project, other. In our experiments,

[1] http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/

we used only the four most common categories: student, faculty, course, other (respectively abbreviated here as St, Fa, Co, Ot). Each category is employed as the positive class, and the rest of the categories as the negative class. This gives us four datasets. Our task is to identify positive Web pages from the unlabeled set. The construction of each dataset for our experiments is done as follows: Firstly, we randomly select 10% of the Web pages from the positive class and the negative class, and put them into test set to evaluate the performance of classifier. Then, the rest are used to create training sets. For each dataset, a% of the Web pages from the positive class is randomly selected as the positive set P. The rest of the positive Web pages and negative Web pages form the unlabeled set U. Our training set consists of P and U. In our experiments, we range from 10%-70% respectively to create a wide range of settings.

4.2 Performance Measures

To analyze the performance of classification, we adopt the popular F1 measure on the positive class. F1 measure is combination of recall (Re) and precision (Pr), F1=2.Re.Pr/(Re+Pr). Precision means the rate of documents classified correctly among the result of classifier and recall signifies the rate of correct classified documents among them to be classified correctly. The F1 measure which is the harmonic mean of precision and recall is used in this study since it takes into account effects of both quantities.

4.3 Experimental Results and Discussion

We now present the experimental results. For comparison, we include the classification results of the naive Bayesian method (NB)[1], S-EM, OSVM [9] and PEBL. Here, NB treats all the Web pages in the unlabeled set as negative. For SVM implementation, we used the LIBSVM[2]. We set Gaussian kernel as default kernel function of SVM because of its high accuracy. PEBL and OSVM also used LIBSVM. We set $\theta = 7$ for good result in generating tolerance class.

We summarize the average F value results of all a settings in Figure 1. We observe that TRS-SVM outperforms NB, S-EM, OSVM and PEBL. In fact, PEBL performs poorly when the number of positive Web pages is small. When the number of positive Web pages is large, it usually performs well. TRS-SVM performs well consistently. We also ran SVM with positive set and unlabeled set. It for the noisy situation (unlabeled set U as negative set) performs poorly (its F values are mostly close to 0) because SVM does not tolerate noise well. Due to space limitations, its results are not listed.

From Figure 1, we can draw the following conclusions: OSVM gives very poor results (in many cases, F value is around 0.3-0.5). PEBL's results are extremely poor when the number of positive Web pages is small. We believe that this is because its strategy of extracting the initial set of reliable negative Web pages could easily go wrong without sufficient positive data. S-EM's results are worse than TRS-SVM. The reason is that the negative Web pages extracted from U by

[2] http://www.csie.ntu.edu.tw/∼ cjlin/libsvm/

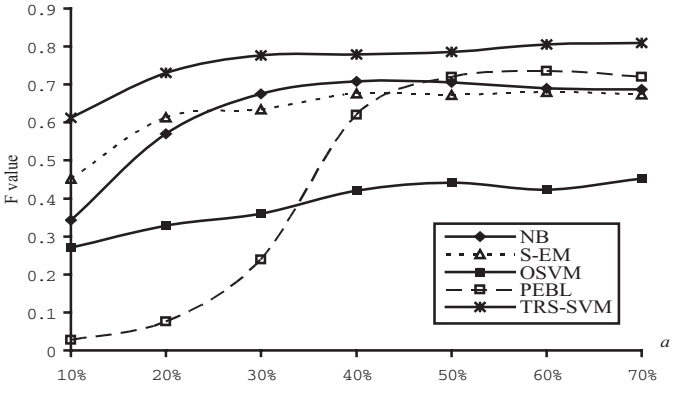

Fig. 1. Average results for all a settings

its spy technique are not reliable. We observe that a single NB slightly outperforms S-EM. TRS-SVM performs well with different numbers of positive Web pages.

Sensitiveness to co-occurrence threshold parameter: Co-occurrence threshold parameter θ is rather important to our TRS-SVM. From definition of tolerance class it is not difficult to get such deduction that inadequate co-occurrence threshold can decrease the performance of the classification results: on one hand, too small co-occurrence threshold can make too many negative examples be extracted as positive examples, on the other hand, too large co-occurrence threshold can make too little latent positive examples be identified from U, both cases can lead to worse performance.

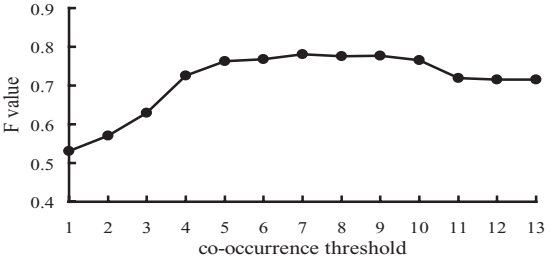

Fig. 2. Sensitiveness to co-occurrence threshold

From Figure 2 we can understand our experimental result corresponds to our deduction: when co-occurrence threshold equals value between 5 and 10, the performance is better, however, when it is out of the interval, the performance is worse (here, a=60% and for other a values, the results are similar).

5 Conclusions

This paper studied the problem of Web page classification with only partial information, i.e., with only one class of labeled Web pages and a set of unlabeled Web pages. An effective technique is proposed to solve the problem. Our algorithm first utilizes the method based on tolerance rough set to extract a set of reliable negative Web pages from the unlabeled set, and then builds a SVM classifier iteratively. The experiment we have carried has showed that the method based on tolerance rough set it offers can extract reliable negative examples by discovering subtle information among unlabeled data, which have positive effects on classification quality. Experimental results show that the proposed technique is superior to S-EM and PEBL.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (No.60475019) and the Ph.D. programs Foundation of Ministry of Education of China (No.20060247039).

References

1. Lewis, D., Ringuette, M.: A Comparison of Two Learning Algorithms for Text Categorization. Third annual symposium on document analysis and information retrieval (1994) 81-93
2. Pawlak, Z.: Rough sets: Theoretical Aspects of Reasoning about Data. Kluwer Dordrecht (1991)
3. Skowron, A., Stepaniuk, J.: Tolerance approximation spaces. Fundamenta Informaticae 27 (1996) 245-253
4. Kryszkiewicz, M.: Rough set approach to incomplete information system. Information Sciences, (1998)112:39-49
5. Tu Bao Ho, Ngoc Binh Nguyen: Nonhierarchical Document Clustering based on A Tolerance Tough Set Model. International Journal of Intelligent Systems, Vol. 17 (2002) 199-212
6. Ngo Chi Lang: A Tolerance Rough Set Approach to Clustering Web Search Results. In: J.-F. Boulicaut et al. (eds.): PKDD 2004. Springer-Verlag, Berlin Heidelberg (2004) 515-517
7. Liu, B., Lee, W. S., Yu, P., and Li, X.: Partially Supervised Classification of Text Documents. ICML-02 (2002)
8. H. Yu, J. Han, and K.C.-C. Chang: PEBL: Web Page Classification without Negative Examples. IEEE Transactions on Knowledge and Data Engineering, Vol. 16, No. 1, 1 (2004) 70-81
9. L.M. Manevitz and M. Yousef: One-Class SVMs for Document Classification. J. Machine Learning Research, vol. 2 (2001) 139-154