Webology, Volume 5, Number 1, March, 2008

Generating best features for web page classification

M. Indra Devi
Department of Information Technology, Thiagarajar College of Engineering, Madurai 15, India. Email: midit (at) tce.edu

R. Rajaram
Dean, Computer Science & Information Technology, Thiagarajar College of Engineering, Madurai 15, India. Email: rrajaram (at) tce.edu

K. Selvakuberan
Assistant System Engineer, Innovation Labs (Web 2.0), TATA Consultancy Services, ETL Infrastructures Services Ltd., SEZ, 200 ft, Thoraippakkam, Pallavaram Ring Road, Chennai-600096, India. Email selvakuberan.k (at) tcs.com

Received March 5, 2008; Accepted March 30, 2008

Abstract

As the Internet provides millions of web pages for each and every search term, getting interesting and required results quickly from the Web becomes very difficult. Automatic classification of web pages into relevant categories is the current research topic which helps the search engine to get relevant results. As the web pages contain many irrelevant, infrequent and stop words that reduce the performance of the classifier, extracting or selecting representative features from the web page is an essential pre-processing step. The goal of this paper is to find minimum number of highly qualitative features by integrating feature selection techniques. We conducted experiments with various numbers of features selected by different feature selection algorithms on a well defined initial set of features and show that cfssubset evaluator combined with term frequency method gives minimal qualitative features enough to attain considerable classification accuracy.

Keywords

Feature selection; Subset generation; Subset evaluation; Machine learning; Classification

---

Introduction

Over the past decade we have witnessed an explosive growth on the Internet, with millions of web pages on every topic easily accessible through the Web. The Internet is a powerful medium for communication between computers and for accessing online documents all over the world but it is not a tool for locating or organizing the mass of information. Tools like search engines assist users in locating information on the Internet. They perform excellently in locating but provide limited ability in organizing the web pages. Internet users are now confronted with thousands of web pages returned by a search engine using

simple keyword search. Searching through those web pages is in itself becoming impossible for users (Daume & Brill, 2004). Thus, it has been of more interest in tools that can help make a relevant and quick selection of information that we are seeking. On October 2007, 31 billion searches were conducted on Google alone, more than one billion queries each day (Lipsman, 2007).

The number of web pages indexed by Google and Yahoo search engines is in billions. In 2005, Yahoo claimed that its index covered more than 20 billion web resources, the largest search engine (Terdiman, 2005). It is believed that the actual size of the Web is at least several times bigger than what search engines currently cover. Describing and organizing this vast amount of content is essential for realizing the Web's full potential as an information resource. Hence the automation of web page classification is necessary. For the web page classification problem, only representative features from a web page are considered and how to choose these representative features is a research problem of current interest. There are so many feature selection or feature extraction algorithms exist. But by applying any one of the existing algorithms, it is found that the number of representative features remains high. To reduce and to get best features, combination of more than one feature selection algorithm is suggested (Lu, Sung & Lu, 1996; Salzberg, 1992).

Web pages contain many irrelevant words than relevant words. So many stop words, punctuation symbols and rare words may present in the web page. Some web pages contain audio, image and/or video information associated with them. Not all the web pages are of uniform size. The web pages are dynamic and volatile in nature. There is no unique format for the web page. Some web pages may be unstructured (full text), some pages may be semi structured (HTML pages) and some pages may be structured (Databases). To find the category of the web page, the textual information in the web page serves as a hint.

Selecting representative features from the web page reduces the time and resource requirement to find the category of the web page. This also reduces the dimensionality. Machine learning classifiers are used to classify the web pages in our approach. Machine learning classifiers automate the process and they are able to learn with experience. So with time and experience more accurate predictions can be got. Support Vector Machine classifiers and Naïve Bayes classifiers are used to evaluate the quality of the features selected using our approach.

Related work

Liu and Setino (1996) propose an algorithm called Inductive Algorithm. It is a probabilistic wrapper model besides the exhaustic search and heuristic approach.

Liu and Yu (2005) proposed that feature selection algorithms for classification and clustering, groups and compares different algorithms with a categorizing framework based on search strategies, evaluation criteria, and data mining tasks, reveals unattempted combinations, and provides guidelines in selecting feature selection algorithms. With the categorizing framework, an integrated system for intelligent feature selection is built up. For employing feature selection, wrapper, filter and hybrid model are adopted.

Combarro et al. (2005) suggested that to select the relevant features by a family of linear filtering approaches. The feature selection approaches are bag of words representation, filter and wrapper approaches, term frequency, document frequency, inverted document frequency and the Information Gain indicates the presence of word in the category or not. The distribution of documents over the categories is considered by introducing the concept of canonical or unconditional rule which says that any document belongs to the category. This rule is used as a reference for the rest of rules of the same category. SVM (Support

Vector Machine) classifier is employed with Reuters 21578 collections as the experimental data.

A feature set is constructed per category and classify into positive and negative examples. Zheng, Srihari and Srihari (2003) proposed a new method that combines positive and negative examples. A comparison is made by using chi-square, correlation coefficient, odds ratio, GSS coefficient and two proposed variants of odds ratio and GSS coefficient: OR-square and GSS-square respectively. The results show that the proposed feature selection method improves text filtering performance.

Kim and Zhang (2003) present a machine learning approach to mine the structure of HTML documents for effective Web-document retrieval. A genetic algorithm is described that learns the importance factors of HTML tags which are used to re-rank the documents retrieved by standard weighting schemes. The proposed method has been evaluated on artificial text sets and a large-scale TREC document collection and proved that it significantly improves the performance in retrieval accuracy. And also they demonstrated the use of the document-structure mining approach tends to move relevant documents to upper ranks, which is especially important in interactive Web-information retrieval environments.

Borkar, Deshmukh and Sarawagi (2001) presented a method for automatically segmenting unformatted text records into structured elements. Several useful data sources today are human-generated as continuous text whereas convenient usage requires the data to be organized as structured records. Their prime motivation is the warehouse address cleaning problem of transforming dirty addresses stored in large corporate databases as a single text field into subfields like "City" and "Street".

Feature Selection

Feature extraction or selection is one of the most important pre-processing steps in pattern recognition or pattern classification, data mining, machine learning and so on. It is also an effective dimensionality reduction technique and an essential preprocessing method to remove noise features. The basic idea of feature selection algorithms is searching through all possible combinations of features in the data to find which subset of features works best for prediction. The selection is done by reducing the number of features of the feature vectors, keeping the most meaningful discriminating ones, and removing the irrelevant or redundant ones (Krishnapuram et al., 2004; Chen & Liu, 1999; Vafaie & De Jong, 1993; Yan & Pederson, 1997). During generation and evaluation of subsets of features increasing feature brings disadvantages for classification problem.

Issues in Feature Selection

On the one hand, feature increased gives difficulties to calculate, because the more data occupy amount of memory space and computerization time, on the other hand, a lot of features include certainly many correlation factors respectively, which results to information repeat and waste. It is necessary to take measures to decrease the feature dimension under not decreasing recognition effect; this is called the problems of feature optimum extraction or selection (Yan & Pederson, 1997). The characteristics of good features should be simple, moderate, less redundancy and unambiguous.

Proposed approach

The goal of this paper is to find the best combination of feature selection techniques for web page categorization problem. It also overcomes the issues in feature selection. This process contains 3 stages: a) the extraction of representative features, to describe content -

the initial set, b) the selection of the best features from initial set by applying another feature selection technique (minimizing the number of features and maximizing the discriminative information carried by them) and c) the training and classification using the resulting features in the different classifiers to determine the quality of features.

Algorithm Two level Feature Selection (set of keywords)

```
{
1. Initial Feature Selection
Do stop word, Common Word, Punctuation symbols and HTML tags removal
Do stemming
Find frequency of occurrence of independent words
Select words which occur n times
These words form the initial feature set.
2. Final feature selection
Apply cfs subset evaluator with either rank or random search methods.
3. Classification phase
Test the quality of the features with the final feature set to find the best method
}
```

1 Feature Selection phase

Feature selection is normally done by searching the space of attribute subsets, evaluating each one. This is achieved by combining attribute subset evaluator with a search method. In this study, we choose seven attribute evaluators with five search methods to find the best feature set.

For the feature selection phase, two objects must be set up: a feature evaluator and a search method. The evaluator determines what method is used to assign a worth to each subset of features. The search method determines what style of search is performed.

The feature selection can be done in two ways: 1) using full training set (the worth of the feature subset is determined using the full set of training data), or 2) by cross-validation (the worth of the feature subset is determined by a process of cross-validation). In addition, the classifying time grow dramatically with the number of features, rendering the algorithm impractical for problems with a large number of features. In practice, the choice of a learning scheme (the next phase) is usually far less important than coming up with a suitable set of features. We experimented with several evaluators and search methods:

I. Evaluators:

- CfsSubsetEval - Evaluates the worth of a subset of features by considering the individual predictive ability of each feature along with the degree of redundancy between them; subsets of features that are highly correlated with the class while having low inter-correlation are preferred.
- ConsistencySubsetEval - Evaluates the worth of a subset of features by the level of consistency in the class values when the training instances are projected onto the subset of features.
- ClassifierSubsetEval- classifier attributes subset evaluator.
- SymmetricalUncertAttributeSetEval- Evaluates the worth of a feature by measuring the symmetrical uncertainty with respect to the class.
- WrapperSubseEval- Wrapper attributes subset evaluator
- OneRAttributeEval

- PCA - Performs a principal components analysis and transformation of the data.

II. Search methods:

- BestFirst - Searches the space of feature subsets by greedy hill-climbing augmented with a backtracking facility.
- GeneticSearch - Performs a search using the simple genetic algorithm
- Ranker - Ranks features by their individual evaluations. Use in conjunction with feature evaluators (ReliefF, GainRatio, Entropy etc).
- Greedy Stepwise - performs greedy search
- FCFB Search - performs search based on who comes first.

2 Classification phase

In our approach we use initial set of features for the course category of the WebKB data set, a benchmarking data set for the selection phase. The initial set of features is shown in Table 1. For the classification phase using the initial set of features we experimented with two classifiers: Support Vector Machine (SVM) and Naïve Bayes classifier. We select these two classifiers as they are proved to be the best classifiers for the text/web mining applications. The results using the initial set of features are shown in Table 2.

I. Naïve Bayes classifier

The Naïve Bayes classifier is the simplest instance of a probabilistic classifier. The output **p(c|d)** of a probabilistic classifier is the probability that a pattern d belongs to a class c after observing the data **d** (posterior probability). It assumes that text data comes from a set of parametric models (each single model is associated to a class). Training data are used to estimate the unknown model parameters. During the operative phase, the classifier computes (for each model) the probability **p(d|c)** expressing the probability that the document is generated using the model. The Bayes theorem allows the inversion of the generative model and the computation of the posterior probabilities (probability that the model generated the pattern). The final classification is performed selecting the model yielding the maximum posterior probability. In spite of its simplicity, a Naïve Bayes classifier is almost as accurate as state-of-the-art learning algorithms for text categorization tasks. The Naïve Bayes classifier is the most used classifier in many different web applications, such as focus crawling, recommending systems, etc.

II. SVM based Classifiers

Support Vector Machine (SVM) has been demonstrated its excellent performance in terms of solving document classification problem. SVM is the first choice for web page classification because of its advantage on non-effective of feature dimension scale. In its simplest form, a linear SVM is a hyperplane that separates a set of positive examples from a set of negative examples with maximum margin. The margin is the distance from the hyperplane to the nearest of the positive and negative examples. For problems that are non linearly separable, kernel methods can be used to transform the input space so that some non-linear problems can be learned. SVM is used mostly with other classifiers as shown below for better performance (Mitchell, 1999).

Experimental setup

For our experiment the database used is WebKB data set and is downloaded from the UCI repository. It is a benchmarking dataset for machine learning problems. This is the

university database having seven categories of web pages: Course, Project, Student, Faculty, Department, Staff and others. We select as positive examples all the pages in the course category (930 pages) and as negative examples some pages from the non course category (66 pages) and as positive examples all the pages from the student category (1341 pages) and as negative examples some pages from the non student category (229 pages). The experimental set up is shown in Table 1. The initial set of features for "course" and "student" categories are listed in Tables 2 and 3 respectively.

Table 1: Experimental Setup

S. No.	Name of the category	No. of positive examples	No. of negative examples
1.	Course	930	66
2.	Student	1341	229

The measures used here are CCI and F-measure. CCI refers correctly classified instances.

F-Measure is used in Information Retrieval to characterize the performance of the classifier. It is defined as

$$\text{F measure} = \frac{2 \times recall \times precision}{recall + precision} = \frac{2 \times TP}{2 \times TP + FP + FN}$$

Where TP –the number of True Positives
FP – the number of false positives
FN – the number of false negatives

$$recall = \frac{\text{Number of documents retrieved that are relevant}}{\text{Total number of documents those are relevant}}$$

$$precision = \frac{\text{Number of documents retrieved that are relevant}}{\text{Total number of documents that are retrieved}}$$

Results and analysis

1 Bi-level Feature Selection Process

1.1 Initial Feature Selection Phase

Initial set of features are selected from the training set as follows. All the HTML marks ups are removed and the web page is converted into a text file. All the stop words, common words and punctuation symbols are removed. Then all the words in the training set are sorted and the number of occurrence of each word is found out. Then the words which occur more than 150 times in the training set are taken as features. This is because these words may very well represent the category of the web page as they occur frequently in the training set. The initial set of features obtained using above method is listed in Table 2 for Course category and Table 3 for Student category.

Table 2: Initial set of features for the Course category

Course, class, syllabus, handout, homework, cs, lecture, notes, slides, solution, problem, program, instructor, information, project, paper, guide, study, prelim, professional, activities, resume, publications, language, research, teaching, contact,

projects, professor, interests, department, personal, office, advisor, home, page, links, phone

Table 3: Initial set of features for the Student category

Computer, university, department, page, research, information, student, systems, engineering, computing, work, institute, technology, graduate, conference, email, interest, tech, learning, technical, resume, group, advisor, report

Using the initial set of features for the course category listed in table 2, we perform web page classification using Support Vector Machine classifier and Naïve Bayes classifier and the results we obtain are shown in Table 4.

Table 4: Classification accuracy using Initial set of features (course)

Sl. No.	SVM Classifier		Naïve Bayes Classifier	
	CCI total 996	F-measure	CCI total 996	F-measure
1	980	.988	973	.985

The classification results using the initial features for student category listed in Table 3 with the Support Vector Machine classifier and Naïve Bayes Classifier are listed in Table 5.

Table 5: Classification accuracy using Initial set of features (student)

Sl. No.	SVM Classifier		Naïve Bayes Classifier	
	CCI total 1570	F-measure	CCI total 1570	F-measure
1	1343	.921	1353	.923

The F-measure values in tables 4 and table 5 shows that this initial set of features provide good classification accuracy. The number of initial features in the initial feature set remains high and so to reduce the number of features without any reduction in the classification performance, we go for next level of experiments. By reducing the number of features, the advantages are less storage, and minimum processing time required and is the requirement of today's world.

1.2 Final Feature Selection phase

We apply many feature selection techniques on the initial set of features to find which method best suits to get the final set of features. The feature selection evaluators, search methods and the list of features selected by this combination are shown in Table 6 and Table 7 for the course and student category respectively.

Table 6: Feature selection phase from the initial set of features (course)

Method Name	Search Name	No. of features selected	Selected Features
Principal Components	Ranker	32	course, class, syllabus, handout, homework, cs, lecture, notes, slides, solution, problem, program, instructor, office, information, project, paper, guide, study, prelim, professional, activities, resume, publications,

			language, research, teaching, contact, projects, professor, interests, department
Consistency Subset Eval	Best First	24	class, syllabus, homework, lecture, notes, problem, program, instructor, office, information, project, guide, study, professional, resume, publications, language, contact, projects, professor, department, home, page, links
	Greedy Stepwise	24	class, syllabus, homework, lecture, notes, problem, program, instructor, office, information, project, guide, study, professional, resume, publications, language, contact, projects, professor, department, home, page, links
	Genetic Search	30	class, syllabus, handout, homework, cs, lecture, slides, solution, problem, program, instructor, office, information, project, paper, guide, study, prelim, professional, activities, research, contact, projects, professor, department, advisor, home, page, links, phone
Cfs Subset Eval	Genetic Search	18	handout, homework, lecture, notes, instructor, guide, study, prelim, activities, publications, language, research, teaching, contact, projects, associate, interests, personal
	Greedy Stepwise	4	lecture, slides, prelim, home
	Rank Search	4	lecture, study, prelim, home
	Best First	4	lecture, slides, prelim, home
Wrapper Subset Eval	Genetic Search	3	program, professional, publications
Symmetrical Uncert Attribute Set Eval	FCBF Search	4	lecture, study, prelim, home
Classifier Subset Eval	Best First	12	class, syllabus, homework, instructor, information, guide, study, resume, associate, advisor, home, links
OneRAttributeEval	Ranker	35	class, syllabus, handout, homework, cs, lecture, notes, slides, solution, problem, program, office, information, project, paper, guide, study, prelim, professional, activities, publications, language, research, teaching, contact, projects, professor, interests, department, personal, advisor, home, page, links, phone

Table 7: Feature selection phase from the initial set of features (student)

Method Name	Search Name	No. of features selected	Selected Features
Principal Components	Ranker	21	computer, university, department, page, research, information, student, systems, engineering,

			computing, work, institute, technology, graduate, conference, email, interest, tech, learning, technical, resume
Consistency Subset Eval	Genetic search	20	computer, university, department, page, research, information, student, systems, engineering, computing, work, institute, technology, graduate, conference, email, interest, tech, learning, technical
Cfs Subset Eval	Genetic Search	7	page, student, engineering, interest, resume, group, advisor
	Greedy Stepwise	6	page, student, interest, resume, group, advisor
	Rank Search	6	page, student, interest, resume, group, advisor
	Best First	6	page, student, interest, resume, group, advisor
	Random Search	6	page, student, interest, resume, group, advisor
Wrapper Subset Eval	Rank Search	1	advisor
	Genetic search	1	University
Classifier Subset Eval	Genetic search	1	Advisor
	Rank Search	1	university

The number of features in the final feature set ranges from 2 to 35 for the course category and 1 to 21 for the student category. To determine whether this reduction in the initial set is worth having, we use SVM and Naïve Bayes classifiers for web page classification with this final set of features.

1.3 Classification phase

We choose SVM and Naïve Bayes classifier for web page classification with this final feature set so as to compare it with the results obtained using initial feature set. The results are tabulated for the course category in Table 8 and for the student category in table 9. Experimental results in table 8 shows that even with 4 features instead of 38 features, we can classify the web pages with acceptable accuracy. Only very minimal degradation in the performance, but the memory and time requirement is very less. For the project category, it is interpreted from the results that even a single good quality feature is enough to attain acceptable classification accuracy.

Table 8: Experimental results using final set of features (course)

Method Name	Name of the search	No. of attributes selected	Naïve Bayes		SVM	
			Correctly classified Instances	Macro F-measure	Correctly classified Instances	Macro F-measure
Principal Components	Ranker	32	974	0.986	982	0.99
Consistency Subset Eval	Best First	24	960	0.978	973	0.986
	Greedy	24	960	0.978	973	0.986

	Stepwise					
	Genetic Search	30	967	0.982	974	0.986
Cfs Subset Eval	Genetic Search	18	964	0.981	961	0.979
	Greedy Stepwise	4	934	0.966	934	0.966
	Rank Search	4	934	0.966	934	0.966
	Best First	4	934	0.966	934	0.966
Wrapper Subset Eval	Genetic Search	3	938	0.968	941	0.969
Symmetrical Uncert Attribute Set Eval	FCBF Search	4	934	0.966	934	0.966
Classifier Subset Eval	Best First	12	953	0.975	959	0.978
OneRAttributeEval	Ranker	35	967	0.984	975	0.987

Table 9: Experimental results using final set of features (student)

Method Name	Name of the search	No. of attributes selected	Naïve Bayes		SVM	
			Correctly classified Instances	Macro F-measure	Correctly classified Instances	Macro F-measure
Principal Components	Ranker	21	1346	0.921	1343	0.921
Classifier Subset Eval	Rank search	1	1343	0.921	1343	0.921
	Genetic Search	1	1343	0.921	1343	0.921
Cfs Subset Eval	Genetic Search	7	1347	0.922	1343	0.921
	Greedy Stepwise	6	1346	0.922	1343	0.921
	Rank Search	6	1347	0.922	1343	0.921
	Best First	6	1347	0.922	1343	0.921
	Random Search	6	1347	0.922	1343	0.921
Consistency Subset Eval	Genetic search	20	1343	0.921	1343	0.921
WrappersubsetEval	Rank search	1	1343	0.921	1343	0.921
	Genetic search	1	1343	0.921	1343	0.921

Here the results reveal that Cfs Subset Evaluator is the best method to get final feature set. Greedy stepwise search method takes more time and so either rank search or random search is suggested to have a good feature set. We show that term frequency method is the

easiest method to get the initial feature set. To refine this initial feature set, we can use Cfs subset evaluator method with either rank search or best first search. This will give good quality features that reduce the time and resource required for processing.

Conclusion and future work

Applying only one level of feature selection techniques results in more number of features and so time and resource required for web page classification problem remains more. To avoid this, in this paper we suggest an approach that refine the initial set of features. Our results show that applying two levels of feature selection techniques, produces good quality features that reduce the resource requirement. As a future work, this work may be combined with feature extraction algorithm to get more relevant features for classification.

References

- Borkar, V.R., Deshmukh, K., & Sarawagi, Sunita (2001). Automatic segmentation of text into structured records. *SIGMOD Conference*, pp.175-186.
- Chen, K., & Liu, H. (1999) Towards an evolutionary algorithm: Comparison of two feature selection algorithms. *Proceedings in Congress on Evolutionary Computation*.
- Combarro, E.F., et al. (2005). Introducing a family of linear measures for feature selection in text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 17(9), 1223-1232.
- Daume III, H., & Brill, E. (2004). Web search intent induction via search results partitioning. *Proceedings of HLT 2004*.
- Kim, S., & Zhang, B.T. (2003). Genetic mining of html structures for effective web-document retrieval. *Applied Intelligence*, 18(3), 243-256.
- Krishnapuram, B., Harternink, A.J., Carin, L., Figueiredo, M.A.T. (2004). A bayesian approach to joint feature selection and classifier design. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26 (9), 1105-1111.
- Lipsman, A. (2007). 61 billion searches conducted worldwide in August. *ComScore: Measuring the Digital World*, October 10, 2007.
- Liu, H., & Setino, R. (1996). Feature selection and classification- A probabilistic wrapper approach. *Proceedings of IEA-AIE* 1996.
- Liu, H., & Yu, L. (2005). Towards integrating feature selection algorithm for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17(4), 491-502.
- Lu, H., Sung, S.Y., & Lu, Y. (1996). On preprocessing data for effective classification. *Proceedings of ACM SIGMOD '96 Workshop on Research Issues on Data Mining and Knowledge Discovery*, Montreal, Canada, June 1996.
- Mitchell, T.M. (1999). The role of unlabeled data in supervised learning. *Proceedings of the Sixth International Colloquium on Cognitive Science*, San Sebastian, Spain.
- Salzberg, S. (1992). Improving classification methods via feature selection. *Proceedings of ACM*.
- Terdiman, D. (2005). Yahoo passes Google in search index capacity. *News.com*, August 8, 2005.
- Vafaie, H., & De Jong, K. (1993). Robust feature selection algorithms. *Proceedings on Fifth International Conference on Tools with Artificial Intelligence*, Boston, MA: IEEE Computer Society Press, pp. 356--363.
- Yan, Y., & Pederson, J.O. (1997). Comparative study of feature selection in text categorization. *Proceedings on Fourteenth International Conference on Machine Learning (ICML'97)*, pp. 412-420.
- Zheng, Z., Srihari, R., & Srihari, S. (2003). A feature selection framework for text filtering. *Third IEEE International Conference on Data Mining* (ICDM), November

19-22, 2003, pp. 705- 708.

Bibliographic information of this paper for citing:

Indra Devi, M., Rajaram, R., & Selvakuberan, K. (2008). "Generating best features for web page classification." *Webology*, **5**(1), Article 52. Available at: http://www.webology.org/2008/v5n1/a52.html

---
