# Resource Optimization in Automatic web page classification using integrated feature selection and machine learning

Indra Mahadevan
Department of IT,
Thiagarajar College of Engg.,
Madurai

Selvakuberan Karuppasamy
Innovation Labs(Web 2.0)
TATA Consultancy Services
Chennai

Rajaram Ramasamy
Department of CSE
Thiagarajar College of Engg
Madurai

**Abstract** *Increasing with the number of users, the need for automatic classification techniques with good classification accuracy increases as search engines depend on previously classified web pages stored in classified directories to retrieve the relevant results. Preprocessing is the important step in web page classification problem as most of the web pages contain more irrelevant information than relevant details useful for finding its category. For the classification purpose the representative words in the web page called as features are used for classification rather than the entire web page to reduce time and space requirements. The selection of relevant features which reduces the high dimensionality and redundancy is the current research topic. In this paper selecting relevant features from the pages is treated as an optimization problem and we propose an algorithm to find the optimal features for web page classification. Machine learning techniques for automatic classification gains more interest as the classifier improves its performance with experience. In this paper we use Naïve Bayes, Kstar, Random Forest and Bagging machine learning classifiers.*

**Keywords:** *Web page classification, resource optimization, feature selection, machine learning*

## 1. Introduction

Information databases today contain millions of electronic documents. There is an exponential increase of information available on the World Wide Web. In 2005, Yahoo claimed that its index covered more than 20 billion web resources, the largest search engine [11]. It is believed that the actual size of the Web is at least several times bigger than what search engines currently cover. URL of the web page is the unique resource used to retrieve the web page. It is impossible for the users to remember the URL of all the web pages. They rely on search engines most of the times to fetch the required information. Search engines also have a large repository of web pages and they need a better classification technique to retrieve the required information. Even though manual classification of web pages will be an excellent one, it is impractical for search engines to adopt it as the sheer size and dynamic nature of the web pages prevent this. A typical short query of one to three keywords submitted to a search engine easily retrieves tens of thousands of web pages. As the amount of online texts such as web pages dramatically increases, the demand for text categorization to aid efficient retrieval, by filtering out unsound web pages and management of the World Wide Web is increasing. Describing and organizing this vast amount of content is essential for realizing the web's full potential as an information resource. At popular portal sites such as Yahoo Service, web pages should be classified into many categories since they have directory style search engines.[2] At present, however the classification of web pages into many categories relies on time consuming and expensive human effort. Such popular portal sites thus need to save time and costs by classifying the web pages automatically. Most of the search engines use machine learning algorithms for automatic classification of web pages as these machine learning classifiers learn themselves from the examples and improve with experience. The web pages have thousands of words and there is no uniform size for the web pages. Instead of using all the words in a web page to find its category, representative features from the web pages may be used for classification. This approach reduces the dimension, removes redundancy and improves classification accuracy. This paper mainly focus on how OR concepts (Resource Optimization) have an impact to find the relevant features of the web page.

The paper is organized as follows. Section 2 focuses on related work. Section 3 describes about the Resource Optimization. Section 4 explains about the feature selection and the issues in feature selection. Section 5 describes our algorithm; section 6 shows the experimental set up and the discussion of results. Section 7 ends with conclusion. Section 8 describes the references.

## 2. Related work

Rudy Setiono and Huan Liu proposed the use of Principal Component Analysis to get a new set of representative features[6]. Zhaohui Zheng, Rohini Srihari Sargur Srihari proposed a new method that combines positive and negative examples [1]

Hongjun Lu, Sam Yuan Sung and Ying Lu proposed conflict analysis that is finding a set of attributes having perfect association with the class labels. Contingency table analysis is used with the nominal variables –the variables whose values are from an unordered set used chi square statistics.[3]

Ali Selamat, Hidekazu Yanagimoto and Sigeru Omatu propose a news web page classification method (WPCM). The WPCM uses a neural network with inputs obtained by both the principal components and class profile-based features (CPBF). The fixed number of regular words from each class will be used as vectors with the reduced features from the PCA. These feature vectors are then used as the input to the neural networks for classification [7]

Ismail Sengör Altingövde and Özgür Ulusoy propose a crawler which employs canonical topic taxonomy to train a naïve-Bayesian classifier, which then helps determine the relevancy of crawled pages. The crawler also relies on the assumption of topical locality to decide which URLs to visit next. Building on this crawler, they developed a rule-based crawler, which uses simple rules derived from interclass (topic) linkage patterns to decide its next move. This rule-based crawler also enhances the baseline crawler by supporting tunneling.[5]

Susan Dumais and Hao Chen explores the use of hierarchical structure for classifying web Pages using Support Vector Machine Classifiers. The hierarchical structure is initially used to train different second-level classifiers. In the hierarchical case, a model is learned to distinguish a second-level category from other categories within the same top level.[8]

We differ from the above approaches by applying OR techniques to the feature selection problem. In the above papers only one algorithm is used to find the representative features from the web pages. In our approach, we propose two levels of feature selection. In the second level we apply OR techniques and found that cfssubset evaluator method gives the best feature set for web page classification.

## 3. Resource Optimization

Resource Optimization is the most recent trend in the classification of pages. The operations research community has recently made significant contributions in this area and in particular to the design and analysis of data mining algorithms. For example, mathematical programming formulations of support vector machines have been used for feature selection and data clustering. The intersection of OR and data mining is not limited to algorithm design and data mining can play an important role in many OR applications. The data mining process is usually effective for generating insights and patterns from large data sets, it is a model free approach and the insights are typically unstructured and require substantial interpretation. Thus, optimization methods can potentially be applied to the output of the data mining process to optimize the desired objective while accounting for relevant business constraints.

In this paper we describe this optimal feature selection problem for web page classification as an optimization problem and we propose a new two level feature selection algorithm to find the optimal features for the given problem.We follow OR concept to find the features for the given problem, In OR a basic decision model is to be established first and a search procedure is applied to determine the problem solution. We apply several evaluators and search methods to find the best features for the web page classification problem and propose a novel algorithm to find the optimal features for the web page classification problem. The web page classification is considered as an Operational Research problem by defining

Maximize *classification accuracy*
subject to minimum *number of features*
with respect to *any dimensions of web pages*..

## 4. Feature Selection

Feature selection is one of the most important steps in pattern recognition or pattern classification, data mining, machine learning and so on. It is difficult to measure classification information in all features.[9][10] Data preprocessing is an indispensable step in effective data analysis. It prepares data for data mining and machine learning, which aim to turn data into business intelligence or knowledge. Feature selection is a data

preprocessing technique commonly used on high dimensional data. Feature selection studies how to select a subset or list of attributes or variables that are used to construct models describing data. Its purposes include reducing dimensionality, removing irrelevant and redundant features, reducing the amount of data needed for learning, improving algorithm's predictive accuracy, and increasing the constructed model's comprehensibility. Feature-selection methods are particularly welcome in interdisciplinary collaborations because the selected features retain the original meanings domain experts are familiar with. The rapid developments in computer science and engineering allow for data collection at an unprecedented speed and present new challenges to feature selection. Wide data sets, which have a huge number of features but relatively few instances, introduce a novel challenge to feature selection.[2]

Large number of features brings disadvantages for classification problem. On one hand, increased features give difficulties to calculate, because the more data occupy large amount of memory space and require more computerization time, on the other hand, a lot of features include certainly many correlation factors respectively, which results to information repeat and waste. Therefore, we must take measures to decrease the feature dimension under not decreasing recognition effect; this is called the problems of feature optimum extraction or selection. The number of features needs to be constrained to reduce noise and to limit the burden on system resources.

## 5. Proposed Approach

In this paper we propose an algorithm which selects the optimal features to be used for classification of web pages using OR techniques. 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 RO in Classification (set of keywords)
{
 First Feature Selection (keywords)
{
  //select the features using term frequency approach
  return initial set of features;
}
Second Feature Selection(initial set of features)
{
 // Minimize the number of features by selecting the most relevant ones by using evaluators and search methods
   return final set of features;
}
Classification (final set of features)
 // classify using machine learning classifiers
     Maximize classification accuracy with minimal number of features
}
```

### Our proposed approach is follows:

#### i. Feature Selection Phase I

In this phase the first step is preprocessing the web page. First we remove all the stop words, punctuation symbols in the web page. Then all the distinct words with their frequencies are found out. Then we remove all the words whose frequencies are less than the threshold. Here all the rare and infrequent words are removed. The remaining words in the web page are the keywords of the particular category. This phase selects the features which we call as initial set of features. The selected features are based on the number of occurrences of the particular feature in a web page and this approach is called as term frequency approach. We select this approach to find initial set of features for the following reasons: easy to implement, guaranteed to get best features, no relevant feature gets omitted, time, processing power and memory requirement are affordable. Even this set of features can be used for classification, a refinement on the feature set shows worthier results.

#### ii. Feature Selection phase II

This phase selects the most relevant attributes from the initial set of features using the combination of evaluators and search methods as described below. It will result in "final set of features". These "final set of features" are proved to be most relevant for web page classification
Feature selection is 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 paper we choose four 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 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:

**5.1.1 Evaluators:** [4]

- CfsSubsetEvalutor - 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.
- ConsistencySubsetEvaluator - 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.
- PCA - Performs a principal components analysis and transformation of the data.
- Wrapper Subset Eval –Generates the subset of attributes using wrappers.

**5.1.2 Search methods:** [4]

- Best First - Searches the space of feature subsets by greedy hill-climbing augmented with a backtracking facility.
- Genetic Search - 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).
- Exhaustive search- It searches a particular keyword throughout the web page. Its performance is little bit slower.
- Forward Selection- It arbitrarily selects the most relevant attributes starting from the first one. It follows a hill climbing approach.

**iii. Classification phase**

Use the final set of features obtained from step 2, classify using machine learning classifiers. Machine learning classifiers used here are naïve bayes, K Star, Random Forest and Bagging

**Naïve Bayes**

The Naïve Bayes classifier is the simplest instance of a probabilistic classifier. The output ***p(a|b)*** of a probabilistic classifier is the probability that a pattern ***b*** belongs to a class ***a*** after observing the data ***b*** (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(b\a)*** 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. This section shows how this Naïve Bayes Algorithm is used for Web Page Classification by various researchers. It obeys the principle of Bayes theorem which is based on the probabilistic function

$$p(A/B) = \frac{P(B/A)P(A)}{P(B)}$$

where A and B are two stochastic events**.**

**KStar**

KStar is an instance-based classifier that is the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function.
The use of entropy as a distance measure has several benefits. Amongst other things it provides a consistent approach to handling of symbolic attributes, real valued attributes and missing values. K* is an instance-based learner which uses such a measure.

### Random Forests

A random forest is a classifier consisting of a collection of tree structured classifiers {h(**x**,Θk ), k=1, ...} where the {Θk} are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x .Random forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. The generalization error for forests converges as to a limit as the number of trees in the forest becomes large. The generalization error of a forest of tree classifiers depends on the strength of the individual trees in the forest and the correlation between them. Given an ensemble of classifiers $h_1(x)$, $h_2(x)$, ... ,$h_K(x)$, and with the training set drawn at random from the distribution of the random vector Y,X, define the

margin function as

$$mg(X,Y) = av_k\ I(h_k(X) = Y) - \max_{(j!=Y).} av_k\ I(h_k(X) = j$$

Where

$I\tilde{}()$ is the indicator function.

The margin measures the extent to which the average number of votes at X, Y for the right class exceeds the average vote for any other class. The larger the margin, the more is the confidence in the classification. The generalization error is given by

$$PE^* = P_{X,Y}(mg(X,Y) < 0)$$

where the subscripts X,Y indicate the probability over the space.

The advantages of random forests are

- For many data sets, it produces a highly accurate classifier.
- It handles a very large number of input variables.
- It estimates the importance of variables in determining classification.
- It generates an internal unbiased estimate of the generalization error as the forest building progresses.
- It includes a good method for estimating missing data and maintains accuracy when a large proportion of the data are missing.
- It provides an experimental way to detect variable interactions.
- It can balance error in class population unbalanced data sets.
- It can also learn very fast

### Bagging

Bagging is a ``bootstrap'' ensemble method that creates individuals for its ensemble by training each classifier on a random redistribution of the training set. Each classifier's training set is generated by randomly drawing, with replacement, *N* examples - where *N* is the size of the original training set; many of the original examples may be repeated in the resulting training set while others may be left out. Each individual classifier in the ensemble is generated with a different random sampling of the training set.

## 6. 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 all the pages in the course category (930 pages) as positive examples and non course category (66 pages) as negative examples to classify course category pages. We select all the pages from the project category (400 pages) as positive examples and non project category (80 pages) as negative examples to classify project category web pages. The experimental set up is listed in Table 1.The initial set of features for "course" category is listed in Table 2 and the initial set of features for "project" category are listed in Table 3. We have done some of our experiments with WEKA software to select final set of features.

**Table 1. Experimental Setup**

| S.No | Name of the category | No. of positive examples | No. of negative examples |
|---|---|---|---|
| 1. | Course | 930 | 66 |
| 2. | Project | 400 | 80 |

To determine the quality of the features we use Macro F measure. 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

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

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

## 7. Results and Analysis

The results of feature selection phase-I are tabulated in Table 2 for course category and Table 2 for project category. The preprocessing stage contains stop words, common word, punctuation symbols, rare words and infrequent words removal and stemming. After this preprocessing step, by applying term frequency approach, the course category contains 38 features and project category contains 22 features. Even though these features are enough for web page classification to achieve maximum classification accuracy with minimum number of features, feature selection phase-II is applied on these feature sets.

**Table 2. Initial set of features for the Course category**

| |
|---|
| Course(1), class(2), syllabus(3), handout(4), homework(5), cs(6), lecture(7), notes(8), slides(9), solution(10), problem(11), program(12), instructor(13), information(14), project(15), paper(16), guide(17), study(18), prelim(19), professional(20), activities(21), resume(22), publications(23), language(24), research(25), teaching(26), contact(27), professor(28), interests(29), department(30), personal(31), office(32), advisor(33), home(34), page(35), associate(36), phone(37), links(38) |

**Table 3. Initial set of features for the project category**

| |
|---|
| university(1), research(2), computer(3), edu(4), project(5), department(6), laboratory(7), group(8), applications(9), information(10), work(11), computing(12), performance(13), lab(14), software(15), problems(16), faculty(17), tools(18), techniques(19), database(20), learning(21), communication(22) |

### 7.1 Final Feature Selection phase

In this phase as described in section 5 we apply four attribute evaluators with five search methods on the initial feature set. The results are tabulated in Table 4 for the course category and in Table 5 for the project category. Here the number of features selected range from 2 to 36 for the course category and 1 to 21 for the project category. Some features in the initial set like links in the course category and communication in the project category are not selected by any of the feature selection techniques. This shows that some of the least significant features get selected in the initial category and this time and memory resources utilized by these types of features get wasted. Now to judge the quality of these final features, we go for the classification phase.

**Table 4. Final Feature Set (course category)**

| Method Name | Search Name | No. of features selected | Selected Features |
|---|---|---|---|
| Principal Components | Ranker | 36 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35 |
| Consistency Subset Evaluator | Best First | 32 | 1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 32, 14, 15, 16, 18, 22, 23, 24, 25, 26, 27, 36, 28, 29, 30, 31, 33, 34, 35, 37 |
| | Exhaustive search | 36 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35 |
| | Forward selection | 30 | 1, 2, 3, 5, 6, 7, 8, 11, 12, 13, 32, 14, 15, 16, 18, 22, 23, 24, 25, 27, 36, 28, 29, 30, 31, 33, 34, 35, 37 |
| Cfs Subset Evaluator | Genetic Search | 12 | 1, 2, 3, 22, 23, 24, 25, 36, 29, 30, 31, 33 |
| | Forward Selection | 11 | 1, 2, 3, 5, 13, 22, 23, 25, 29, 30, 33 |
| | Rank Search | 10 | 1, 2, 3, 13, 22, 23, 25, 29, 30, 33 |
| | Best First | 11 | 1, 2, 3, 5, 13, 22, 23, 25, 29, 30, 33 |
| | Exhaustive Search | 35 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 32, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 36, 28, 29, 30, 31, 33 |
| Wrapper Subset Evaluator | Rank Search | 2 | 29, 33 |

**Table 4 . Final Feature Set (project category)**

| Method Name | Search Name | No. Of features selected | Selected Features |
|---|---|---|---|
| Principal Components | Ranker | 19 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19 |
| Consistency Subset Evaluator | Best First | 15 | 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17 |
| | Genetic search | 16 | 1, 2, 3, 4, 5, 8, 15, 9, 10, 11, 12, 13, 14, 18, 19, 21 |
| | Rank search | 21 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21 |
| Cfs Subset Evaluator | Genetic Search | 4 | 2, 5, 7, 9 |
| | Best First | 4 | 2, 5, 7, 9 |
| | Rank Search | 4 | 2, 5, 7, 9 |
| | Greedy Stepwise | 4 | 2, 5, 7, 9 |
| | Random search | 5 | 2, 5, 7, 9, 18 |
| Wrapper Subset Evaluator | Rank Search | 1 | 2 |
| | Genetic search | 1 | 9 |
| Classifier Subset Evaluator | Genetic search | 1 | 9 |
| | Rank search | 1 | 2 |

As shown in Table 3 and table 4 number of final features vary from 1 to n where n < m where m is the number of initial features.The classification accuracy obtained using the initial set of features for the course and project categories are shown in tables 5 and 6 respectively.

**Table 5. Classification accuracy using initial set of features (course category)**

| Sl.No | Naïve Bayes | | Kstar | | Random Forest | | Bagging | |
|---|---|---|---|---|---|---|---|---|
| | CCI total 996 | F - measure | CCI total 996 | F - measure | CCI total 996 | F - measure | CCI total 996 | F - measure |
| 1 | 977 | 0.988 | 997 | 0.998 | 965 | 0.981 | 932 | 0.965 |

**Table 6. Classification accuracy using initial set of features (project category)**

| Sl.No | Naïve Bayes | | Kstar | | Random Forest | | Bagging | |
|---|---|---|---|---|---|---|---|---|
| | **CCI** total 480 | **F - measure** | **CCI** total 480 | **F - measure** | **CCI** total 480 | **F - measure** | **CCI** total 480 | **F - measure** |
| 1 | 423 | 0.922 | 425 | 0.927 | 431 | 0.935 | 429 | 0.932 |

The classification accuracies obtained using final set of features are shown in Table 7 for the course category and in table 8 for the project category.

**Table 7. Classification accuracy using final set of features (course category)**

| S. No | Classifier name | Search name | Naïve bayes | | Kstar | | Random Forest | | Bagging | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | CCI | Macro F | CCI | Macro F | CCI | Macro F | CCI | Macro F |
| 1. | cfssubsetevaluator | Bestfirst | 980 | 0.989 | 976 | 0.987 | 977 | 0.988 | 932 | 0.965 |
| 2 | Cfssubsetevaluator | Forward selection | 980 | 0.989 | 976 | 0.987 | 977 | 0.988 | 932 | 0.965 |
| 3 | Cfssubsetevaluator | Genetic Search | 976 | 0.987 | 967 | 0.983 | 973 | 0.986 | 932 | 0.965 |
| 4 | Cfssubsetevaluator | Exhaustive search | 970 | 0.984 | 967 | 0.982 | 966 | 0.982 | 932 | 0.965 |
| 5 | cfssubsetevaluator | Rank search | 980 | 0.989 | 972 | 0.985 | 977 | 0.988 | 932 | 0.965 |
| 6 | Wrapper subsetevaluator | Rank search | 951 | 0.974 | 937 | 0.967 | 951 | 0.974 | 932 | 0.965 |
| 7 | Consistency subsetevaluator | Exhaustive search | 975 | 0.986 | 974 | 0.986 | 964 | 0.981 | 932 | 0.965 |
| 8 | Consistency subsetevaluator | Bestfirst | 978 | 0.988 | 972 | 0.985 | 973 | 0.986 | 932 | 0.965 |
| 9 | Consistency subsetevaluator | Forward selection | 976 | 0.987 | 970 | 0.984 | 971 | 0.984 | 932 | 0.965 |
| 10 | Principal components | Ranker | 975 | 0.986 | 974 | 0.986 | 964 | 0.981 | 932 | 0.965 |

The results tabulated in Table 7 shows that the Macro F measure value ranges from .974 to .989 for the final feature set whereas it is .988 for the initial set. The minimum value (.974) is for wrapper subset evaluator method which is obtained using only one feature where for the initial set the number of features used for classification is 38. Time and memory requirement is very less and so this minimal deviation in the results is acceptable. Cfs subset evaluator produces approximately equal results with less number of features (10 or 11). For the Kstar and Random forest classifiers also the same results are obtained. For bagging classifier there is no difference in the results. This shows that applying wrapper subset evaluator with rank search on the initial set of features obtained using term frequency approach, reduces the time and memory requirement and for applications like web page classification which requires large memory this proposed approach perform better.

**Table 8. Classification accuracy using final set of features (project category)**

| S. No | Classifier name | Search name | Kstar | | Naïve Bayes | | Random Forest | | Bagging | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | CCI | Macro F | CCI | Macro F | CCI | Macro F | CCI | Macro F |
| 1. | Cfssubsetevaluator | Bestfirst | 434 | 0.938 | 433 | 0.937 | 434 | 0.938 | 434 | 0.938 |
| 2 | Cfssubsetevaluator | Genetic search | 434 | 0.938 | 433 | 0.937 | 434 | 0.938 | 434 | 0.938 |
| 3 | Cfssubsetevaluator | Greedy stepwise | 434 | 0.938 | 433 | 0.937 | 434 | 0.938 | 434 | 0.938 |
| 4 | Cfssubsetevaluator | Random search | 434 | 0.938 | 433 | 0.937 | 434 | 0.938 | 434 | 0.938 |
| 5 | Cfssubsetevaluator | Rank search | 434 | 0.938 | 433 | 0.937 | 434 | 0.938 | 434 | 0.938 |
| 6 | Consistency subsetevaluator | Best First | 436 | 0.94 | 429 | 0.93 | 431 | 0.935 | 426 | 0.929 |
| 7 | Consistency subsetevaluator | Genetic search | 432 | 0.936 | 422 | 0.921 | 424 | 0.926 | 427 | 0.93 |
| 8 | Consistency subsetevaluator | Rank search | 423 | 0.923 | 421 | 0.919 | 436 | 0.941 | 427 | 0.93 |
| 9 | Principal components | Ranker | 428 | 0.93 | 422 | 0.921 | 437 | 0.942 | 427 | 0.93 |
| 10 | Classifier subset evaluator | Genetic search | 402 | 0.907 | 402 | 0.907 | 402 | 0.907 | 402 | 0.907 |
| 11 | Classifier subset evaluator | Rank search | 402 | 0.907 | 392 | 0.888 | 395 | 0.893 | 402 | 0.907 |
| 12 | Wrapper subset evaluator | Genetic search | 402 | 0.907 | 402 | 0.907 | 402 | 0.907 | 402 | 0.907 |
| 13 | Wrapper subset evaluator | Rank search | 402 | 0.907 | 392 | 0.888 | 395 | 0.893 | 402 | 0.907 |

The results tabulated in Table 8 also prove that applying wrapper subset evaluator on the initial set produces very minimal number of feature ( here 1) with small degradation in the classification accuracy. Applying cfssubset evaluators on the initial set produces more accurate results with minimal number of features. By applying OR technique for the web page classification problem , we found that cfssubset evaluator applied on the initial set formed using term frequency approach produces minimal number of features which provides maximum classification accuracy.

## 8. Conclusion and Future Work

Experimental results show that our proposed approach classifies the web pages more accurately. We use machine learning classifiers for web page classification and so the accuracy of classification increases with more number of testing and real time data. These machine learning classifiers learned from the training and testing examples and with more number of pages used for classification, our algorithm certainly will produce a reliable and valuable result. As a future work, feedback from interested users may be used to train the machine learning classifiers.

## References


[1] Zhaohui Zheng Rohini Srihari Sargur Srihari, "A Feature Selection Framework for Text Filtering" *Third International Conference on Data Mining ICDM'03*

[2] Tom M.Mitchell, "The Role of Unlabeled data in Supervised Learning" *Proceedings of the Sixth International Colloquium on Cognitive Science,1999*

[3] Hongjun Lu, Sam Yuan Sung and Ying Lu, "On Preprocessing Data for Effective Classification",*Workshop on Research Issues on Data Mining and Knowledge Engineering, 1996*

[4] Witten. I.H and Frank.E, "*Data mining: Practical machine learning tools and techniques with Java implementations",* Morgan Kauffmann, San Francisco, CA,2000

[5] Ismail Sengör Altingövde and Özgür Ulusoy, "Exploiting Interclass Rules for Focused Crawling", IEEE Intelligent Systems, 2004, pp 66 – 73

[6] Rudy Setiono and Huan Liu, "Feature Selection via Discretization" *IEEE Transactions on Knowledge and Data Engineering, Vol 9,Issue 4, 1997 ,pp 642-645*

[7] Ali Selamat, Hidekazu Yanagimoto and Sigeru Omatu, "Web News Classification Using Neural Networks Based on PCA*", SIC€ 2002*

[8] Susan Dumais, Hao Chen, "Hierarchical Classification of Web Content", SIGIR 2000 , ACM, pp 256 –263

[9] Balaji Krishnapuram, Alexander J. Hartemink, Lawrence Carin and Mario A.T. Figueiredo, "A Bayesian Approach to Joint Feature Selection and Classifier Design", IEEE *Transactions on Pattern*

*Analysis and Machine Intelligence, Vol. 26, No. 9,2004, pp1105 – 1111*
[10] Yiming Yan and Jan O. Pederson, "Comparative Study of feature selection in Text Categorization", *Proceedings on Fourteenth International Conference on Machine Learning (ICML'97) pp 412-420*
[11] Terdiman, D. Yahoo passes Google in search index capacity. *News.com, August 8, 2005.*

M.Indra Devi received the B.E. degree in Computer Science and Engineering from Madurai Kamaraj University in 1990 and M.E. degree in Computer Science and Engineering from Madurai Kamaraj University in 2003. She is currently a lecturer at Information Technology Department at Thiagarajar College of Engineering, Madurai. She has published fourteen papers in national and international conferences. Her research interests include machine learning applications and web mining. She is a life member of Computer Society of India, Institution of Engineers, India and Indian Society for Technical Education.

K.Selvakuberan received the B.Tech degree in Information Technology from Thiagarajar College of Engineering, Madurai in 2007. He is currently an associate in Tata Consultancy Services. He is working in Innovation Labs(Web 2.0) in Chennai and his research interests include data mining, web mining and machine learning. He has published fourteen papers in national and international conferences.

Dr R.Rajaram, Dean, Computer Science and Information Technology, Thiagarajar College of Engineering, Madurai has a B.E. Degree (1966) in Electrical and Electronics Engineering from University of Madras. He secured the M.Tech Degree (1971) in Electrical Engineering from IIT Kharagpur and the Ph.D degree (1979) from Madurai Kamaraj University. He has been teaching computer hardware and software and supervising research activities. He and his research students have published nearly 45 papers in journals, seminars and symposia. Six of his research students have secured the Ph.D degree from Madurai Kamaraj University, two are waiting for the results and seven are currently pursuing their works. His areas of interest are Data mining, Machine learning, Neural Networks, Network Security, Fuzzy Systems and Genetic algorithms. He has published more than 13 text books for schools and colleges in English and one book in Computer Science in Tamil. He attended the International Symposium on Solar Energy at the University of Waterloo, Canada, during August 1978. He served at the Makerere University, Kampala, Uganda during 1977 – 79, and at the Mosul University, Iraq during 1980 – 1981. He secured two best technical paper awards from the Institution of Engineers (India), one from the Indian Society of Technical Education. He studied at Malaysia and has traveled to London, Paris, New York, Toronto, Nairobi and Colombo.