Madhumita Dange : mvd130130@utdallas.edu

Project title- **To learn classifiers to predict the type of a webpage from the text**

Abstract: Basic idea of the project is to apply different classifying approaches on same data set and analyses the results of different classifiers. I have 6 classes to classify. I have take WebKb Data Set and classified the web pages. This dataset contains webpages from 4 universities, labeled with whether they are professor, student, project, or other pages. I have trained on three of the universities plus the *misc* collection, and testing on the pages from a fourth(utexas), held-out university and apply different approaches and record the results.

About Dataset: This data set contains WWW-pages collected from computer science departments of various universities in January 1997 by the World Wide Knowledge Base (Web->Kb) project of the CMU text learning group. The 5,440 pages were manually classified into the following categories:

- student (1641)
- faculty (1124)
- staff (137)
- department (182)
- course (930)
- project (504)

All the data training data is converted into arff format in order to construct models using Weka.

Train Test Spilt:

I have separated one university's (uttexas)data as Test data(256 files). And other 3 universities and misc data is used for training.

For this project, I used 6 classifiers with appropriate tuning and one combined Vote classifier:

1. J48- Decision tree learning uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value. More descriptive names for such tree models are classification trees or regression trees. In these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels.

2.SMO- support vector machines are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and

regression analysis. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the output, making it a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. I have used linear Kernel.

3.KNN 1 - the k-nearest neighbors algorithm (k-NN) is a non-parametric method for classification and regression, that predicts objects' "values" or class memberships based on the k closest training examples in the feature space. k-NN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. The k-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors (k is a positive integer, typically small). k = 1, then the object is simply assigned to the class of that single nearest neighbor.

4.KNN 3 : with k=3 Distance measured used : Euclidian

5.Naive Bayes- A naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model".

6. Adaboost with Decision Stump - It is a meta-algorithm, and can be used in conjunction with Decision Stump which is model consisting of a one-level decision tree.

Model settings- threshold : 100000 , iterations -30

7.My Vote Classifier - I have constructed weighted vote classifier which uses Decision tree, Naive Bayes and Support vector machine models.

Data Preprocessing :

For data preprocessing I have used StringToWord vector, Snowball stemmer and removed stop words using weka's meta.FilteredClassifier.

Minimum word occurrence frequency used: 2

Snowball- a framework for writing stemming algorithms, and implemented an improved English stemmer together with stemmers for several other languages.

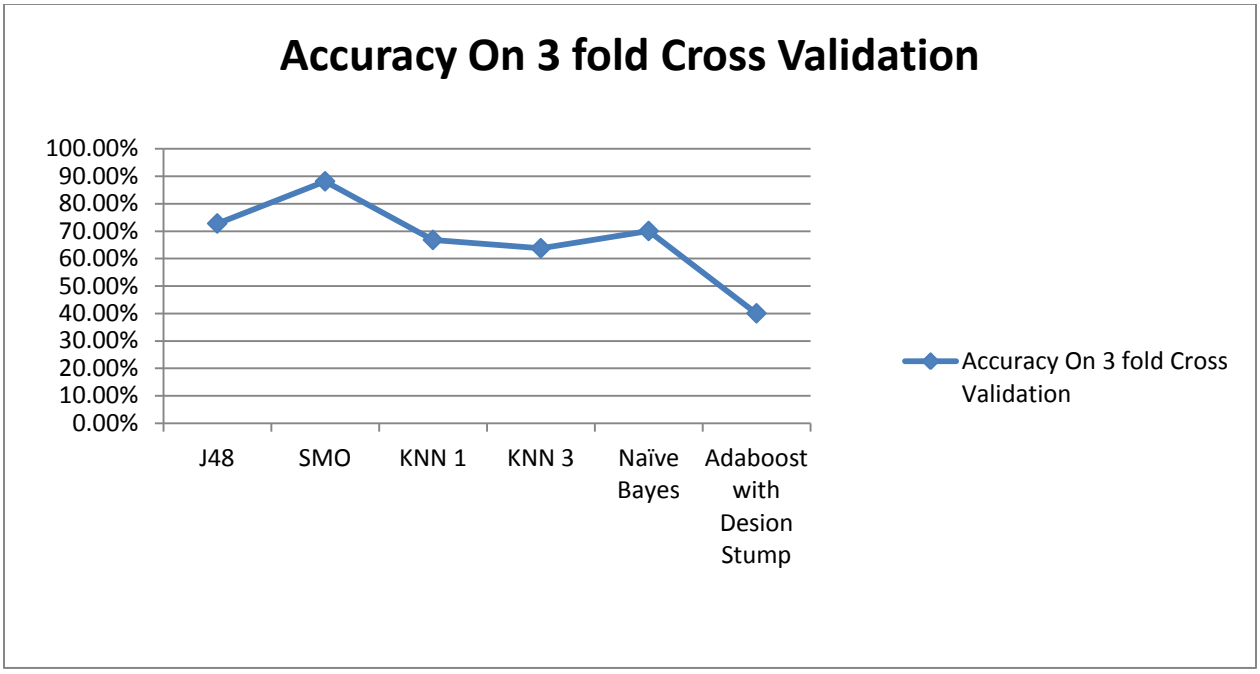
I used attribute selection and Ranker with information gain as criteria to rank attributes.

Training :

I have trained a model using **3 fold cross validation** for each of 6 classifiers and saved those models.

Training accuracies:

Classifier	Accuracy On 3 fold Cross Validation
J48 (Decision Tree)	72.76%
SMO	88.12%
KNN 1	66.78%
KNN 3	63.75%
Naïve Bayes	70.02%
AdaboostwithDecision Stump	40.03%



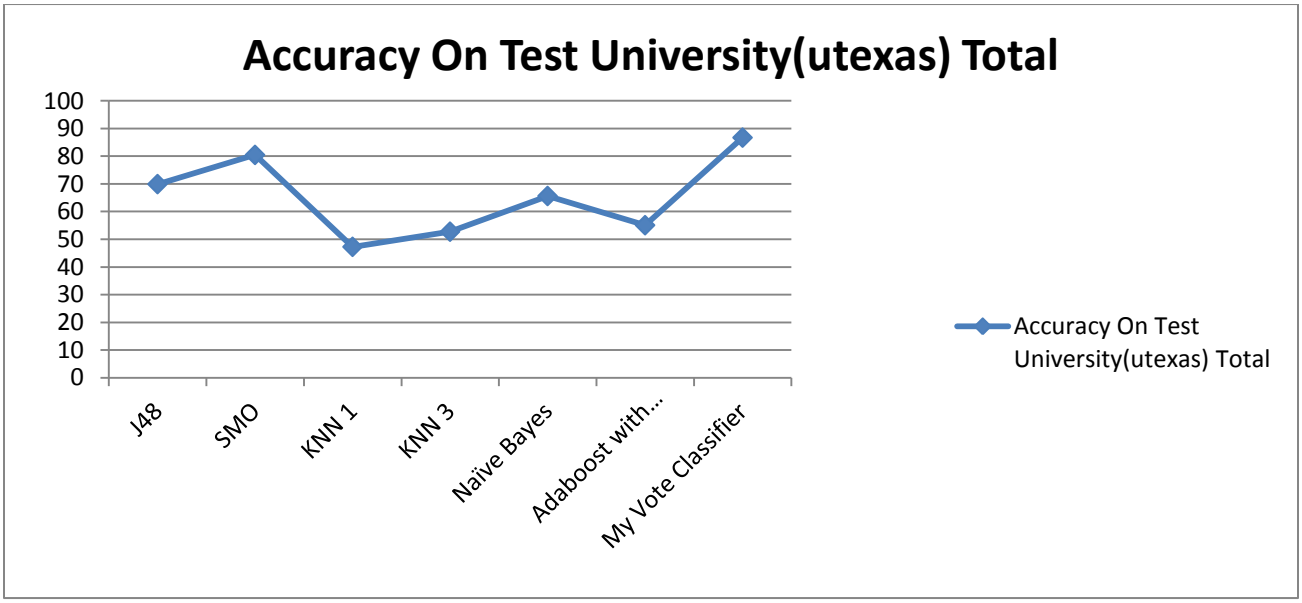
Testing:

I have used these 6 models to test Test data

As cross-validation gives maximum accuracy for SMO(support vector machine with linear Kernel), J48 and Naïve Bayes So I have constructed one weighted vote classifier using these three as My Vote classifier.

Results For the Test university data :

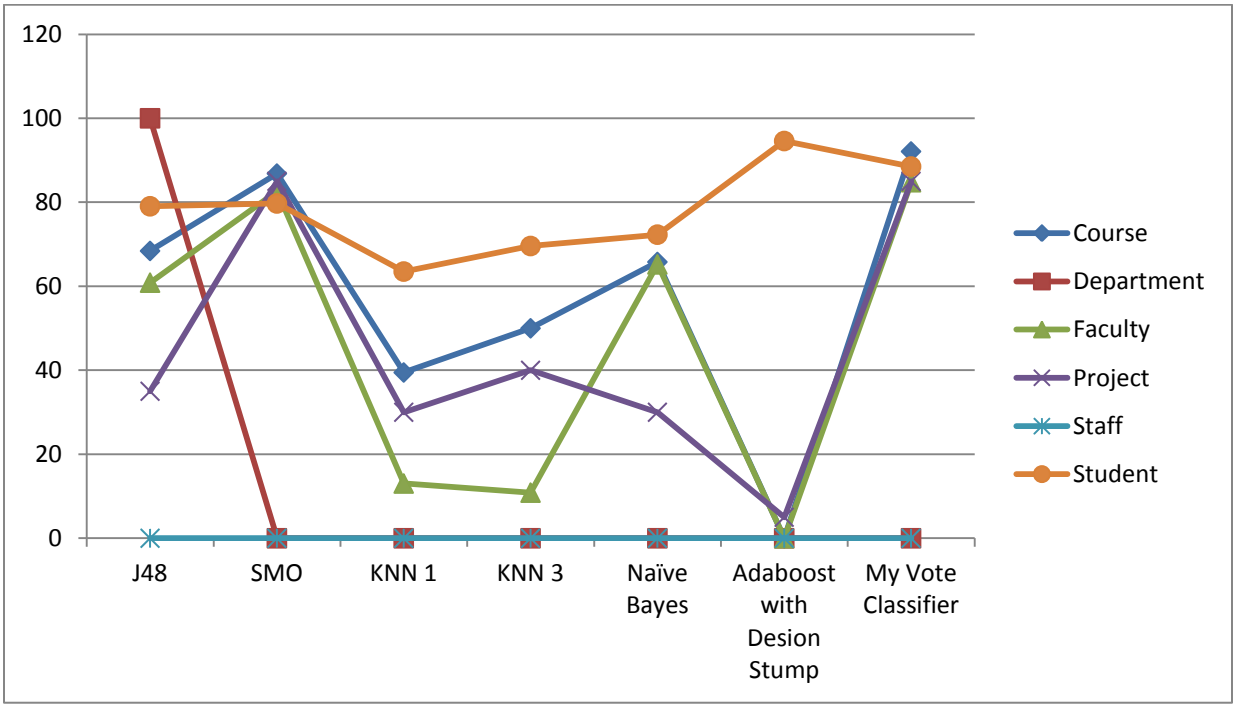
Classifier	Accuracy On Test University(utexas) Total
J48	69.921875
SMO	80.46875
KNN 1	47.265625
KNN 3	52.7344
Naïve Bayes	65.625
Adaboost with Decision Stump	55.078125
My Vote Classifier	86.71875



Class wise Accuracies:

Classifier	Course	Department	Faculty	Project	Staff	Student
J48	68.42105263	100	60.86956522	35	0	79.05405
SMO	86.84210526	0	82.60869565	85	0	79.72973
KNN 1	39.47368421	0	13.04347826	30	0	63.51351
KNN 3	50	0	10.86956522	40	0	69.59459
Naïve Bayes	65.78947368	0	65.2173913	30	0	72.2973
Adaboost with Desion Stump	0	0	0	5	0	94.59459
My Vote Classifier	**92.105263**	**0**	**84.7826**	**85**	**0**	**88.51**
Total Test files	38	1	46	20	3	148

Classwise accuracies for Test data



Observations and Analysis:

Combined Vote classifier works better on Test set. Adaboost is failed on this dataset. Accuracies for department and staff webpages are poor because less amount of training data compare to other classes. Student class has maximum accuracy as training data is large for that class.As large number of attributes are used data is linearly separable .

Future Work : If given more computational resources, I could experiment with more webpages and also experiment with large number of classifiers.

Conclusion :

More data and Voted classification is the best strategy for my Project to get higher accuracies.

Bibliography:

1. http://www.cs.waikato.ac.nz/ml/weka
2. http://en.wikipedia.org
3. http://www.youtube.com/watch?v=IY29uC4uem8&list=PLB8Egcmig-2cr7jFlEqSQJPtlSVczvSX2
4. http://www.cs.cmu.edu/~webkb/
5. http://www.esp.uem.es/jmgomez/tmweka/index.html
6. http://jmgomezhidalgo.blogspot.com/2013/02/text-mining-in-weka-revisited-selecting.html
7. http://www.hlt.utdallas.edu/~vgogate/ml/lectures.html