**PROJECT REPORT**

**DATA MINING**

**Guided By: Team Members:**

Frank Calliss Mohit Motiani (1206306331)

Qingyang Li Basavaprasad Chandu (1207081560)

**INTRODUCTION:**

The project meant to extract data from snopes.com and to come up with a classifier which predict the class label of testing data and provides good accuracy to evaluate the quality of predicted labels by comparing it with hidden labels. For this purpose we have to use different classifiers to predict the class label of testing data.

The project in data mining is mainly divided in three tasks.

The first task is to extract data from one of the category (Politics) from snopes.com. The output of the extraction is needed to be stored in a specified format.

The second part which includes the conversion of data to frequency set is done by Qingyang Li.

The third part includes building a classifier which can predict the output label of testing data with maximum accuracy.

We have used different classifiers to predict the output label and the below literature will show our research work done for building the classier and finding the maximum accuracy.

**LITERATURE REVIEW:**

Classification is a data mining technique that assigns items to a particular category or a class. The main task of classification is to predict class label for each and every data in the set[1]. Say for example, we have been given training data set and testing data set. Also the class label of training data set is given. We have to predict the label of testing data. Analysis of data also includes prediction besides classification for the data which we have used in this project[2]. Now prediction of output label can only be done through a tool. So for that purpose we have used weka.

Weka is a tool which comprises of certain number of machine learning algorithms to implement data mining tasks[3]. Weka itself contains tools for classification, preprocessing, clustering, visualization, association rules.

For the purpose of predicting class labels, we have used classification tool in the weka. This tool accepts training data and build up a training model with that data and by using this model on our testing data we can predict the output of testing data. For building up a model we can use any classifier. Weka provides a large number of classifier for building a model.

Classifier such as rainforest, naïve bayes, support vector machine are present inside the weka. Initially we have used naïve bayes classifier for building that training model. A naïve bayes classifier is probabilistic model which is used here to predict the output label of testing data with strong independence assumption. A naïve bayes classifier assumes that the value of a particular feature is related or unrelated to any other feature. In the Naive Bayes classifier, all variables are assumed to be nominal variables, it means that each variable has a finite number of variables[4].

After first submission of class label of testing data through naïve bayes classifier, we have used OneR classifier. OneR classifier provides a framework for the construction of decision tree for the classifying large data sets.

**APPROACH USED:**

**Task 1:** In this task we have to extract data from the POLITICS category of the snopes.com website. In this task, we have to extract data and store them in a specified format. The scratched data includes subtopics within the main topic and these sub topic includes sub-sub topics. The files in which these data are to be stored are saved in a text file. Say for example, we need to extract the data from <http://www.snopes.com/politics/arts/livemusic.asp> and then we have to store the data in a text file named politics\_arts\_livemusic.txt. For this data extraction, we have used a jsoup, a HTML parser that takes java extensions and allowed us to extract the data in needed format.

**Task 3:** In this task, we have to build a classifying model which provides maximum accuracy on the prediction of label of testing data. There are total 1842 instances in training set, 952 instances in testing set and 26364 different attributes.

In our **first submission**, we have used Naïve Bayes classifier for building training model from the training data set. Weka accepts data in a particular format which is .arff format. The approach which we have used is that we have created a matrix of data set which do not includes zero. The matrix will be the representation of training data. To create this matrix we have written a code in java which accepts the training.txt file and read the complete file line by line and store that in the form string.

Each row in the matrix represents an instance and there are 1842 rows in the matrix and each row has 26364 attributes. The instance whose feature value is not present is taken zero by default and this default value is not shown in the matrix. After creating the matrix from training.txt file, we have attached label at the end of each row of the matrix from label\_training.txt file. After creating the complete training data set, the file is converted to .csv file which is further converted to .arff file which is a weka format.

Same approach is used for creating testing data set. But in this, we have left the column for label because that will be predicted by the weka. We have used cross validation of 10 folds in Naïve Bayes classifier technique to build our model

In our **second submission**, we have used OneR classifier in which features are randomly selected and classified. As, the features are randomly selected, it is very easy to know how features are related to each other. For this classifier also, we have used cross validation of 10 folds in OneRs classifier technique to build the model.

**ENCOUNTERED PROBLEMS:**

* We faced a lot of problems during the implementation of project. Initially when we have faced the problem with the training data set. We have created the matrix from data set which includes zero, that is, for the feature whose values are not given is taken as zero. By doing this, the size of the matrix reaches around 180Mb. Building a model in weka with such a huge data set is quite difficult because of its heap size.
* After solving the problem of heap size, when model has built, the problem which occurred was the problem of predicting output label from testing data. The matrix which we created from testing set was working perfectly but it was not generating the output, i.e. label for testing data is coming as ZERO for all instances.

**SOLUTION:**

* The first problem of heap size was solved by adjusting the heap size, i.e. by increasing the heap size in the installation file of weka. After changing the heap size, the problem of model building with large file of data was over.
* For solving second problem, we have used a different technique. We have removed all the features whose values are zero, i.e. we have not used the default zero value for any of the feature. The main benefit which we had from this technique is that, size of the training set and the testing set has reduced significantly, which eventually solved the first problem, and we have easily got classification model on the training set which is further used to predict the output label of testing set.

**REFERENCES:**

**[1]** [**http://docs.oracle.com/cd/B28359\_01/datamine.111/b28129/classify.htm#DMCON004**](http://docs.oracle.com/cd/B28359_01/datamine.111/b28129/classify.htm#DMCON004)

**[2]** [**http://www.tutorialspoint.com/data\_mining/dm\_classification\_prediction.htm**](http://www.tutorialspoint.com/data_mining/dm_classification_prediction.htm)

**[3]** [**http://www.cs.waikato.ac.nz/ml/weka/**](http://www.cs.waikato.ac.nz/ml/weka/)

**[4] Yongchuan Tang, Wuming Pan, Haiming Li and Yang Xu. Fuzzy Naïve Bayes classifier based on fuzzy clustering.**