Assignment – B11

Roll no -4234

Naive Bayes

import java.io.BufferedReader;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.IOException;

import java.util.Random;

import weka.classifiers.Evaluation;

import weka.classifiers.bayes.NaiveBayes;

import weka.core.Instances;

public class Classifier {

public static BufferedReader readDataFile(String filename) {

BufferedReader inputReader = null;

try {

inputReader = new BufferedReader(new FileReader(filename));

} catch (FileNotFoundException ex) {

System.err.println("File not found: " + filename);

}

return inputReader;

}

public static void main(String[] args) {

BufferedReader read;

try {

//read = readDataFile("src/train.arff");

//System.out.println(read.readLine());

//Instances train = new Instances(read);

//train.setClassIndex(train.numAttributes()-1);

//read.close();

read = new BufferedReader (new FileReader("src/test.arff"));

Instances train = new Instances(read);

train.setClassIndex(train.numAttributes()-1);

NaiveBayes nB = new NaiveBayes();

nB.buildClassifier(train);

Evaluation eval = new Evaluation(train);

eval.crossValidateModel(nB, train, 10, new Random(1));

System.out.println("Naive Bayes Classification\n");

System.out.println(eval.toSummaryString("\nResults===============>\n",true));

//System.out.println(eval.fMeasure(1)+" "+eval.precision(1)+" "+eval.recall(1));

System.out.println(eval.toMatrixString("\nConfusion Matrix: false positives and false negatives\n======\n"));

System.out.println(eval.toClassDetailsString("Class Details..\n"));

//System.out.println(eval.truePositiveRate(1));

} catch (FileNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

/\*

Naive Bayes Classification

Results===============>

Correctly Classified Instances 457 62.2616 %

Incorrectly Classified Instances 277 37.7384 %

Kappa statistic -0.0003

K&B Relative Info Score -52265.1312 %

K&B Information Score -464.2993 bits -0.6326 bits/instance

Class complexity | order 0 626.0881 bits 0.853 bits/instance

Class complexity | scheme 1173.3596 bits 1.5986 bits/instance

Complexity improvement (Sf) -547.2714 bits -0.7456 bits/instance

Mean absolute error 0.141

Root mean squared error 0.2884

Relative absolute error 157.8886 %

Root relative squared error 137.7829 %

Total Number of Instances 734

Confusion Matrix: false positives and false negatives

======

a b c d e f <-- classified as

0 2 0 0 1 0 | a = audio

0 438 0 30 152 7 | b = pdf

0 2 0 0 1 0 | c = excerpt

0 31 0 1 5 0 | d = pamphlet

0 32 0 0 18 0 | e = unknown

0 14 0 0 0 0 | f = video

Class Details..

TP Rate FP Rate Precision Recall F-Measure ROC Area Class

0 0 0 0 0 0.621 audio

0.699 0.757 0.844 0.699 0.764 0.547 pdf

0 0 0 0 0 0.419 excerpt

0.027 0.043 0.032 0.027 0.029 0.51 pamphlet

0.36 0.232 0.102 0.36 0.159 0.717 unknown

0 0.01 0 0 0 0.642 video

Weighted Avg. 0.623 0.665 0.729 0.623 0.665 0.558

\*/

DECISION TREE

import java.awt.BorderLayout;

import java.io.BufferedReader;

import java.io.FileReader;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import weka.classifiers.Evaluation;

import weka.classifiers.trees.J48;

import weka.core.Instances;

import weka.gui.treevisualizer.PlaceNode2;

import weka.gui.treevisualizer.TreeVisualizer;

public class Classifier {

private static final Logger logger = LoggerFactory.getLogger(Classifier.class);

public static void main (String[] args) throws Exception

{

BufferedReader breader = null;

breader = new BufferedReader (new FileReader("src/test.arff"));

Instances train = new Instances(breader);

train.setClassIndex(train.numAttributes()-1);

breader = new BufferedReader (new FileReader("src/test.arff"));

Instances test = new Instances(breader);

test.setClassIndex(test.numAttributes()-1);

breader.close();

J48 tree= new J48();

tree.setOptions(null);

tree.buildClassifier(train);

// Instances labeled = new Instances (test);

Evaluation eval = new Evaluation(train);

eval.evaluateModel(tree, test);

System.out.println("Decision Tree\n");

logger.info(eval.toSummaryString("\nSummary\n======\n", false));

logger.info(eval.toClassDetailsString("\nClass Details\n======\n"));

logger.info(eval.toMatrixString("\nConfusion Matrix: false positives and false negatives\n======\n"));

// System.out.println("Fmeasure"+eval.fMeasure(1));

// visualize(tree);

}

public static void visualize(J48 tree) throws Exception{

// display classifier

final javax.swing.JFrame jf = new javax.swing.JFrame("Tree Visualizer: J48");

jf.setSize(500,400);

jf.getContentPane().setLayout(new BorderLayout());

TreeVisualizer tv = new TreeVisualizer(null,(tree).graph(),new PlaceNode2());

jf.getContentPane().add(tv, BorderLayout.CENTER);

jf.addWindowListener(new java.awt.event.WindowAdapter() {

public void windowClosing(java.awt.event.WindowEvent e) {

jf.dispose();

}

});

jf.setVisible(true);

tv.fitToScreen();

}

}

/\*

Decision Tree

Oct 05, 2015 11:26:29 PM Classifier main

INFO:

Summary

======

Correctly Classified Instances 627 85.4223 %

Incorrectly Classified Instances 107 14.5777 %

Kappa statistic 0

Mean absolute error 0.0876

Root mean squared error 0.2093

Relative absolute error 98.2695 %

Root relative squared error 99.9929 %

Total Number of Instances 734

Oct 05, 2015 11:26:29 PM Classifier main

INFO:

Class Details

======

TP Rate FP Rate Precision Recall F-Measure ROC Area Class

0 0 0 0 0 0.5 audio

1 1 0.854 1 0.921 0.5 pdf

0 0 0 0 0 0.5 excerpt

0 0 0 0 0 0.5 pamphlet

0 0 0 0 0 0.5 unknown

0 0 0 0 0 0.5 video

Weighted Avg. 0.854 0.854 0.73 0.854 0.787 0.5

Oct 05, 2015 11:26:29 PM Classifier main

INFO:

Confusion Matrix: false positives and false negatives

======

a b c d e f <-- classified as

0 3 0 0 0 0 | a = audio

0 627 0 0 0 0 | b = pdf

0 3 0 0 0 0 | c = excerpt

0 37 0 0 0 0 | d = pamphlet

0 50 0 0 0 0 | e = unknown

0 14 0 0 0 0 | f = video

\*/