# **HW 05 – DECISION TREES**

**DATA PREPARATION**

1. Removed all the disputed essays from the data set

Text

Description automatically generated

1. Essays (without disputed authorship) are divided as **training and testing dataset with 60% and 40%** proportion respectively



Application

Description automatically generated with medium confidence

**So, 44 essays** are used for **training** the decision tree classifier and **30 essays** for **testing** the predictions.

1. Both training and testing essays have the essays of all the authors – Hamilton, Madison, Jay, and Hamilton & Madison

Text

Description automatically generated with medium confidence

**BUILDING MODEL AND POST PRUNING**

1. Created decision tree classifier with ‘entropy’ criterion for determining the best split at every level

Text

Description automatically generated

Decision tree created by the model from the training data looks like:

A picture containing text, sign

Description automatically generated

**Textual Representation of the Decision Tree**

|--- upon <= 0.02

| |--- and <= 0.51

| | |--- class: Madison

| |--- and > 0.51

| | |--- would <= 0.04

| | | |--- class: HM

| | |--- would > 0.04

| | | |--- class: Jay

|--- upon > 0.02

| |--- class: Hamilton

Feature **upon** has the highest information gain and is considered to be the first feature to classify the authors.

1. Validating the model to check if there is any overfitting problem using confusion matrix
   1. Training data is having 100% accuracy

Table

Description automatically generated

A picture containing logo

Description automatically generated

* 1. Model is not performing well on the testing data. Essays authored by Jay and Madison are being classified as HM. It can happen that the model is overfitted which is common problem with decision trees

Table

Description automatically generated

Square

Description automatically generated with low confidence

**Performing Post Pruning to avoid overfitting**

Pruning technique is parameterized by the cost complexity parameter, **ccp\_alpha**. Greater values of **ccp\_alpha** increase the number of nodes pruned. It is necessary to choose right **ccp\_alpha** to cut down the branches of the decision tree.

Based on different ccp\_alpha values found from the training data, accuracy was plotted for training and testing data sets

Graphical user interface, text, application

Description automatically generated

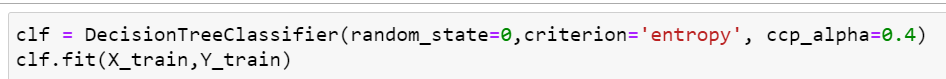
A picture containing text

Description automatically generated

From the plot, cost complexity value between 0.2 to 0.8 seems to be stable with accuracy on training and testing data sets.

We can also choose alpha value above 0.0 but less than ~0.05 because the accuracy remained constant for training and testing data around that value.

**Training the model with minimum cost complexity parameter = 0.4**



Decision Tree resulted from the above classifier looks like:

Graphical user interface, text, application, chat or text message

Description automatically generated

|--- upon <= 0.02

| |--- class: Madison

|--- upon > 0.02

| |--- class: Hamilton

This model is performing well in classifying Hamilton and Madison essays but not w.r.t Jay and Hamilton & Madison essays.

Confusion matrix of the training data

Table

Description automatically generated

Icon

Description automatically generated with medium confidence

Confusion matrix of the testing data

Table

Description automatically generated

Background pattern, icon

Description automatically generated with medium confidence

**PREDICTING AUTHOR OF DISPUTED ESSAYS**

According to the prediction by the decision tree classifier, all the **disputed essays are written by Madison**.

Table

Description automatically generated