Intro to IS Lab 2:

Wikipedia Language Classification

CSCI 630 – Foundations of Artificial Intelligence

Submitted by,

SHEETAL SANTOSH KASHID

Introduction

The lab assignment aims to classify Wikipedia language into English or dutch exploring two different implementations.

One is decision tree and another is adaboost algorithm. We were asked to collect the data and train the respective algorithms and predict based on the decision stumps obtained from the algorithm.

Data Collection

Dataset was obtained from an online resource. It is of the following form :

<Language encoding(en or nl)> | <sentence from wiki>

En – English

Nl – Dutch

Feature Extraction

The features were extracted from the data which helped train the model.

The following features were found to be most accurate and hence giving 100% accuracy with decision tree classifier.

1. If the sentence contained any of the following words 'she', 'the', 'a', 'an', 'and' , 'you' , 'but' then the language is english. Therefore this is one of the features having ‘YES’ if the language contains any of these words and ‘NO’ if it doesn’t.
2. If the sentence contained any of the following words 'het', 'de', 'een' , 'en' , 'de' , 'eng' , 'maar' then the language is Dutch. Therefore this is one of the features having ‘YES’ if the language contains any of these words and ‘NO’ if it doesn’t.
3. Dutch words are comparatively longer. Therefore greater than average length is one of the features which gives ‘YES’ if the length is greater than 5 and ‘NO’ if it is not.
4. Dutch words contain a greater number of ks and js than English. Hence this is also one factor which can be considered. That is if the count of k and j in a sentence is greater than 3.

5. Another more common feature in Dutch words is the occurrence of double letters. Therefore, if occurrence of double letters is more than 2 then it is probably a Dutch word.
6. The Dutch sentences also contain the word q which is unlikely in English. Hence presence of q is also a feature.

Decision Tree Learning

Decision tree uses the concept of information gain which helps in choosing the best attribute to use to split at that level.

Entropy is calculated as

$$\text{Entropy:} \quad H(V) = \sum_k P(v_k) \log_2 \frac{1}{P(v_k)} = -\sum_k P(v_k) \log_2 P(v_k) \ .$$

Image credits: textbook

We want to determine which attribute in a given set of training feature vectors is most useful for discriminating between the classes to be learned.

• Information gain tells us how important a given attribute of the feature vectors is.

• We will use it to decide the ordering of attributes in the nodes of a decision tree.

$$Remainder(A) = \sum_{k=1}^{d} \frac{p_k+n_k}{p+n} B(\frac{p_k}{p_k+n_k}) \ .$$

$$Gain(A) = B(\frac{p}{p+n}) - Remainder(A) \ .$$

Image credits: textbook

There are 4 cases to be considered while implementing the dtree algorithm.

1. If all the examples have the same target value left ie all English or all dutch then return that value

2. If we are left with no attributes or features then return the parent classification
3. If we go more than the maxdepth then return the parent dtree
4. If no rows left again return the tree
5. Otherwise you can add more branches to the tree

Growing branches if choosing the best split using Information gain and then recursively calling the sub data formed after splitting the data based on the best attribute.

Hence we call the decision tree algorithm recursively.

Depth parameter is added to make it more concise.

Testing results:

Testing with 80 random samples gave an accuracy of 100% with this algorithm.

Some of the cases with their output is given below:

Cases:

1. als station, terwijl de stationschef in de dienstwoning uit 1839 bleef wonen. Pas in 1931
2. be imposed. Decision tree learning is the construction of a decision tree from class-labeled training
3. decision tree, so that every internal node has exactly 1 leaf node and exactly 1
4. werd het dienstgebouw opgetrokken, dat zich eveneens onder een schilddak bevindt, langs de straatzijde verspringend
5. internal node as a child (except for the bottommost node, whose only child is a
6. of shooting are correctly formalized. (Predicate completion is more complicated when there is more than
7. beperken, zorgde de NMBS in 2004 voor 60 extra parkeerplaatsen aan het station van Duffel.
8. root node. There are many specific decision-tree algorithms. Notable ones include: While the Yale shooting
9. van de Vlaamse overheid, om de overlast tijdens de werken aan de Antwerpse Ring te

10. described received the AAAI Classic Paper award. In spite of being a solved problem, that

Respective ouputs:

1. nl
2. en
3. en
4. nl
5. en
6. en
7. nl
8. en
9. nl
10. en

Adaboost Implementation

To implement adaboost, decision stumps need to be considered. Each decision stumps act as weak learners and we iteratively find k weak learners which correct each other by updating the weights and giving more weight to the incorrectly classified ones and less to the correctly classified. The next stump will give more weight to the incorrect ones and hence more samples of which are generated and fed to a new weak learner in the next iteration. The various stumps or hypothesis with their weights form our model for the adaboost classification.

Adaboost uses the information gain to find the best feature and uses it as one decision stump then adaptively forms various weak learners which combined gives us a strong classifier.

1. Initially the weights are 1/N being the number of samples
2. Classify samples using stumps
3. Calculate the error ie the number of misclassified samples

4. Calculate the alpha for the stump

   α **= ½ ln (1 – Total error/Total error)**

   This alpha is used as performsnce measure and is used to update the weight of the wrongly classifies ones in the next step.

5. Update the weights

   New weight = Weight * $e^{(performance)}$ if incorrect
   New weight = Weight * $e^{-(performance)}$ if correct

6. Normalize the weights
7. Update the weights in the prediction
8. The stumps and final weights of each give the model.

Conclusion

Both the classifier give good accuracy and requires somewhat same time to train the classifier.