Foundations of Artificial Intelligence
Lab2
Venkata Karteek Paladugu

Features:

Based on manual inspection and doing research on several examples of Dutch and English from WIkipedia Random page I have come up with the following features.

1.Repetition of Vowel words:

Many of the dutch words had repetition of letters based on my research on the data seen in wiki random pages. These are the repetition of vowel words that i found are common:
aa, ee, ii, oo, uu. These can be used to separate Dutch sentences.

2.Common dutch words:

Based on my research and manual inspection found these common Dutch words:
op, wat, het, en, ik, je, niet, dat, de. These can be used to separate Dutch sentences.

3.Common english words:

Based on my research and manual inspection found these common English words:
the, be, and, of, to, a, in, have, it. These can be used to separate English sentences.

4.Average word length:

Based on my research I found out that average word length of English words are around 5 but for Dutch the average word length is slightly higher so took average word length > 5.

5.Presence of ij in word:

Based on my research I found out Dutch has ij in most of the words but it's very rare in English, so I took this attribute.

6.Presence of dutch diphthongs:

From the wikipedia page of Dutch language found out these diphthongs are usually found in Dutch language: ae, ei, au, ai, eu, ie, oe, ou, ui. So these can be used to separate Dutch sentences.

7.Words starting with s:

Based on manual inspection I found out that most of the English words start with the letter s.So took this feature.

8.Presence of English letters:

Based on my research Dutch has characters other than alphanumeric (0-9 and (A-Z) so put a feature to separate sentences that have words with other characters.

9.Presence of y in word:

Based on my research I found out that frequency of y in English is 2.04% whereas frequency of y in Dutch is 0.06% so used this to classify sentences.

10.Words starting with d:

Based on manual inspection and research I found out that most of the words in dutch start with the letter d so took this feature to classify the sentences.

Decision Tree:

In my implementation of code, the decision tree uses Information gain to get the best attribute among all to classify the data. Information gain uses entropy to find the maximum difference of entropies of root node and its children that's because entropy is nothing but randomness in data the less entropy we have the less randomness so we choose attributes that will reduce the entropy than previous source entropy. The tree ends with the following 3 conditions:
1.If all the examples in the data are of the same classification, that is the entropy of the data is zero then return the classification.
2.If there are no attributes to further classify the data then return the majority value in the data.
3.If data to classify is empty then return the majority value in the parent dataset.

Based on the prediction the following attributes are chosen based on the information gain in order of decreasing information gain:
1.Common Dutch words.
2.Repetition of vowels.
3.Common English words.
4.Presence of ij in word
5.Presence of y in word.
6.Average word length

Ada Boost:

Ada Boost as the name suggests it boosts the classification of weak learning classifiers. It is an ensemble of many weak classifiers which together form a strong classifier. In my implementation of the code weak classifier is taken as a stump(decision tree of depth 1). Initially each example in the dataset is given an equal normalized weight Each stump classifies the data and the wrongly classified data has their weights increased accordingly and the next stump is chosen such that the wrongly classified data is classified correctly using weights. Each tree

has an associated weight based on wrong classification and the final classification is the weighted majority of all stumps and their weights.

Parameters:

All of the following are tested on 1969 lines of mixed english and dutch sentences:

For 250 examples in training set these gave the best results:
depth of tree: 5 --> accuracy: 97.25749111223972
no of stumps: 7 --> accuracy: 98.9842559674962

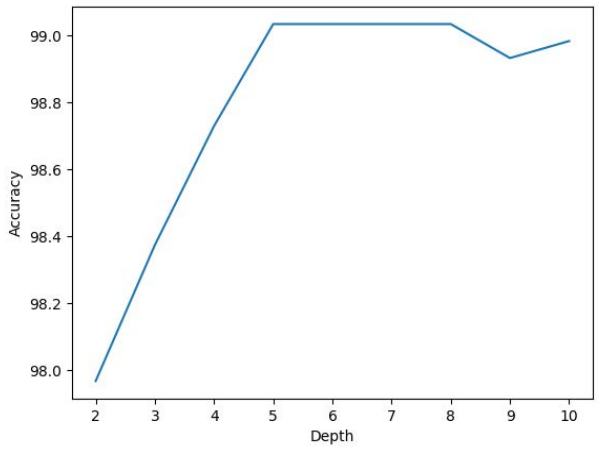
For 500 examples in training set these gave the best results:
depth of tree: 6 --> accuracy: 98.67953275774505
no of stumps: 6 --> accuracy: 98.17166074149314

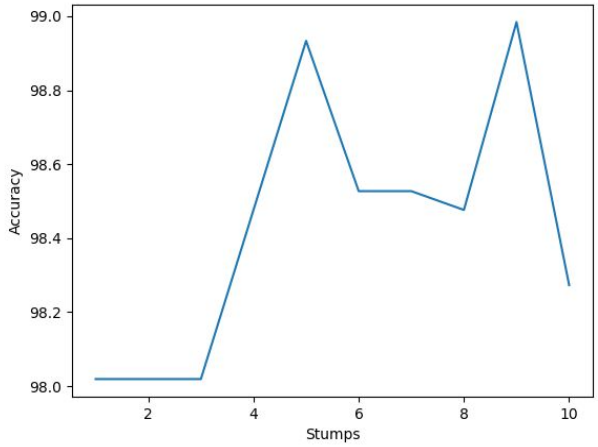
For 750 examples in training set these gave the best results:
depth of tree: 5 --> accuracy: 98.88268156424581
no of stumps: 6 --> accuracy: 98.88268156424581

For 1000 examples in training set these gave the best results:
depth of tree: 6 --> accuracy: 99.03504316912138
no of stumps: 6 --> accuracy: 98.52717115286947

For 1753 examples in training set these gave the best results:
depth of tree: 6 --> accuracy: 98.9842559674962
no of stumps: 6 --> accuracy: 98.47638395124429

On an average **decision tree of depth 6 and 6 stumps for ada boost** gave best results and fixed 1000 examples as my final training set. These are the following graphs for 1000 examples




99.0
98.8
98.6
98.4
98.2
98.0
Accuracy
2
4
6
8
10
Stumps