Harry Potter Pages Classification Using a Naïve Bayes Algorithm

The Algorithm

The Naïve Bayes Algorithm uses a list of probabilities and Bayes Rule to classify data into groups.

The idea of this project was to classify sets of pages from the harry potter books into the books they belong to.

We start off by splitting all the books into page sets (We will explain later how we optimized the `pageSetSize` which is the number of pages in a set). We then took around 20% of these from each book and set them aside for validation data which we used for training our hyperparameters, we set a further 20% aside for testing and the remaining 60% we used for training data.

We removed all punctuation from the data, except for question marks and exclamation marks which were separated into their own words as we felt this may give more insight as J.K. Rowling may have asked more questions or written more exclamations in some books than others.

We then took our training data and we calculated the probability for each word appearing in any one page set from each book. We did this by counting the number of page sets a word appears in in each book and dividing it by the total number of page sets in that book. This gave us our probability tables, each of which contain probabilities for around 20 000 words. Each of the tables contains a different number of words because of how the data is split up and which data the trainer sees and which is used for testing (see folder probTables where we have stored each of our tables).

In order to narrow down our dictionary (to improve both speed and accuracy) we decided to remove common words that don’t differentiate well between books. We did this by comparing the probabilities for each word of it appearing in each book and if at least one of these seven probabilities where greater than `upperThreshold` and at least `requiredNum` probabilities are less than `lowerThreshold` we then kept the word. By doing this we only keep words that appear often in one or more books and barely appear in multiple other books.

Using the validation data we optimized the hyperparameters `pageSetSize`, `upperThreshold`, `lowerThreshold` and `requiredNum`. We did this using the following algorithm

for each pageSetSize ranging from 1 – 13

probailityTable = create probability table based off pageSetSize using trainingData

for each requiredNum in range 1 - 5

currTable = probailityTable with words removed based off upperThreshold, lowerThreshold and requiredNum

while accuracy on validation data is improving

while accuracy on validation data is improving

Decrease lowerThreshold by alpha

currTable = probailityTable with words removed based off new upperThreshold, lowerThreshold and requiredNum

Retest on validation data

while accuracy on validation data is improving

Increase upperThreshold by alpha

currTable = probailityTable with words removed based off new upperThreshold, lowerThreshold and requiredNum

retest on validation data

Change alpha by a factor of a half

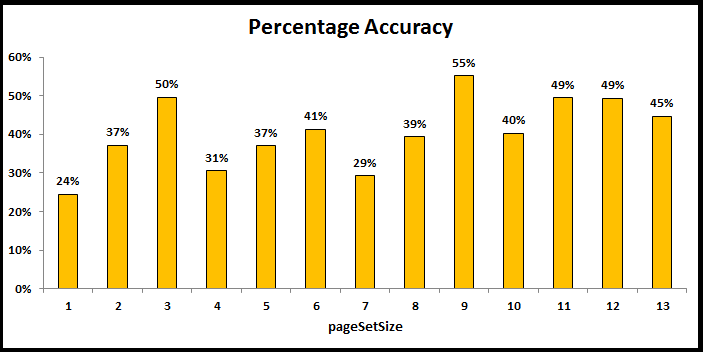
If alpha is less than a threshold exit the loop

take the pageSetSize, upperThreshold, lowerThreshold and requiredNum that yield the best accuracy

test on testing data

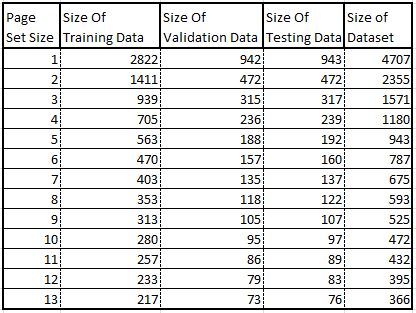
Results

The following graph shows the accuracy given for the different pageSetSizes on our test data, it should be noted that since there are 7 options a random guess will be correct 14.29% of the time.



It is quite clear a pageSetSize of 9 gives the highest accuracy of 55%, we however feel that the pageSetSize of 3 pages with an accuracy of 50% is the best result as it balances as few pages as possible with the best accuracy possible. In addition despite a pageSetSize of 1 only having a 24% accuracy this is nearly double random and we feel it is still noteworthy and meaningful that based off a single page we can classify with that accuracy.

It should also be noted that because we have a limited number of pages (there are 4707 pages in the series) when we increase the pageSetSize it decreases our dataset size. This means that the higher pageSetSizes have less training data and less testing data so the results aren’t as trustworthy. We believe this is the reason that the percentage accuracy doesn’t change as drastically over the last few pageSetSizes. The following table gives the breakup of pageSetSize and dataset size.



Thus the best classifier that we found was trained using the following parameters:

pageSetSize = 3

requiredNum = 2

upperThreshold = 7.05 %

lowerThreshold = 0.9 %

Trained on 939 pageSets

Validated on 315 pageSets

The classifier uses 84 unique words to classify the pages, for interest’s sake these 84 words are:

lying roars strange sleep results word december uses pages devon mrs strongly hung enclosed disgruntled burn eclairs fan hocus-pocus hills elastic vanish thunderous require fainted fix embrace victim guessed w-what’s soggy emitted chill unbelievablel curved sealed fang’s closing pomfrey’s plumbing fudge champion dementors pettigrew lupin hogsmeade dementor patronus tournament black’s firebolt lupin’s wormtail crookshanks buckbeak merely cedric diggory cho eaters triwizard eater moody moody’s madame maxime champions karkaroff viktor krum tent skeeter durmstrang fleur bagman beauxbatons ludo luna umbridge umbridge’s locket slughorn horcruxes horcrux

The following is the confusion matrix we got after testing this classifier on 317 test cases

| 1 | 2 | 3 | 4 | 5 | 6 | 7 |

----|-------|-------|-------|-------|-------|-------|-------|

1 | 14 | 9 | 2 | 5 | 11 | 4 | 11

2 | 1 | 3 | 1 | 1 | 1 | 3 | 0

3 | 0 | 1 | 23 | 3 | 3 | 0 | 6

4 | 0 | 1 | 1 | 30 | 5 | 3 | 2

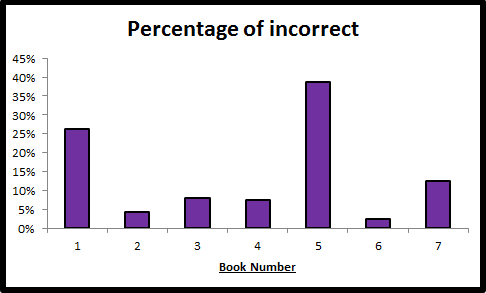
5 | 8 | 12 | 5 | 9 | 52 | 13 | 15

6 | 0 | 0 | 0 | 1 | 0 | 15 | 3

7 | 1 | 0 | 1 | 5 | 2 | 11 | 20

This confusion matrix has an accuracy of 49.53%. There a quite a few interesting observations that can be made about the harry potter books from this matrix.

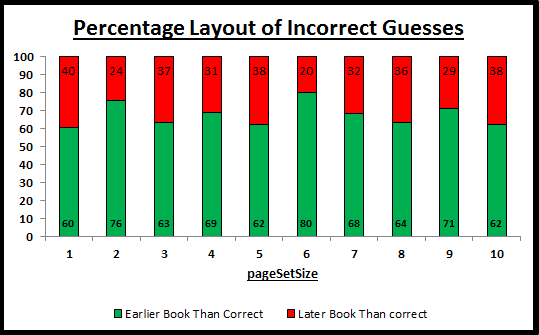
The first thing that jumps out as obvious in this matrix (and is reflected in all the others) is an obvious bias to book 1 and book 5.



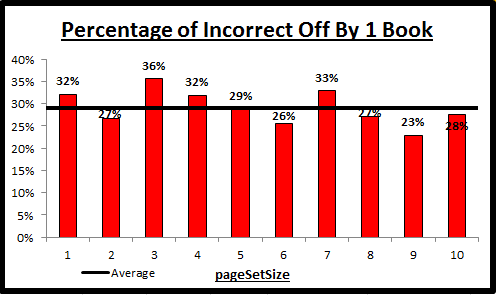
This could indicate multiple things. The bias towards book 1 is most likely because a lot is most likely introduced in book one that is referenced in later books. The bias towards book 5 could be because book 5 is the largest of all the harry potter books. It could also mean that in book 5 there are a lot of references and similarities to all of the earlier books and for this reason they are easily confused with book 5 and the classifier chooses book 5 over them since it is nearly double the size of the earlier books and thus more likely. See the following Table:

C:\Users\Dell-User\Documents\Wits\Third Year\Coms\Machine Learning\Project\numBookPages.PNG

Another interesting aspect of this confusion matrix is that other than the bias to book 5 it is nearly an upper triangular matrix, with majority of the incorrect guesses being in the upper triangle. This means that when it chooses incorrectly the classifier is more likely to pick an earlier book than it is to pick a later one. This result is to be expected as it means the later books are more similar to the earlier books than the earlier books are to the later ones. This is most likely because names of people and places, jargon and ideas are introduced in one book and carried through to the later books but all the books before the one in which this was introduced will have no reference to it. This is not only evident in this matrix but in all the other matrices as well, see the following graph which shows the split up of incorrect guesses between those that where later than the correct book and those that where earlier, as you can see majority of the time we classify as an earlier book rather than a later book.



A further surprising idea seen from the confusion matrices is that you would expect that when a page set is classified incorrectly it is most likely to be classified as either the previous book or the following book. The confusion matrices show that on average 29% of the incorrect guesses are classified as either the previous or following book. This however is very close to random as the probability of a page set being any one book is 14%, so the probability of it being any two is 28%, thus not much information can be derived from this.



Interestingly however, it seems from the confusion matrices that book 6 in particular is easily confused with and thus very similar to books 5 and 7.