Horror Authors Text Classification

Kieran Baker

Executive Summary

This is an analysis of the 'Spooky Author Identification' dataset from Kaggle.com, which consists of a list of sentences from the works of three horror authors, along with an explanation of the steps to develop a model which can reliably predict which sentence belongs to which author.

In an attempt to keep this report from being too long, only basic natural language processing techniques were used for feature creation, with the final goal being to train two models, one linear and one non-linear, to see which obtains the greater accuracy. Most of the exploratory analysis and plot creation was done in Python (Spyder) and the model training in Microsoft Machine Learning Studio.

Initial Data Analysis

	id	text	author
0	id26305	This process, however, afforded me no means of...	EAP
1	id17569	It never once occurred to me that the fumbling...	HPL
2	id11008	In his left hand was a gold snuff box, from wh...	EAP
3	id27763	How lovely is spring As we looked from Windsor...	MWS
4	id12958	Finding nothing else, not even gold, the Super...	HPL
5	id22965	A youth passed in solitude, my best years spen...	MWS
6	id09674	The astronomer, perhaps, at this point, took r...	EAP
7	id13515	The surcingle hung in ribands from my body.	EAP
8	id19322	I knew that you could not say to yourself 'ste...	EAP
9	id00912	I confess that neither the structure of langua...	MWS

Illustration 1: Examining the initial data set

The author column contains the initials of each author, which are:

- EAP – Edgar Allen Poe: 19th century American author famous for stories such as *The Raven* and *The Tell-Tale Heart*, regarded as the inventor of the detective story and as important to the development of science fiction, he even died under mysterious and somewhat tragic circumstances.
- HPL – Howard Phillips Lovecraft: Early 20th century author mostly famous for his stories centering around the so-called Cthulu Mythos (Cthulu being a giant squid-like monster), extremely influential on later science fiction authors thought like Poe he never achieved widespread recognition in his lifetime and died in near poverty.
- MWS – Mary Wollstonecraft Shelley: 19th Century British novelist and travel writer, best known as the author of *Frankenstein*, being friends with Lord Byron and her

marriage to Percy Bysshe Shelley. Unlike the others she was from a wealthy family, though her life was also fairly tragic.

Examining the distribution of the three authors among the sentences in the data set, we see that though their contributions are not equal, the difference is negligible given the large number of entries.

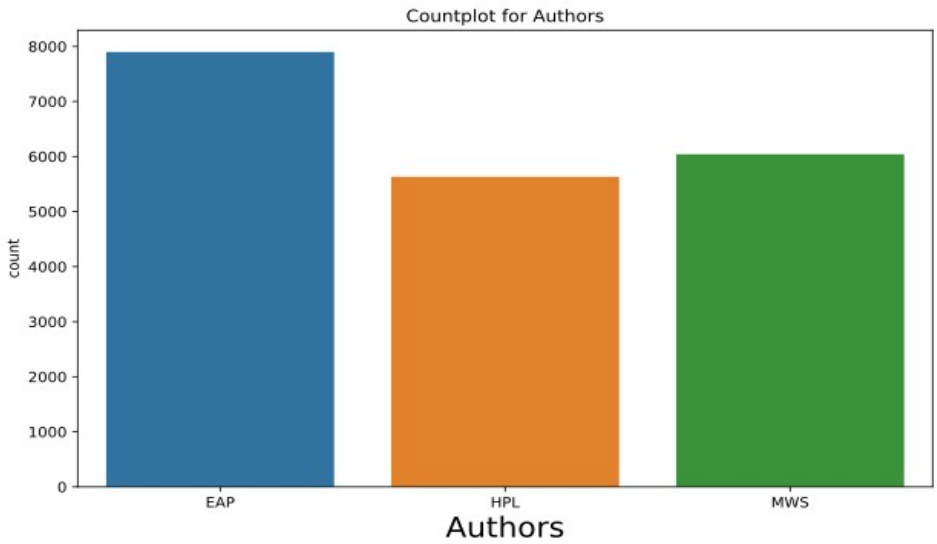

Illustration 2: The three authors contributed roughly equal proportions of the total data set

Preprocessing

Before going any further with examination of the data it is necessary to perform what is called preprocessing. If we were to graph the top ten or even fifty words used by each author using the raw data we would find them to be nearly all the same – words like ‘a’, ‘the’, ‘there’, ‘then’, known as stopwords. Since these words do not serve to distinguish the authors they are essentially junk data and so must be removed. The other important step in preprocessing this data set is lemmatization, in which each remaining word (after stop words have been removed) is reduced to a stem (i.e. ‘wins’, ‘winning’, ‘winner’ would all be reduced to ‘win’) using an existing dictionary known as a lemma. These and other more minor preprocessing steps are handled by the preprocessing text module in Machine Learning Studio.

Correlation and Relationships

Word Clouds using Preprocessed Text Data

Having performed the preprocessing we can start to get an idea of how the vocabularies of each author are different (in case it’s not clear the word clouds are supposed to be shaped like a raven and the heads of Cthulu and Frankenstein):

Edgar Allen Poe Word Cloud

H.P. Lovecraft Word Cloud

Mary Shelley Word Cloud

The word clouds don't exactly help with building the model, but they do show a clear difference in the words used by each author thus indicating an approach based on categorizing sentences using the author's word choice may be successful. It is also good to note that the words seem to correspond to the themes and topics favoured by each author, if you are already familiar with their work.

Analysis

Vectorization

To perform actual analysis, we must first vectorize the text data. Using the 'Extract N-grams Features from Text' module the column of preproccessed text is converted into many columns, one for each word (1-gram) and for each combination of two or three consecutive words (2- and 3-grams). These columns have values corresponding to the number of times the N-gram appears in each sentence.

In an attempt to keep this report from getting too long, we will only use and compare the results of two models, linear (multiclass logistic regression) and non-linear (multiclass decision jungle).

Linear Model

The multiclass logistic regression module was used, with hyperparameters determined by the tune model hyperparameters module using random grid search on the default logistic regression parameters. Results were compared for accuracy using the RMSE metric. Cross-validation was not used. This model achieved an average accuracy of 0.859948.

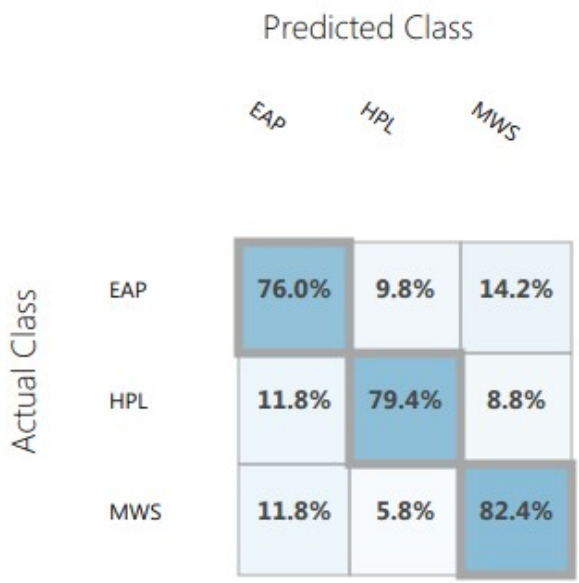

Illustration 3: Confusion matrix for linear model

Non-Linear Model

Similarly to the logistic regression, we used the tune model hyperparameters module random grid search on the default parameters, testing for RMSE, without cross validation. This model performed extremely poorly, with an average accuracy of 0.616502.

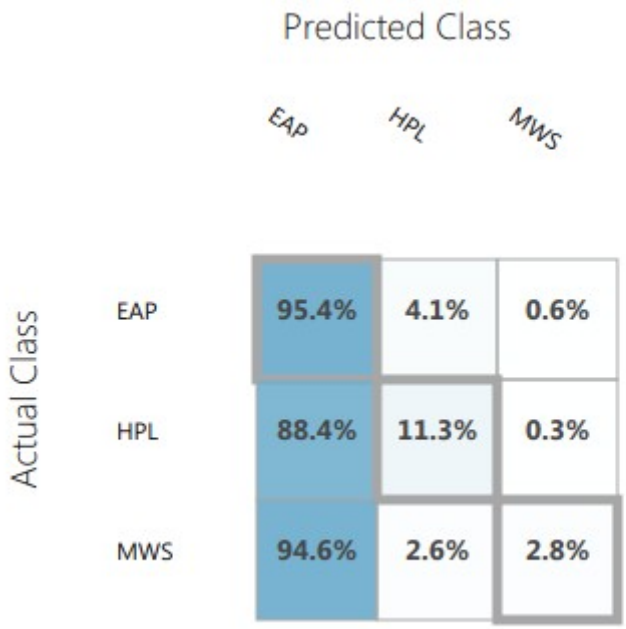

Illustration 4: Confusion matrix for non-linear model

Conclusion

The linear model vastly outperformed the non-linear model. It is possible that this is because the relationship being modelled really is mostly linear, in a situation like this the decision jungle would perform worse for larger number of features (and there are thousands in this model), but it is more likely that the large number of features has simply caused the decision jungle to overfit the training data, a problem which the simpler linear model is less prone to. Thus a process of feature selection on the N-grams before model training would likely increase the accuracy of the jungle model. This would probably be the first step of any further analysis.

Better parameter tuning methods such as cross validation or sweeping the entire grid of parameters would also likely increase the accuracy of both models.