Spooky Author Identification – project proposal

Problem attempting to solve

The aim of this Halloween-themed project is to predict the author of horror stories passages based on their writing. In particular, we will be analyzing texts from Edgar Allan Poe, Mary Shelley, and HP Lovecraft with the goal to predict the probability for each excerpt to be written by one of these figures of harrowing horror.

Importance of problem/solution

There are many applications for NLP such as search results, spam filtering, sentiment analysis, chat bots and digital assistants, language identification/ translation/ generation. Moreover, solving this specific author identification challenge might help uncover anonymous authors or provide readers with recommendations of authors with similar writing style. In addition, might help discover plagiarists, fake news and/or uncover writings with malicious intent therefore helpful in forensic investigation cases.

Data Source

Data was sourced from Kaggle: https://www.kaggle.com/c/spooky-author-identification. Dataset contains text from works of fiction written by: Edgar Allan Poe (EAP), HP Lovecraft (HPL) and Mary Wollstonecraft Shelley (MWS). Kaggle states that, because the “data was prepared by chunking larger texts into sentences using CoreNLP's MaxEnt sentence tokenizer,” occasional incomplete sentences may appear in the samples. There are separate files for training and test: train.csv and test.csv as well as a sample_submission.csv. The latest file contains the probability per author (same probabilities across all samples, so that EAP is always 0.40, HPL is always 0.29, and MWS is always 0.31) which matches the percentage of samples that each author has in the total training dataset of 19,579 samples.

What techniques from the program will be used?

I will approach the project with the typical steps involved in working with text data: exploratory data analysis (EDA) and visualization, text processing, feature extraction, and modeling. For text cleaning I will apply: lowercasing, replacing punctuation with spaces, removing stop words, eliminating infrequent occurrences (such as less than 5 or 10 words), stemming, and lemmatization. For feature extraction I plan to use BoW (bag of words) model and TF-IDF vectorizer (Term Frequency and Inverse Document Frequency). With pre-trained word embeddings, I’d probably experiment with word2vec, GloVe, fastText, ELMo, and CoVe and evaluate what works best. For modeling I will be checking several machine learning classification models as well as deep learning CNNs, plain RNNs, LSTMs, and/or GRUs from Keras. Each model will be evaluated based on the logloss using either 5-fold or 10-fold cross validation; the lower the logloss, the better the model. Also, for the top performing algorithms, I plan to run random search to tune certain hyperparameters.

If time permits, I might try to perform text generation (Markov chains & Gensim).

What challenges I'll be facing

First will be the time restriction: I might not be able to do everything I intend to do. At the same time, computation power might be a challenge. Also, I might not find the best algorithms and the results might be lower than expected. However, this is a fun project which will allow me to go over learning and implementing NLP techniques as well as applying supervised ML and deep learning models.