Project 1 – Naive Bayes Classification – Proposal

Evan A Bonsignori, Jordan Brown

9/27/2017

**1. Classification Proposal:**

We seek to classify books into two similar and often difficult to distinguish genres: Science Fiction and Fantasy. Our features will be word occurrences (occurs or doesn’t occur) among the 2000 most frequently occurring words in each book (our training example).

**2. Corpus:**

We have a library of over 200 science fiction and fantasy eBooks and have organized them by genre (at least 100 of each) and converted them into .txt files to ease preprocessing.

**3. External Corpora:**

We will be using the NLTK corpus of common English stop words for cleaning the data in the preprocessing phase.

**3. Architecture:**

The project will be implemented in Python 3.6.1 and use the following standard libraries: os, random, and time. See Page 2 for project diagram.

**5. API’s:**

In addition to removing stop words, **NLTK’s** method *word\_tokenize* will be used to tokenize books and the class *FreqDist* to find the tokenized word frequencies. The **Pickle** module will be used to save objects in memory from python program execution to a file in binary format. This will allow us to save the classifier and cleaned data into binary files, which will greatly reduce computation. Furthermore, **NLTK’s** *NiaveBayesClassifier* class will be used for classification.

**6. Performance:**

The performance specification for our machine learning scenario is genre classification accuracy. We will separate 15% of our data to be used as labeled test examples. After running the 15% through the trained classifier, we will label the false positives, true positives, false negatives, and true negatives for accuracy and error rate calculations. We will consider Python graphical graphing utilities for portraying the ROC curve.

**7. Team Status:**

Team members will accomplish each task in pair program.

