

Project 5

ReadMe

Mike Roylance
roylance@uw.edu

Language coded	Java						
Approach	<p>I created 3 classes to help me complete this project</p> <table> <tr> <td>App.java</td><td>This was in charge of doing the File IO to bring in the words and their count and then the sentences. This also prints out the result of each word</td></tr> <tr> <td>BayesianReporter.java</td><td>This class was in charge of counting up the probabilities for each sentence. It takes in all the languages (in a nested HashMap) and introduces Laplace smoothing for calculating the probability of a word belonging to a particular language.</td></tr> <tr> <td>Tests.java</td><td>This class tests the logic in Bayesian Report. I only needed a few tests to verify my functionality</td></tr> </table>	App.java	This was in charge of doing the File IO to bring in the words and their count and then the sentences. This also prints out the result of each word	BayesianReporter.java	This class was in charge of counting up the probabilities for each sentence. It takes in all the languages (in a nested HashMap) and introduces Laplace smoothing for calculating the probability of a word belonging to a particular language.	Tests.java	This class tests the logic in Bayesian Report. I only needed a few tests to verify my functionality
App.java	This was in charge of doing the File IO to bring in the words and their count and then the sentences. This also prints out the result of each word						
BayesianReporter.java	This class was in charge of counting up the probabilities for each sentence. It takes in all the languages (in a nested HashMap) and introduces Laplace smoothing for calculating the probability of a word belonging to a particular language.						
Tests.java	This class tests the logic in Bayesian Report. I only needed a few tests to verify my functionality						
Smoothing	<p>I decided to use LaPlace Smoothing for this problem because I find that method effective and easy to implement. Here is the algorithm:</p> $\hat{\theta}_i = \frac{x_i + \alpha}{N + \alpha d} \quad (i = 1, \dots, d),$ <p>On line 18 of BayesianReporter, I am adding in the denominator portion (1 for every word).</p> <p>On lines 52 and 60, I'm adding in the numerator addition (+1). These ensure that I will never have a probability of 0.</p>						
Results	Please look at output.txt created from condor.cmd for more information						