Program #4

Prof. Rivka Levitan

November 29, 2015

Due date

This assignment is due at 11:59pm on Monday, December 7. Four total late days are available to you for this course. One point will be deducted for each unexcused late day. Homework submissions will not be accepted after 11:59pm on Monday, December 14.

Description

Write a program to calculate bigram probabilities from a corpus.

A bigram is a pair of words. For example, the bigrams in the previous sentence are (A, bigram), (bigram, is), (is, a), (a, pair), (pair, of), and (of, words). Bigrams can also be word pairs that are unlikely or impossible to exist in English, such as (of, of).

Bigram probabilities are important in many natural language applications such as voice recognition, natural language generation, and machine translation. They can be interpreted as "how likely is it that word 2 will follow word 1?" They are usually computed from large corpora by the following formula:

$$\frac{freq(w1, w2)}{freq(w1)} \tag{1}$$

`calc_bigrams.cpp` is provided for you. Fill in the program to calculate bigram probabilities for every pair of words in a document. You will need two maps: one to keep track of how many times each pair of words occurs (for the numerator in (1)) and one to keep track of how many times each word occurs (for the denominator). You may use the STL map.

I will test your code on the documents in the "gutenberg" corpus provided by nltk (linked from the class website). You will know you've implemented the formula correctly if your program outputs `0.0181488` when run on `whitman-leaves.txt`, `0.00440529` when run on `austen-emma.txt`, and `0` when run on `shakespeare-macbeth.txt`.

Submission

Please use **Blackboard** to submit this homework. Your submission should consist of one file only: your modified `calc_bigrams.cpp`. Put your name in the "Name" comment in line 1 of the program. Do not change the program filename.