A SIMPLE STYLOMETRIC COMPARATOR*

NIFTY ASSIGNMENT

Steven Benzel
Division of Mathematics
Georgia Highlands College
5441 Highway 20, NE
Cartersville, GA 30121
sbenzel@acm.org

Stylometry is the study of linguistic style and a common concern is the determination of authorship of a written work. We present a very straightforward syntactical comparator for two works which is surprisingly useful in predicting authorship. The comparator is simple enough that it can be introduced in an introductory data structures class. To begin, let N be a small positive integer, τ the set of tokens consisting of the words in the English language, and *T* a stream of such tokens, for example a text in English stripped of punctuation. We can then construct an associative array A with key:value pairs consisting of token sequences of length N along with their frequency in *T*. For example, if we take N=3 with the text Huckleberry Finn by Mark Twain and we sort A by highest to lowest values, the first 10 entries of the array will be:

83 by and by
61 out of the
53 was going to
49 there was a
47 all the time
47 it was a
42 the old man
38 said it was
34 a couple of
34 a lot of

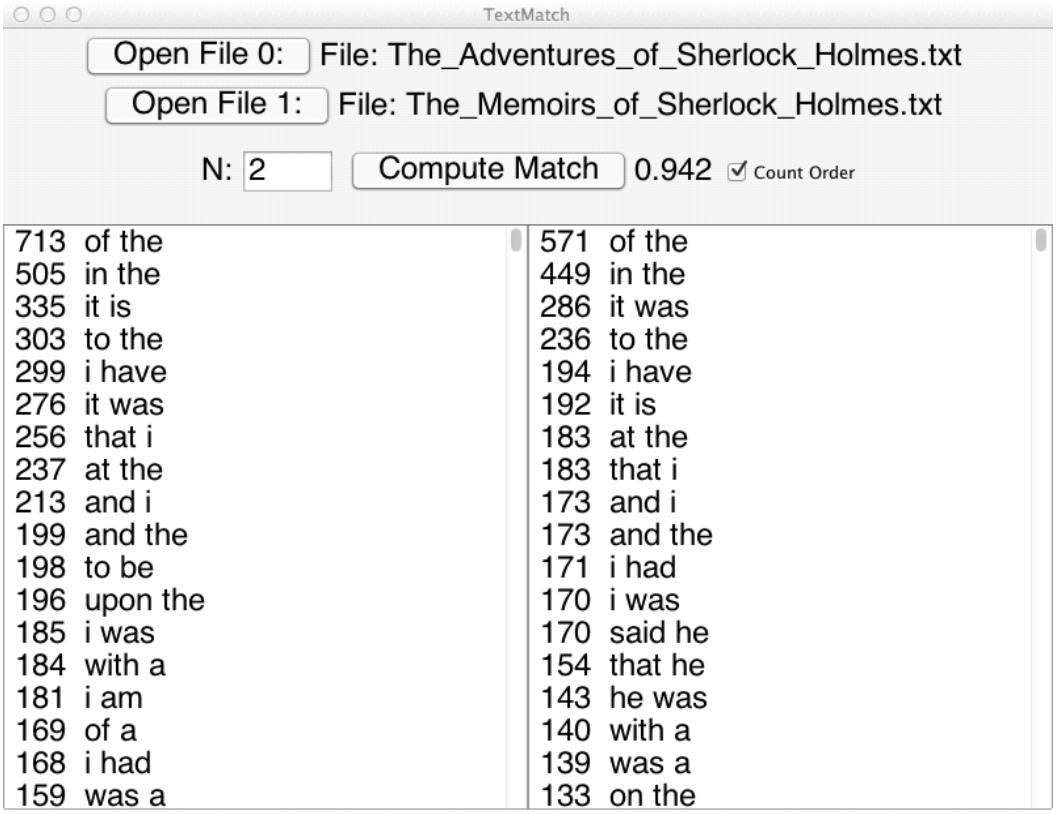
We define the norm of A to be the square root of the sum of the squares of the values. Given two streams T_1 and T_2 with the corresponding associative arrays A_1 and A_2 we define the comparison $c(A_1,A_2)$ to be the sum over all keys of the corresponding value from A_1 times the corresponding value from A_2 divided by the product of the norms. Mathematically this is the normalized dot product of two vectors from a formal

vector-space and geometrically is the cosine of the angle between the two vectors. We always have

$$0 \leq c(A_1, A_2) \leq 1$$

with the value 1 if A_1 and A_2 are identical and value 0 if A_1 and A_2 share no N-token sequences (are orthogonal.) If the arrays are ordered by the lexicographical ordering of the keys the comparison can be computed in linear time.

If the comparison of two texts is close to 1 the two texts can be considered syntactically similar in style and this could indicate common authorship. A very pleasant example with N=2 is provided by comparing two works of Sir Arthur Conan-Doyle with the match 0.942 corresponding to an angle of 2°:



Associative arrays are perhaps not the first data structure developed in an introductory class. Fortunately streams can be processed and comparisons computed using more basic structures. The Ordered List structure is well suited for processing streams but processing will be of quadratic complexity and will indeed take time on novels. A nice set of examples in this case is the set of presidential speeches with an interesting question: Can we predict which presidents wrote their own speeches? If the Binary Search Tree structure is available processing reduces to log-linear complexity and novels can be efficiently processed. A nice example problem in this case is the mystery text problem: students are given four novels by four different authors (for a total of 16 novels) and one mystery novel guarantied to by written by one of the four. The challenge is of course to predict who wrote the mystery text.