CSE 5243 HW5

Name: Neng Shi

Methods

Feature Extraction

I used a binary feature, i.e, whether a word appears in the sentence or not. For the pre-processing, first transform all the words into lowercase, then discard stop words.

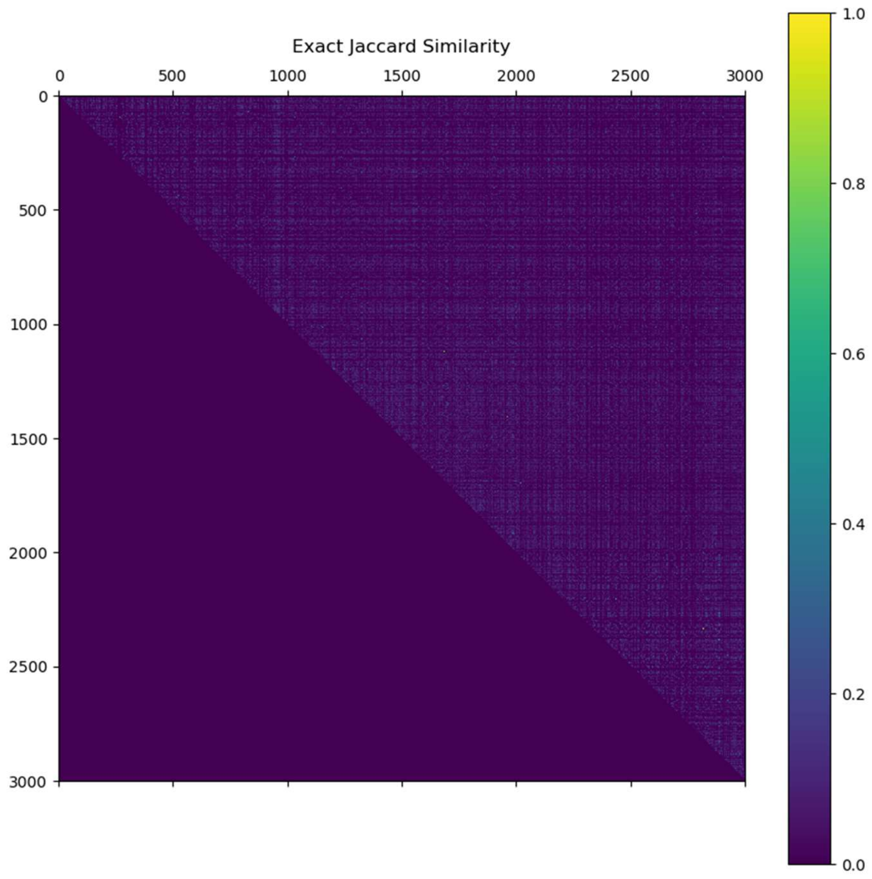
k-MinHash Signature

The radndom hash function takes the form of h(x) = (a*x + b) % c, where 'x' is the input value, 'a' and 'b' are random coefficients, and 'c' is a prime number just greater than the number of unique words in all the sentences.

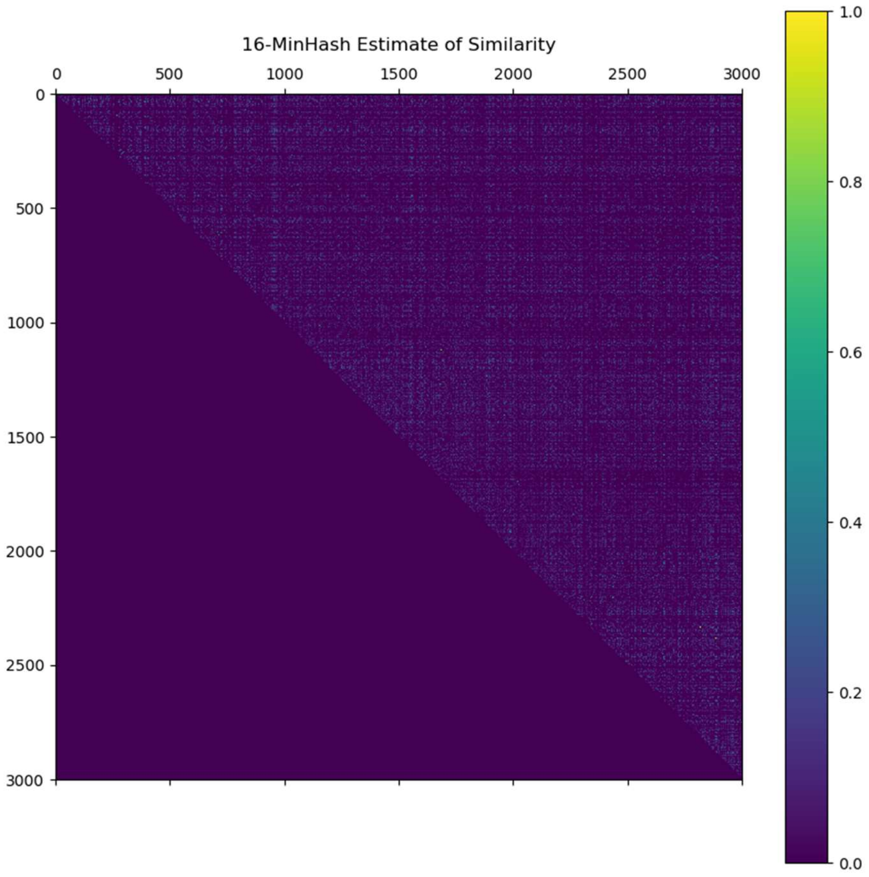
We generate a list of 'k' random coefficients for the random hash functions. Rather than generating a random permutation of all possible words, I just hash the IDs of the words that are *actually in the sentence*, then take the lowest resulting hash code value. Finally, the similarity of two signatures is the fraction of the hash functions in which they agree.

Results

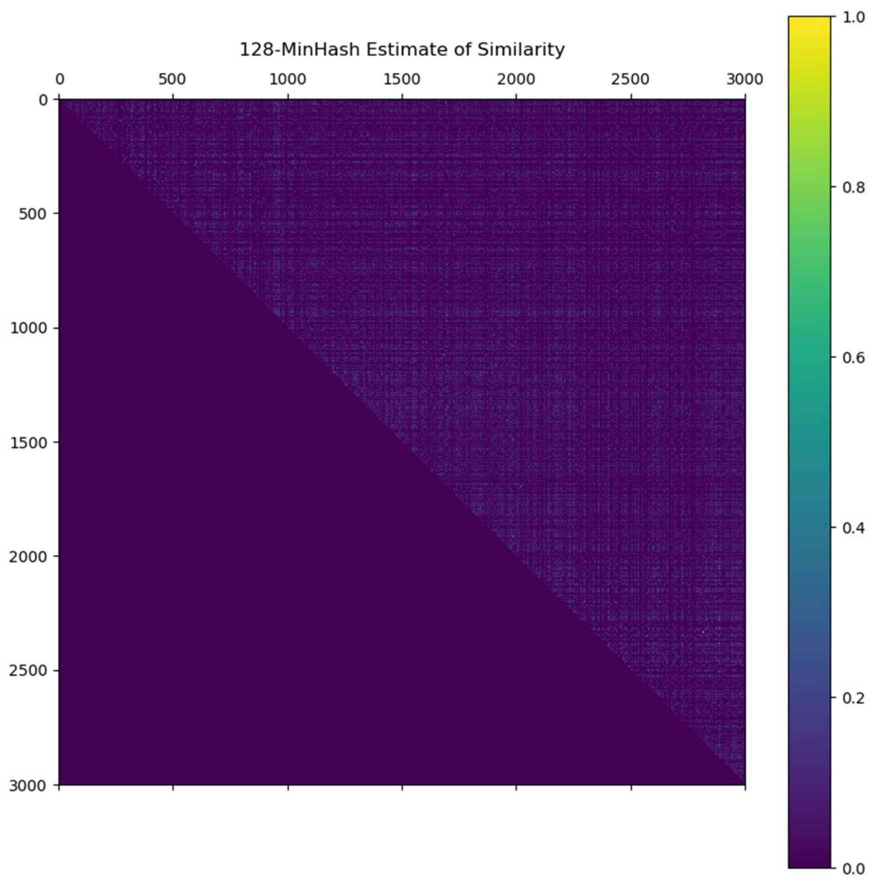
Exact Jaccard Similartity



16-MinHash Estimated Jaccard Similarity



128-MinHash Estimated Jaccard Similarity



The running time for the three similarities is listed below:

Method	Running Time (s)
Raw binary feature	75.41
16-MinHash	14.62
128-MinHash	15.00

The mean-squared error for the two estimated similarities is listed below:

Method	Running Time (s)
16-MinHash	0.000828
128-MinHash	0.000104

Acknowledgments

The “Indexer” class of the code is from homework one of CSE 5525.

The k-MinHash Signature implememation is adopted from the following blog: https://chrisjmccormick.wordpress.com/2015/06/12/minhash-tutorial-with-python-code/.