Algorithmic Foundations of Data Science: Assignment #3

Khaled Al Hosani (kah579), Myunggun Seo (ms9144)

New York University Abu Dhabi
Date: February 19, 2018

Problem 1

From a universal set U, we choose two sets S and T at random and with size m for both. Let $|S \cap T| = k$ such that $0 \leq k \leq m$.

There are $\binom{n}{m}$ choices for set S

There are $\binom{m}{k}$ $\binom{n-m}{m-k}$ choices for set T, this is because you need to choose k elements from the m elements in S. Then, pick the remaining $m-k$ elements from the $n-m$ elements that did were not chosen for S from the universal set.

The probability $P(|S \cap T| = k) = \frac{\binom{m}{k}\binom{n-m}{m-k}}{\binom{n}{m}}$

The Jaccard Similarity is $\frac{|S \cap T|}{|S \cup T|} = \frac{k}{2m-k}$

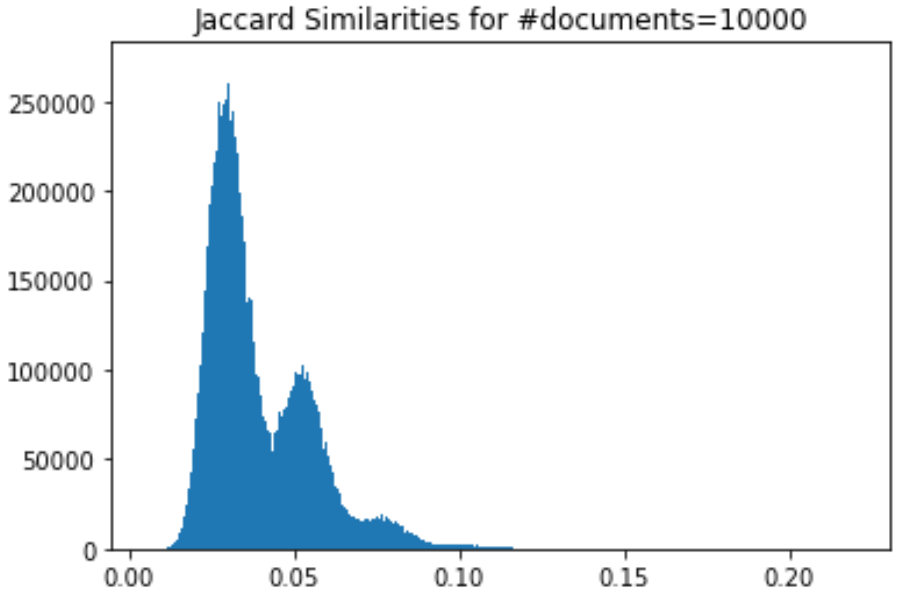
The expected value of SIM(S,T) is $\sum_{k=0}^{m} \frac{k}{2m-k} \frac{\binom{m}{k}\binom{n-m}{m-k}}{\binom{n}{m}}$

Problem 2

Please see the attached files '`preAnalysis.py`', '`functions.py`', and '`main.py`'.

Conclusion: We found that documents 197904 and 704395 are similar with JS = 0.9487 and documents 322400 and 370208 are similar with JS = 1.0000.

Methodology: The naive approach would be to compute the Jaccard Similarity for all document pairs. We tried computing the JS for 10^4 documents and 10^8 document pairs, which took 2 hours. Considering that there are 10^{12} document pairs in our data, it would have taken 833 days to compute. The following histogram shows the distribution of values of JS for the 10^8 pairs. The highest JS value in this set was 0.2197 and the lowest was 0.0055. This tells us that the number of pairs that have $JS \geq 0.75$ is an extremely small percentage if not non-existent.



In order to speed up the process, we use Locality-Sensitive Hashing with banding strategy in order to compare the minhash values of 10^6 documents and produce a set of candidates that can have their JS manually computed before the due date.

k-Shingling 10^6 documents for k=5 took roughly 10 seconds. The shingles were normalized by removing whitespace and punctuation as well as converting to lowercase characters.
We hashed each shingle into 16 strings of 8 characters using SHA-512 and slicing the resultant 128 characters-long hash. We then calculate the minhash for each hash of the n=16 hash functions and saved them to a file. Hashing the shingles and computing the set of minhashes for each document took about 1.5 hours in total.

We used banding method with (r,b) = (2,8) because the threshold t=0.258 was high enough to ignore the majority of document pairs which had $JS \leq 0.2$ and thus led to a small number of false positives. Also, $f(0.75)$ for $f(s) = 1 - (1 - s^r)^b$ was 0.9987 which meant the possibility of false negatives was $1 - 0.9987 = 0.0013$. (We would have to be really unlucky to get a false negative, however, we learned from our peers that we DID get unlucky and our parameters missed a document pair. We examined the minhashes of the two documents and found that none of them matched.)

This produced a set of candidate pairs of size 30865, each of which had its Jaccard similarity manually calculated. This took approximately 15 minutes.

After which, two pairs were found with a Jaccard similarity that is ≥ 0.75.

Documents 197904 and 704395, with JS = 0.9487; and documents 322400 and 370208, with JS = 1.0000.