Group 25 Solution for Homework 1

How to build and run

Run `hw1.ipynb` with Jupyter Notebook at root directory.

Solution

Input text (as raw string) is first shingled with k=9, then we tried to print the jaccard similarity between these shingled sets. For larger dataset, we generated 200 hash functions and used minhash to calculate its minhash signature, and finally, we used LSH with band size of 40 to find similar text pairs.

Screenshots

Jaccard similarity between texts (stra, strb, and strc)

In [4]:
```
stra = "Russian Prime Minister Viktor Chernomyrdin on Thursday proposed a three-phase solution to end the three-month-old conflict in Chechnya,
strb = "Russian Prime Minister Viktor Chernomyrdin on Thursday proposed a three-phase solution to end the three-month-old conflict in Chechnya,
strc = "South Africa's monetary authorities will follow a restrictive monetary policy in 1995, the governor of the central Reserve Bank, Chris
```

In [5]:
```
my_shingling = sl.Shingling()
stra_shinglingSet = my_shingling.shingling(docs=stra)
strb_shinglingSet = my_shingling.shingling(docs=strb)
strc_shinglingSet = my_shingling.shingling(docs=strc)
```

In [6]:
```
compareSets = cSet.CompareSets()
shinglingSetA = set(stra_shinglingSet)
shinglingSetB = set(strb_shinglingSet)
shinglingSetC = set(strc_shinglingSet)
jaccard_similarity_ab = compareSets.compare(set_a=shinglingSetA, set_b=shinglingSetB)
jaccard_similarity_ac = compareSets.compare(set_a=shinglingSetA, set_b=shinglingSetC)
jaccard_similarity_ab, jaccard_similarity_ac
```

Out[6]:
```
(0.9869451697127938, 0.0023171135385633896)
```

Similar texts using LSH, from the result, we can see webpages from KTH are grouped together while pages from Stack Overflow are grouped together.

In [11]:

```
import os

datasetDir = "./dataset"
# more data
data = []
fileNames = []
for filename in os.listdir(datasetDir):
    if filename.endswith(".html"):
        fileNames.append(filename)
        with open(datasetDir + "/" + filename, "r", encoding='utf-8') as f:
            data.append("".join(f.readlines()))

# print(data[0][:100])

my_shingling = sl.Shingling()
shingledData = list(map(my_shingling.shingling, data))
minHash = mh.MinHashing(nHash, 1000000007) # 1e9+7
minHashSigs = list(map(minHash.signature, shingledData))

band = 40
threshold = 0.5 # (1/40)^(1/5) = 0.478

LSH = lsh.LSH()
pairs = LSH.findSimilarPairs(minHashSigs, band, threshold)
```

In [12]:
```
for (a, b) in pairs:
    print(fileNames[a], fileNames[b])
```

```
assignment1.html assignment2.html
assignment2.html assignment3.html
stackoverflow-software-transaction-memory.html stackoverflow-sorting1.html
stackoverflow-sorting1.html stackoverflow-transaction1.html
stackoverflow-sorting1.html stackoverflow-sorting2.html
assignment1.html assignment3.html
stackoverflow-sorting2.html stackoverflow-transaction1.html
stackoverflow-software-transaction-memory.html stackoverflow-transaction1.html
stackoverflow-software-transaction-memory.html stackoverflow-sorting2.html
```

In [ ]: