import re

import math

with open('Text.txt','r',encoding="utf-8") as fh:

data = fh.read()

data = data.lower()

lines = data.split("\n")

newLines = []

for line in lines:

newLine = re.sub('[^a-z]', '', line)

newLines.append(newLine)

#Calc the freq of each char at every line

freq = {}

freqInLines = {}

for line in newLines:

for i in line:

if i in freq:

freq[i] +=1

else:

freq[i] = 1

freqInLines[line] = freq

freq = {}

# Probability of each char in line

prob = {}

for line in freqInLines:

for i in freqInLines[line]:

prob[i] = freqInLines[line][i]/len(line)

freqInLines[line] = prob

prob = {}

# Entropy of each line

relEntropy=0

for line in freqInLines:

for i in freqInLines[line]:

p = freqInLines[line][i]

relEntropy += p\*math.log2(p)

freqInLines[line] = relEntropy

relEntropy = 0

orig = {}

ciph = {}

cnt = 0

for line in freqInLines:

if cnt>=30:

ciph[line] = freqInLines[line]

else:

orig[line] = freqInLines[line]

cnt += 1

linesCiph = []

for line in ciph:

linesCiph.append(line)

alphabet = {}

for line in orig:

for line2 in ciph:

if orig[line] == ciph[line2]:

for i in range(len(line)):

if i not in alphabet:

alphabet[line2[i]] = line[i]

print("The table of alphabet mapping between original and ciphered texts" )

for i in alphabet:

print(i, ':', alphabet[i])

deCiph = []

for line in linesCiph:

s=''

for i in line:

if i in alphabet:

s += alphabet[i]

deCiph.append(s)

print('Ciphered Number',' - '"Original Text")

for i in enumerate(deCiph,1):

print(i[0],'-',i[1])