from os import system, name

import re

def process\_files(filenames):

file\_to\_terms = {}

for file in filenames:

pattern = re.compile('[\W\_]+')

file\_to\_terms[file] = open(file, 'r').read().lower();

file\_to\_terms[file] = pattern.sub(' ',file\_to\_terms[file])

re.sub(r'[\W\_]+','', file\_to\_terms[file])

file\_to\_terms[file] = file\_to\_terms[file].split()

return file\_to\_terms

def index\_one\_file(termlist):

fileIndex = {}

for index, word in enumerate(termlist):

if word in fileIndex.keys():

fileIndex[word].append(index)

else:

fileIndex[word] = [index]

return fileIndex

def make\_indices(termlists):

total = {}

for filename in termlists.keys():

total[filename] = index\_one\_file(termlists[filename])

return total

def fullIndex(regdex):

total\_index = {}

for filename in regdex.keys():

for word in regdex[filename].keys():

if word in total\_index.keys():

if filename in total\_index[word].keys():

total\_index[word][filename].extend(regdex[filename][word][:])

else:

total\_index[word][filename] = regdex[filename][word]

else:

total\_index[word] = {filename: regdex[filename][word]}

return total\_index

def one\_word\_query(word, invertedIndex):

pattern = re.compile('[\W\_]+')

word = pattern.sub(' ',word)

if word in invertedIndex.keys():

return [filename for filename in invertedIndex[word].values()]

else:

return []

def free\_text\_query(string,index):

pattern = re.compile('[\W\_]+')

string = pattern.sub(' ',string)

result = []

for word in string.split():

result += one\_word\_query(word,index)

return list(set(result))

def phrase\_query(string, invertedIndex):

pattern = re.compile('[\W\_]+')

string = pattern.sub(' ',string)

listOfLists, result = [],[]

for word in string.split():

listOfLists.append(free\_text\_query(word,invertedIndex))

setted = set(listOfLists[0]).intersection(\*listOfLists)

for filename in setted:

temp = []

for word in string.split():

temp.append(invertedIndex[word][filename][:])

for i in range(len(temp)):

for ind in range(len(temp[i])):

temp[i][ind] -= i

if set(temp[0]).intersection(\*temp):

result.append(filename)

print('\n temp : \n')

print(temp)

return result

filenames=['d:/document1.txt','d:/document2.txt']

termslist=process\_files(filenames)

print('\nterm list \n')

print(termslist)

print('\n\n')

print('\n\n')

totaldict=make\_indices(termslist)

print('total dictionary \n')

print(totaldict)

print('\n\n')

print('\n\n')

index=fullIndex(totaldict)

print('full index \n')

print(index)

print('\n\n')

#one\_word\_query('exceptions', index)

#query\_word=free\_text\_query('exceptions',index)

#print(query\_word)

system('cls')

print('\n\n')

print('\n\n')

#r=phrase\_query('python has exceptions handling',index)

#print (r)