import re

from collections import Counter

def parse\_text(filename):

import re

from collections import Counter

res = Counter()

res1 = []

with open(filename, encoding="utf8") as f:

for l in f:

res1.extend(l.split(' '))

l = l[:-1]

for w in res1:

if re.search("[.;:,!\"/\\?'ï»¿'.\n]", w):

res1.remove(w)

for w in res1:

if len(w) <3:

res1.remove(w)

for w in res:

if len(w)>2:

if w[0].isupper():

res1.remove(w)

res = Counter(res1)

return res

print(res)

#parse\_text('./resources-01/kafka-metamorphosis.txt')

len\_multi(Counter(parse\_text('./resources-01/kafka-metamorphosis.txt')))

# Split `l` into words and increase their respective count in `res`

def jaccard\_multi(setA, setB):

multiUnion = len\_multi((setA&(setB)))

multiIntersection = len\_multi((setA|(setB)))

if multiUnion == 0:

if multiIntersection == 0:

jacMulti = 1

else:

jacMulti = 0

else:

jacMulti = multiUnion/multiIntersection

return jacMulti

for key, multi in texts.items():

norm = texts[key].most\_common(1)[0][1]

for w in multi:

multi[w] /= norm