## Python 文本相似度计算代码

import jieba

from sklearn.feature\_extraction.text import TfidfTransformer, TfidfVectorizer,

CountVectorizer

import numpy as np

import re

from zhconv import convert

def simCosine(x, y):

"""

:param x: m x k array

:param y: n x k array

:return: m x n array

"""

xx = np.sum(x \*\* 2, axis=1) \*\* 0.5

x = x / xx[:, np.newaxis]

yy = np.sum(y \*\* 2, axis=1) \*\* 0.5

y = y / yy[:, np.newaxis]

dist = np.dot(x, y.transpose()) # cosine similarity

return dist

file\_names = [

'华电国际 2011 社会责任报告.txt',

'华电国际 2012 社会责任报告.txt',

'华电国际 2013 社会责任报告.txt',

'华电国际 2014 社会责任报告.txt',

'华电国际 2015 社会责任报告.txt',

'华电国际 2016 社会责任报告.txt',

'华电国际 2017 社会责任报告.txt',

'华电国际 2018 社会责任报告.txt',

'华电国际 2019 社会责任报告.txt',

'华电国际 2020 社会责任报告.txt',

'华电国际 2021 社会责任报告.txt',

]

documents = []

for file\_i in range(len(file\_names)):

file = file\_names[file\_i]

if file\_i > 5:

encoding = "utf-8"

else:

encoding = 'gbk'

with open(file, encoding=encoding, errors='ignore') as f:

lines = f.readlines()

lines = ''.join(lines).replace('\n','').replace(' ','')

pattern = re.compile(r'[^\u4e00-\u9fa5]')

chinese = re.sub(pattern, '', lines)

simplified\_chinese = convert(chinese, 'zh-hans')

documents.append(simplified\_chinese)

documents = [" ".join(jieba.cut(item)) for item in documents]

vectorizer = TfidfVectorizer(token\_pattern=r'(?u)\b\w+\b', stop\_words =

[f'{i}' for i in range(10)] + ['的', '地', '得'])

X = vectorizer.fit\_transform(documents)

word\_dict = vectorizer.get\_feature\_names()

print(word\_dict)

tf\_idf\_mat = X.toarray()

print(tf\_idf\_mat)

dis\_mat = simCosine(tf\_idf\_mat, tf\_idf\_mat)

print(np.round(dis\_mat,3))