Table S1 Calls four different word segmentation methods

1:	import jieba
2:	words = jieba.lcut(text)
3:	import pynlpir
4:	pynlpir.open()
5:	segments = pynlpir.segment(text,pos_tagging=Fals,pos_names='all', pos_english=False)
6:	pynlpir.close()
7:	from snownlp import SnowNLP
8:	s = SnowNLP(text).words
9:	import thulac
10:	thulac = thulac.thulac(user_dict="dict.txt", seg_only=True)
11:	thulac.cut_f("input.txt", "tmp\\thulac_word_seg.txt")

Table S2 Gensim performs Word2Vec modeling and similarity calculation

1:	from Gensim import models, similarities
2:	file_object = open("tmp\\word_seg.txt", encoding='UTF-8')
3:	sentence = file_object.read()
4:	texts = [[word for word in sentence.split()]]
5:	count_dict=[]
6:	for words in texts[0]:
7:	if(len(words) >= 2):
8:	if(words not in count_dict):
9:	count_dict[words] = 0
10:	count_dict[words] += 1
11:	count_rs = sorted(count_dict.items(), key=lambda x:x[1])
12:	save_count =''
13:	size = len(count_rs)
14:	for i in range(size):
15:	save_count += (count_rs[size-i-1][0] + " : " + str(count_rs[size-i-1][1]) + "\n")
16:	save_object = open("result.txt", 'w', encoding='UTF-8')
17:	save_object.write(save_count)
18:	model = models.Word2Vec(texts, size=100, window=5, min_count=5)
19:	similarity = model.similarity("COVID-19", "Epidemic control measure")
20:	print("similarity = %.4f" % similarity)