读取训练数据

并将读取的句子存储到text中

In [1]:
```
import json

text=[]

f_read=open('./data/体育.json', 'r',encoding='utf8',errors='ignore')

for line in f_read:
    line=line.replace('\\u0009','').replace('\\n','')
    obj=json.loads(line)
    sent=obj['contentClean']
    text.append(sent)
```

In [2]:
```
import jieba
processed_text=[]

for sent in text:
    processed_sent=jieba.cut(sent.strip(' '))
    processed_text.append(list(processed_sent))

print(processed_text[0])
```

```
Building prefix dict from the default dictionary ...
Loading model from cache C:\Users\fengl\AppData\Local\Temp\jieba.cache
Loading model cost 0.485 seconds.
Prefix dict has been built successfully.
['远', '在', '土耳其', '打球', '的', '朱婷', '迎来', '自己', '的', '大', '日子', ',', ',
'今年', '11', '月', '29', '日', '是', '这位', '中国女排', '当家', '球星', '的', '22',
'岁', '生日', '。', '尽管', '在', '国外', ',', ',', '但', '朱婷', '还是', '感受', '到',
'了', '家乡', '的', '温暖', ',', ',', '因为', '29', '日', '她', '有', '一场', '特别',
'的', '生日会', ',', ',', '腾讯', '体育', '也', '对', '这场', '生日会', '进行', '了', '全
程', '直播', '。', ',', ',', ',', ',', ',', '郎导', '携', '女儿', '录像', '送祝福', ',', ',
'黄晓明', '成', '意外', '惊喜', ',', ',', '当地', '时间', '13', '点', '30', '分',
',', ',', '朱婷', '的', '生日会', '正式', '开始', '。', '作为', '当天', '的', '绝对', '主
角', ',', ',', '朱婷', '结束', '了', '上午', '的', '训练', '匆匆', '赶来', '。', '她', '身
穿', '运动服', '刚', '进入', '会场', ',', ',', '参加', '生日会', '的', '球迷', '和', '记
者', '就', '颇', '有', '默契', '地', '一起', '为', '朱婷', '高唱', '生日快乐', ',', ',
'现场', '其乐融融', '。', '谦逊', '的', '朱婷', '开口', '第一句', '就是', '感谢',
',', ',', '“', '这是', '第一次', '在', '国外', '过生日', ',', ',', '很', '感动', ',', ',', '大
家', '特意', '从', '国内', '飞过来', ',', ',', '让', '我', '在', '海外', '也', '有家',
'的', '感觉', '。', '以后', '我要', '以', '更好', '的', '训练', '和', '比赛', '来',
'回报', '大家', '。', '”', ',', ',', ',', ',', '朱婷', '来到', '土耳其', '比赛', '之后', ',', ',
'赞助商', '还', '特别', '为', '她', '配备', '了', '一名', '厨师', '随行', ',', ',', '朱
婷', '时不时', '就', '会', '在', '深夜', '“', '放毒', '”', ',', ',', '上传', '各种', '美
食', '照', '。', '朱婷', '辩称', '她', '可不是', '在', '炫耀', ',', ',', '而是', '另有',
'目的', ':', '“', '我', '之所以', '每天', '深夜', '发吃', '的', ',', ',', '是', '希望',
'关心', '我', '的', '朋友', '知道', ',', ',', '我', '在', '土耳其', '生活', '得', '很',
'好', ',', ',', '后盾', '很', '坚强', '。', '”', ',', ',', ',', ',', '虽然', '朱婷', '在', '国内',
'的', '队友', '和', '教练', '不能', '到', '现场', '庆祝', ',', ',', '但', '他们', '还是',
'用', '录像', '视频', '的', '方式', '送上', '了', '祝福', '。', '张', '常宁', '、',
'惠若琪', '、', '龚', '翔宇', '、', '沈静', '思', '、', '单丹娜', '等', '人', '一一',
'出现', '在', '现场', '大屏幕', '上', ',', ',', '朱婷', '恩师', '、', '带领', '中国女排',
'在', '里', '约', '夺冠', '的', '郎平', '教练', '还', '特别', '带', '着', '女儿', '一
起', '为', '爱徒', '录制', '了', '生日歌', ',', ',', '两人', '一边', '唱', '一边', '不',
'忘', '活泼', '搞怪', ',', ',', '看', '得', '原本', '眼眶', '含泪', '的', '朱婷', '破涕为
笑', '。', ',', ',', ',', ',', ',', ',', '郎导', '唱', '完', '生日歌', '后', ',', ',', '视频', '并',
'没有', '就此结束', ',', ',', '生日会', '还给', '了', '朱婷', '一个', '惊喜', ':', '她',
'的', '偶像', '黄晓明', '也', '通过', '大屏幕', '视频', '祝贺', '朱婷', '生日快乐',
```

'，', '，', '黄', '教主', '还', '将', '朱婷', '称呼', '为', '“', '全民', '女神', '”', '。', '看到', '偶像', '出现', '，', '，', '22', '岁', '的', '朱婷', '立马', '成', '了', '小女生', '，', '，', '直接', '爆发', '出', '一声', '尖叫', '：', '“', '啊', '！', '”', '，', '朱婷', '脸上', '显露出', '害羞', '的', '神情', '，', '，', '她', '还', '清楚', '地', '记得', '这是', '黄晓明', '第二次', '为', '她', '录制', '视频', '：', '“', '第一次', '是', '在', '一个', '节目', '里', '，', '，', '（', '看到', '黄晓明', '的', '生日', '祝福', '）', '，', '，', '这次', '生日会', '本来', '已经', '很', '圆满', '，', '，', '但是', '现在', '更', '圆满', '了', '。', '”', '，', '，', '，', '土耳其', '粉丝', '送礼物', '，', '，', '朱婷', '贴心', '为', '他', '煮', '面', '，', '，', '，', '朱婷', '是', '来自', '河南', '的', '姑娘', '，', '，', '所以', '今天', '的', '长寿面', '也', '很', '特别', '，', '，', '厨师', '特别', '准备', '了', '河南', '烩面', '。', '看到', '家乡', '的', '美食', '，', '，', '朱婷', '不', '等', '小料', '上', '齐', '，', '，', '拿', '起', '筷子', '就', '吃', '起来', '。', '嗖嗖', '几口', '下肚', '，', '，', '朱婷', '吃', '得', '一脸', '满足', '。', '“', '生日', '吃', '长寿面', '非常', '有', '意义', '，', '，', '能', '吃', '到', '河南', '烩面', '更', '有', '家乡', '情', '。', '”', '朱婷', '表示', '。', '，', '，', '，', '朱婷', '也', '不是', '光', '吃', '不', '做', '，', '，', '现场', '有', '媒体', '起哄', '让', '朱婷', '给', '大家', '煮面', '，', '，', '没想到', '她', '直爽', '地', '一口', '答应下来', '：', '“', '（', '你们', '）', '愿意', '吃', '吗', '？', '好', '！', '”', '说完', '朱婷', '就', '在', '厨师', '的', '指导', '下', '，', '，', '将', '面条', '丢进', '锅里', '的', '老汤', '煮起来', '。', '要', '知道', '这锅', '老汤', '可不', '一般', '，', '，', '是', '厨师', '用', '中国', '空运', '过来', '的', '羊肉', '，', '，', '加上', '鸡', '、', '鸭', '和', '鲫鱼', '熬制', '了', '一', '晚上', '做成', '的', '。', '，', '，', '，', '，', '那么', '谁', '能', '成为', '吃', '到', '朱婷', '这', '碗面', '的', '幸运儿', '呢', '？', '答案', '是', '土耳其', '的', '一位', '球迷', '。', '作为', '中国女排', '的', '当家', '球星', '，', '，', '朱婷', '也', '吸引', '了', '众多', '国外', '球迷', '的', '关注', '。', '今天', '的', '生日会', '，', '，', '一位', '土耳其', '的', '球迷', '就', '特意', '为', '朱婷', '准备', '了', '礼物', '，', '，', '来到', '现场', '送上', '。', '朱婷', '除了', '和', '他', '合影留念', '，', '，', '还', '将', '自己', '煮', '的', '面送', '上', '，', '，', '贴心', '为', '这位', '球迷', '往面', '里加', '小料', '。', '，', '，', '，', '，', '生日会', '的', '最后', '，', '，', '自然', '是', '切', '蛋糕', '的', '环节', '。', '一个', '大', '蛋糕', '缓缓', '推入', '会场', '，', '，', '准备', '的', '生日', '刀', '居然', '是', '一把', '武士刀', '，', '，', '朱婷', '看到', '先是', '一', '愣', '，', '，', '随后', '还', '俏皮', '地', '拿', '着', '武士刀', '摆起', 'POSE', '。', '最终', '全场', '又', '一次', '齐声高唱', '生日歌', '，', '，', '朱婷切', '下', '蛋糕', '，', '，', '生日会', '画上', '完满', '的', '句号', '。']

word2vec模型的使用

word2vec模型训练

In [ ]:

```
from gensim.models import word2vec
#训练 @hs:
w_model = word2vec.Word2Vec(processed_text,hs=1,sg=0,min_count=1,window=3,size=100)
#min_count是最低出现数，默认数值是5;
#size是gensim Word2Vec将词汇映射到的N维空间的维度数量（N）默认的size数是100;
#iter是模型训练时在整个训练语料库上的迭代次数，假如参与训练的文本量较少，就需要把这个参
#hs：word2vec两个解法的选择。如果是0， 则是Negative Sampling；如果是1，则是Hierarchica
#sg是模型训练所采用的的算法类型：1 代表 skip-gram，0代表 CBOW，sg的默认值为0;
#window控制窗口，如果设得较小，那么模型学习到的是词汇间的组合性关系（词性相异）；如果设



w_model.save('w_model')  # 保存模型
```

读取训练好的模型

In [10]:

```
from gensim.models import word2vec

w_model = word2vec.Word2Vec.load('w_model')  # 加载模型

#包含哪些词？
```

```
#print(w_model.wv.vocab)


vocab=list(w_model.wv.vocab.keys())
print(vocab[:100])
#获得任意词的词向量

vec_example = w_model.wv['日子']
print(vec_example)

#找到与给定词语义最为相似的前10个词
sims = w_model.wv.most_similar('自己', topn=10)

print(sims)

#计算词之间的相似度
print(w_model.wv.similarity('他们','自己'))
```

```
['远', '在', '土耳其', '打球', '的', '朱婷', '迎来', '自己', '大', '日子', ',', ',', '今
年', '11', '月', '29', '日', '是', '这位', '中国女排', '当家', '球星', '22', '岁', '生
日', '。', '尽管', '国外', '但', '还是', '感受', '到', '了', '家乡', '温暖', '因为',
'她', '有', '一场', '特别', '生日会', '腾讯', '体育', '也', '对', '这场', '进行', '全
程', '直播', ' ', '郎导', '携', '女儿', '录像', '送祝福', '黄晓明', '成', '意外', '惊
喜', '当地', '时间', '13', '点', '30', '分', '正式', '开始', '作为', '当天', '绝对',
'主角', '结束', '上午', '训练', '匆匆', '赶来', '身穿', '运动服', '刚', '进入', '会
场', '参加', '球迷', '和', '记者', '就', '颇', '默契', '地', '一起', '为', '高唱', '生
日快乐', '现场', '其乐融融', '谦逊', '开口', '第一句', '就是', '感谢', '“']
[-0.15171613 -0.06484345 -0.1215067   0.0956796   0.13846588  0.02256004
 -0.00509479  0.03878618 -0.21499917  0.15616797 -0.03201747 -0.10941551
 -0.21069099  0.06732899 -0.16809037  0.10370489  0.21571535 -0.15364592
 -0.17161456  0.02397017  0.01246019 -0.00075803 -0.06074083 -0.17089768
  0.16282842  0.1259981   0.0755522   0.01291898  0.14630677 -0.02005492
 -0.11673237 -0.04089251 -0.14170754  0.09147081 -0.02211859 -0.10169024
 -0.08533196  0.13624804 -0.16082017 -0.15923166  0.0438658   0.1483839
  0.07566781  0.20700814  0.11773256  0.08824778 -0.11040846 -0.09966952
  0.07453604 -0.15180115 -0.02228062 -0.14346905  0.14802834  0.09109161
  0.06908747  0.15059818  0.19758904 -0.2485349   0.16166702 -0.15893103
 -0.10208724  0.0101655  -0.07975085  0.05255351 -0.15239155 -0.039686
  0.22876024  0.00090867  0.06044475  0.15433851  0.05482744  0.15471481
 -0.10483679  0.05176261 -0.15132916 -0.023852   -0.0151371   0.03713646
 -0.03332234  0.16032118 -0.00796922 -0.2056723  -0.10068612 -0.00680892
  0.2672329   0.0296125  -0.19710153 -0.24151109  0.14180128 -0.07510007
  0.15489179  0.04017694  0.00808898  0.13279407 -0.20592965  0.04585099
 -0.11290726  0.08352386 -0.09018603  0.18962161]
[('我们', 0.9395385980606079), ('他们', 0.9275588989257812), ('她', 0.926349759101867
7), ('我', 0.9192343950271606), ('他', 0.9180630445480347), ('要', 0.916494786739349
4), ('可以', 0.9161877632141113), ('所以', 0.8964742422103882), ('它', 0.8956633210182
19), ('大家', 0.8934755325317383)]
0.9275589
```

doc2vec模型的使用

doc2vec模型的训练

In [3]:

```
import gensim
from gensim.models.doc2vec import Doc2Vec,LabeledSentence

# 生成固定格式的训练文档集合

train_text=[]

for i,sent in enumerate(processed_text):
    #改变成Doc2vec所需要的输入样本格式,
```

```
    #由于gensim里Doc2vec模型需要的输入为固定格式，输入样本为：[句子，句子序号],这里需要
    tagged_doc=gensim.models.doc2vec.TaggedDocument(sent,tags=[i])
    train_text.append(tagged_doc)
    #print(tagged_doc)

d_model=Doc2Vec(train_text,min_count=5,windows=3,vector_size=100,sample=0.001,nagetiv



d_model.train(train_text,total_examples=d_model.corpus_count,epochs=10)
# 保存模型，以便重用
d_model.save("doc2vec_model")  #保存模型
```

```
D:\Anaconda3\envs\nlp_env\lib\site-packages\gensim\models\base_any2vec.py:743: UserWar
ning: C extension not loaded, training will be slow. Install a C compiler and reinstal
l gensim for fast training.
  "C extension not loaded, training will be slow. "
```

In [5]:

```
train_text[1]
```

Out[5]:

```
TaggedDocument(words=['2016', '赛季', '才', '刚刚', '落幕', ',', ',', '但', '相信', '各
位', '已经', '迫不及待', '的', '盼', '着', '2017', '新', '赛季', '速速', '拉开', '大
幕', '了', ',', ',', '尤其', '是', '作为', '中国', '球员', '们', '最', '喜爱', '的', '大满
贯', '赛', ',', ',', '墨尔本', '公园', '会', '非常', '开心', '的', '准备', '好', '迎接',
'她们', '每', '一个', '人', '的', '再次', '做客', '。', '那么', '以下', '就', '来',
'看看', '中国', '军团', '的', '姑娘', '们', '是', '如何', '制定', '赛程', ',', ',', '为',
'澳网', '备战', '的', '吧', '。', '其中', '尤以', '张帅', '的', '赛程', '安排', '最
受', '关注', ',', ',', '连续', '参加', '布里斯班', '和', '悉尼', '这', '两站', '阵容强大',
'的', '顶级', '赛', ',', ',', '可见', '当前', '信心', '正足', '的', '她', '在', '新年',
'的', '首要', '目标', '直指', '排名', '跻身', 'TOP20', '。', ',', ',', ',', ',', ',', ',', '首
周', '(', '1', '月', '1', '日', '-', '7', '日', ')', '深圳', '站', ':', '王蔷', '、',
'彭帅', '、', '郑赛赛', '直接', '入围', ',', ',', '三位', '世界', '前十', ',', ',', '包括',
'卫冕冠军', 'A', '·', '拉德', '万斯', '卡', '、', '2015', '年', '冠军', '哈勒', '普',
'以及', '在', '今年', '闯入', '澳网', '四强', '的', '孔塔', '领衔', '出战', '。', '此
外', ',', ',', '排名', '靠', '前', '的', '段', '莹莹', '、', '韩馨蕴', '、', '王雅繁',
'三', '人均', '有望', '争取', '到', '一张', '正赛', '外卡', '。', '布里斯班', '站',
':', '张帅', '直接', '入围', ',', ',', '科', '贝尔', '、', '齐布尔', '科娃', '、', '卡
洛', '琳娜', '·', '普利', '斯科娃', '以及', '穆古', '拉扎', '占据', '赛会', '前四',
'种子', '位置', ',', ',', '直接', '入围', '的', '22', '人', '截止', '至', '排名第', '45',
'位', '的', '埃', '拉尼', '。', '奥克兰', '站', ':', '无', '中国', '球员', '入围',
'正赛', '。', '四位', '前', '世界', '第一', '威廉姆斯', '姐妹', '、', '沃兹', '尼亚',
'奇', '以及', '伊万诺维奇', '报名', '参赛', '。', '值得一提的是', ',', ',', '这', '是',
'自', '遥远', '的', '1998', '年', '以来', ',', ',', '大小', '威', '首次', '携手', '参加',
'在', '巡回赛', '中', '级别', '最低', '的', '国际', '赛', '(', '此前', '赛制', '被',
'列为', '四级', '赛', ')', '。', ',', ',', ',', '第二周', '(', '1', '月', '8', '日', '-',
'14', '日', ')', '悉尼', '站', ':', '张帅', '直接', '入围', ',', ',', '前四', '种子', '为
科', '贝尔', '、', 'A', '·', '拉德', '万斯', '卡', '、', '齐布尔', '科娃', '、', '卡
洛', '琳娜', '·', '普利', '斯科娃', ',', ',', '报名', '的', '22', '人', '截止', '至', '排
名第', '29', '位', '的', '贝古', ',', ',', '竞争', '环境', '十分激烈', '。', '霍巴特',
'站', ':', '无', '中国', '球员', '入围', '正赛', ',', ',', '报名', '截止', '第', '65',
'位', '。', '预计', '王蔷', '、', '彭帅', '、', '郑赛赛', '、', '段', '莹莹', '等',
'排名', '较', '高', '的', '中国', '选手', '都', '会', '选择', '报名', '本站', '资格
赛', '。', '此外', ',', ',', '2017', '年', '澳大利亚', '网球', '公开赛', '的', '男女', '单
正赛', '入围', '名单', '将', '于', '下周', '揭晓', ',', ',', '报名', '截止', '于', '下周
一', ',', ',', '即', '12', '月', '5', '日', '这', '一期', '的', '世界', '排名榜', '。',
'正常', '来看', ',', ',', '张帅', '、', '王蔷', '、', '彭帅', '以及', '郑赛', '赛四人',
'入围', '无忧', ',', ',', '排在', '第', '103', '位', '的', '段', '莹莹', '问题', '也', '不
大', ',', ',', '而', '考虑', '到', '今年', '的', '入围', '门槛', '是', '第', '108', '位',
',', ',', '所以', '排在', '第', '111', '位', '的', '去年', '外卡', '赛', '女单', '冠军',
'得主', '韩馨蕴', '恐怕', '这', '一次', '将', '只能', '从', '资格赛', '打', '起',
'。'], tags=[1])
```

doc2vec模型的使用

In [4]:

```
#load doc2vec model...
d_model= gensim.models.doc2vec.Doc2Vec.load("doc2vec_model")
#load train vectors...
text_vecs= d_model.docvecs.vectors_docs
print("专利向量的个数为",len(text_vecs))

#print(text_vecs[0])
v1 = d_model.infer_vector(['我们','是','中国','人','篮球'])
v2 = d_model.infer_vector(['我们','打','篮球'])
print(v1)
print(v2)
```

```
专利向量的个数为 500
[ 0.14764266 -0.06431124  0.03621458  0.0159138   0.03448177  0.0232377
 -0.03285934 -0.01283403  0.06542674  0.03245635  0.03963161  0.02854619
  0.00694009 -0.00532997 -0.01024401 -0.0277176  -0.02449099  0.01868313
 -0.01566983 -0.036056    0.02441941  0.06526142  0.05485668 -0.01322194
  0.01448384 -0.01623162  0.10088346 -0.03357361  0.03316192 -0.05252314
 -0.06916207  0.00803608 -0.02189584  0.00219007 -0.03100408  0.0283435
  0.09257389  0.04835166  0.05022921 -0.01101312 -0.02788181 -0.04904336
 -0.00241582 -0.02891141 -0.07422822  0.02533848 -0.05275474 -0.02649931
  0.01843766 -0.04901703  0.03457853 -0.0366088   0.03669108  0.02066261
  0.02080013  0.00281822 -0.01253506  0.03561695 -0.00192294 -0.00815695
 -0.05161023 -0.02858902  0.06002551 -0.00574364 -0.0454098   0.00255477
 -0.05435323 -0.02342173 -0.11281837  0.00786249 -0.03335616  0.0047664
  0.00560537  0.03777466  0.07558563  0.02697422 -0.04547644  0.01532341
 -0.08717516 -0.04159654 -0.05362939  0.00861659  0.01259687 -0.01708089
  0.01446363  0.03280181  0.00949849  0.0322456   0.00494036 -0.1172976
  0.00188966  0.04610217  0.02556501 -0.03106703  0.00092462  0.00054077
 -0.00096232 -0.10088306 -0.07057656 -0.00815279]
[ 0.09304454 -0.03735169 -0.00175544  0.01777876  0.01303388 -0.00575239
 -0.00204638 -0.03450556  0.01652898  0.00200058 -0.01703838  0.03442948
  0.00191302  0.00530618 -0.0122488  -0.0134324  -0.00997465 -0.01238151
 -0.01223813  0.00500524 -0.01356631  0.02988765  0.0569452  -0.00756418
 -0.01391707  0.00694827  0.02185297 -0.01069848 -0.01892307 -0.04081366
 -0.02460967 -0.02704859  0.0083892   0.01854569 -0.02227442  0.00434633
  0.04330476  0.04138111  0.03255557  0.01383692 -0.00754562 -0.02667187
 -0.03743677 -0.00759231 -0.05890828  0.04771578 -0.02355438 -0.00401311
  0.02845985 -0.04959408  0.03589477  0.00820648  0.02111495  0.01007846
  0.0155894  -0.01692143  0.00564508  0.06546513 -0.01081425  0.00659607
 -0.05124555 -0.02181909  0.03100877 -0.00429172 -0.01795107 -0.03612999
 -0.04840308 -0.00899944 -0.09020904  0.02017352 -0.0107391   0.00718688
  0.01788833 -0.03286805  0.02620029  0.01177383 -0.01584521 -0.00904585
 -0.03408313 -0.06199186 -0.02664232  0.007005   -0.0107647  -0.03746688
  0.01585818  0.0426845   0.01566764  0.02316699  0.01837096 -0.06658714
  0.00920006  0.03851668  0.0275213   0.0128712   0.01616348  0.00969492
  0.02473134 -0.05547648 -0.03232773 -0.03110693]
```

文本相似度计算

相似度计算公式

In [6]:

```
import numpy as np

#余弦相似度
def cosine(p,q):

    # 如果特征长度不同，不计算相似度
    if (len(p) != len(q)):
        raise Exception("feature length must be the same")

    d = np.dot(p, q) / (np.linalg.norm(p) * np.linalg.norm(q))
    return d
```

```
#计算欧几里德距离：
def euclidean(p,q):
    '''
    :param p: list
    :param q: list
    :return:
    '''
    #如果特征长度不同，不计算相似度
    if(len(p) != len(q)):
        raise Exception("feature length must be the same")

    p,q = np.array(p),np.array(q)
    d1 = np.sqrt(np.sum(np.square(p - q)))

    return d1
```

In [7]:

```
sent1=['我们','是','中国','人','篮球']
sent2=['我们','打','篮球']
```

基于doc2vec的文本相似度计算

In [8]:

```
v1 = d_model.infer_vector(sent1)
v2 = d_model.infer_vector(sent2)

print(cosine(v1,v2))
```

0.8190077

基于word2vec的文本相似度计算

In [11]:

```
#得到句子的向量
v1=np.zeros(100)

for token in sent1:
    if token in vocab:
        temp_vec=w_model.wv[token]
        v1=v1+temp_vec
print(v1)

v2=np.zeros(100)

for token in sent2:
    if token in vocab:
        temp_vec=w_model.wv[token]
        v2=v2+temp_vec
print(v2)

# 计算文档相似度

print(cosine(v1,v2))
```

```
[ 0.79935229 -2.8112359  -2.67529592  1.77776997  0.63430199 -0.12110377
  1.34489636  4.07388118 -1.0475929   3.03138485  0.22680779 -0.13090718
 -5.49571759  0.49704505 -1.52391693  1.90278786  1.86197911 -2.64685642
 -3.19367594  0.81205047 -0.28684758  0.5806002  -3.0269957  -4.52041344
  1.10572646  0.12863908  1.25636038  1.57427025  3.22795716 -2.84186592
  0.13919246 -1.04198663  0.05321465  0.45585977 -1.58370693 -0.58145253
  1.01481976  1.49189974 -4.94487463 -2.316903    0.7044914   5.16641307
```

```
  2.02843453   2.38270494   2.12579539 -0.07294151 -1.8555599  -3.55242227
  2.40118575 -3.41706878   0.51931543 -0.57317497   1.55800923 -0.8632395
 -0.55305066   2.90742472   1.72120668 -3.72486615   5.26300067 -0.99686486
 -1.321136     0.58403195 -2.8815799    0.53211874 -3.04243644   0.27373886
  2.62044951 -1.66037121 -0.33685057   5.26045209 -0.21810159   1.92556137
 -4.72116682   1.70094695 -2.00396816   3.38614228 -2.76414939   1.04732002
  0.61012713   3.32019401   1.13252862   1.29509145   1.72295034   0.23659665
  4.09761673   1.37757736 -2.44423392 -2.45158276   2.71991877   0.82890783
  4.74615419 -0.38087733 -0.14525716   2.47699936 -3.09012868   0.25160309
 -3.52699596   1.50458076   0.7305052    1.89207451]
[-2.12521516e-02 -7.83272734e-01 -1.22555894e+00  7.16690391e-01
  9.79586914e-01 -8.87581035e-01  2.81191248e-01  1.44059059e+00
  3.77991736e-01  1.69052561e+00  1.12568790e+00 -8.89009854e-01
 -3.15780538e+00 -1.30746374e-01 -1.27646068e+00  9.71540660e-01
  8.47519174e-01 -1.35621747e+00 -2.38400948e+00  5.20838067e-01
 -6.36118248e-01  3.93431440e-01 -9.82768625e-01 -1.86492687e+00
  6.00586511e-01  5.28518083e-01  2.42248805e-01  1.25438705e+00
  1.99669552e+00 -1.32787015e+00 -5.55584252e-01 -4.43426982e-01
 -9.00207333e-01 -3.50526169e-01 -1.39934735e+00 -5.94904929e-01
  1.91018522e-01  1.72577263e-01 -2.88040760e+00 -1.84913976e+00
  8.21501017e-04  2.86590987e+00  1.29819117e+00  1.84775314e+00
  1.07753652e+00  8.40870649e-01 -1.11712216e+00 -2.52658200e+00
  5.67702720e-01 -1.26339366e+00 -8.24223086e-03 -7.46489835e-01
  6.50034208e-01  2.24902987e-01 -4.73273769e-01  2.24515688e+00
  1.12106232e+00 -1.83959520e+00  2.95569909e+00 -7.54465465e-01
 -1.34974141e+00  1.13248593e+00 -1.36396190e+00 -2.09418342e-01
 -2.75203228e+00  1.52348820e-01  2.39750963e+00 -2.19226152e-01
 -8.20780706e-01  2.86666304e+00  3.28795679e-01  1.16284728e+00
 -1.82247929e+00  2.89548551e-01 -1.35069266e+00  1.30598327e+00
 -7.31808871e-01  5.12695171e-01 -1.71776347e-01  1.62481916e+00
  5.48301145e-01  3.25975925e-01  6.62753731e-02  1.06904447e-01
  2.58547068e+00  7.72553816e-01 -1.00078242e+00 -1.61045823e+00
  1.76580311e+00  2.47926965e-01  2.83740246e+00 -6.96200110e-01
  2.59115845e-02  5.77039324e-01 -2.21691245e+00  7.74447786e-01
 -1.83529833e+00  6.32270493e-01 -1.07084260e+00  1.49380100e+00]
0.9233155228352673
```