File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

In [7]:
```
# %load_ext memory_profiler
!pip install -q zhconv
```

In [10]:
```
!pip install gensim
```

```
Collecting gensim
  Downloading gensim-4.3.1-cp38-cp38-win_amd64.whl (24.0 MB)
     ---------------------------------------- 24.0/24.0 MB 20.5 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.18.5 in c:\users\larry\appdata\local\programs\python\python38\lib\site-packages (from gens
im) (1.23.4)
Requirement already satisfied: scipy>=1.7.0 in c:\users\larry\appdata\local\programs\python\python38\lib\site-packages (from gensi
m) (1.9.1)
Collecting smart-open>=1.8.1 (from gensim)
  Downloading smart_open-6.3.0-py3-none-any.whl (56 kB)
     ---------------------------------------- 56.8/56.8 kB ? eta 0:00:00
Installing collected packages: smart-open, gensim
Successfully installed gensim-4.3.1 smart-open-6.3.0
```

In [15]:
```
!pip install wget
```

```
Collecting wget
  Downloading wget-3.2.zip (10 kB)
  Preparing metadata (setup.py): started
  Preparing metadata (setup.py): finished with status 'done'
Building wheels for collected packages: wget
  Building wheel for wget (setup.py): started
  Building wheel for wget (setup.py): finished with status 'done'
  Created wheel for wget: filename=wget-3.2-py3-none-any.whl size=9682 sha256=482b6c754c5ad0eb525f928fbe4f2841856a2d8e13537464857e
e3e0ff852a68
  Stored in directory: c:\users\larry\appdata\local\pip\cache\wheels\bd\a8\c3\3cf2c14a1837a4e04bd98631724e81f33f462d86a1d895fae0
Successfully built wget
Installing collected packages: wget
Successfully installed wget-3.2
```

In [2]:
```
import urllib.request
url = "https://github.com/fxsjy/jieba/raw/master/extra_dict/dict.txt.big"
filename = "dict.txt.big"
urllib.request.urlretrieve(url, filename)
```

```
('dict.txt.big', <http.client.HTTPMessage at 0x2233b3aab50>)
```

In [3]:
```
import os

# Packages
import gensim
import jieba
import zhconv
from gensim.corpora import WikiCorpus
from datetime import datetime as dt
from typing import List


if not os.path.isfile('dict.txt.big'):
    !wget https://github.com/fxsjy/jieba/raw/master/extra_dict/dict.txt.big
jieba.set_dictionary('dict.txt.big')

print("gensim", gensim.__version__)
print("jieba", jieba.__version__)
```

```
gensim 4.3.1
jieba 0.42.1
```

In [6]:
```
ZhWiki = r"C:\Users\Larry\Downloads\zhwiki-20230501-pages-articles-multistream.xml.bz2"
```

In [7]:
```
zhconv.convert("这原本是一段简体中文", "zh-tw")
```

```
'這原本是一段簡體中文'
```

In [9]:
```
print(list(jieba.cut("中英夾雜的example，Word2Vec應該很interesting吧?")))
```

```
['中', '英', '夾雜', '的', 'example', '，', 'Word2Vec', '應該', '很', 'interesting', '吧', '?']
```

In [1]:
```
# !pip install spacy
```

In [4]:
```
import spacy

# 下載語言模組
spacy.cli.download("zh_core_web_sm")  # 下載 spacy 中文模組
spacy.cli.download("en_core_web_sm")  # 下載 spacy 英文模組

nlp_zh = spacy.load("zh_core_web_sm") # 載入 spacy 中文模組
nlp_en = spacy.load("en_core_web_sm") # 載入 spacy 英文模組

# 印出前20個停用詞
print('--\n')
print(f"中文停用詞 Total={len(nlp_zh.Defaults.stop_words)}: {list(nlp_zh.Defaults.stop_words)[:20]} ...")
print("--")
print(f"英文停用詞 Total={len(nlp_en.Defaults.stop_words)}: {list(nlp_en.Defaults.stop_words)[:20]} ...")
```

```
✓ Download and installation successful
You can now load the package via spacy.load('zh_core_web_sm')
✓ Download and installation successful
You can now load the package via spacy.load('en_core_web_sm')
--

中文停用詞 Total=1891: ['移动', '起头', '难得', '’∈', '所谓', '略微', '成为', '”', '後来', '反之', '↑', '造成', '毋宁', '靠', '顷刻',
'或者', '失去', '//', '不巧', '加强'] ...
--
英文停用詞 Total=326: ['’ll', 'is', 'yours', 'sixty', 'then', 'nine', 'whenever', 'it', 'former', 'thru', 'from', 'via', 'still',
'themselves', 'hereafter', '’re', 'thus', 'put', 'becoming', 'until'] ...
```

In [6]:
```
STOPWORDS =  nlp_zh.Defaults.stop_words | \
             nlp_en.Defaults.stop_words | \
             set(["\n", "\r\n", "\t", " ", ""])
print(len(STOPWORDS))

# 將簡體停用詞轉成繁體，擴充停用詞表
for word in STOPWORDS.copy():
    STOPWORDS.add(zhconv.convert(word, "zh-tw"))

print(len(STOPWORDS))
```

```
2222
3005
```

In [12]:
```
def preprocess_and_tokenize(
    text: str, token_min_len: int=1, token_max_len: int=15, lower: bool=True) -> List[str]:
    if lower:
        text  = text.lower()
    text = zhconv.convert(text, "zh-tw")
    return [
        token for token in jieba.cut(text, cut_all=False)
        if token_min_len <= len(token) <= token_max_len and \
            token not in STOPWORDS
    ]
```

In [13]:
```
print(preprocess_and_tokenize("歐幾里得，西元前三世紀的古希臘數學家，現在被認為是幾何之父，此畫為拉斐爾"))
print(preprocess_and_tokenize("我来到北京清华大学"))
print(preprocess_and_tokenize("中英夾雜的example，Word2Vec應該很interesting吧?"))
```

```
['歐幾', '裡得', '西元前', '世紀', '古希臘', '數學家', '幾何', '父', '此畫', '拉斐爾']
['來到', '北京', '清華大學']
['中', '英', '夾雜', 'example', 'word2vec', 'interesting']
```

In [ ]:
```
# Do this cell in colab
print(f"Parsing {ZhWiki}...")
wiki_corpus = WikiCorpus(ZhWiki, tokenizer_func=preprocess_and_tokenize, token_min_len=1)
print("finish")
```

```
Parsing C:\Users\Larry\Downloads\zhwiki-20230501-pages-articles-multistream.xml.bz2...


c:\users\larry\appdata\local\programs\python\python38\lib\site-packages\gensim\utils.py:1333: UserWarning: detected Windows; alias
ing chunkize to chunkize_serial
  warnings.warn("detected %s; aliasing chunkize to chunkize_serial" % entity)
```

In [18]:
```
g = wiki_corpus.get_texts()
print(next(g)[:10])
print(next(g)[:10])
print(next(g)[:10])



# print(jieba.lcut("".join(next(g))[:50]))
# print(jieba.lcut("".join(next(g))[:50]))
```

```
['歐幾', '裡得', '西元前', '世紀', '古希臘', '數學家', '幾何', '父', '此畫', '拉斐爾']
['蘇格拉底', '之死', '雅克', '路易', '大衛', '所繪', '1787', '年', '哲學', '研究']
['文學', '狹義', '一種', '語言藝術', '語言文字', '手段', '形象化', '客觀', '社會', '生活']
```

In [7]:
```
WIKI_SEG_TXT = r"C:\Users\Larry\Desktop\wiki_seg.txt"
```

In [10]:
```
%%time

from gensim.models import word2vec
import multiprocessing

max_cpu_counts = multiprocessing.cpu_count()
word_dim_size = 300  #  設定 word vector 維度
print(f"Use {max_cpu_counts} workers to train Word2Vec (dim={word_dim_size})")


# 讀取訓練語句
sentences = word2vec.LineSentence(WIKI_SEG_TXT)

# 訓練模型
model = word2vec.Word2Vec(sentences, vector_size=word_dim_size, workers=max_cpu_counts)

# 儲存模型
output_model = f"word2vec.zh.{word_dim_size}.model"
model.save(output_model)
```

```
Use 12 workers to train Word2Vec (dim=300)
CPU times: total: 1h 14min 42s
Wall time: 34min 27s
```

In [12]:
```
print(model.wv.vectors.shape)
model.wv.vectors
```

```
(1281108, 300)
```

```
array([[ 2.3356509e+00,  9.9428272e-01, -2.5183547e+00, ...,
         8.9413118e-01,  3.9101195e-01, -3.4498594e+00],
       [ 2.5912645e+00,  1.1913005e+00, -2.6872811e+00, ...,
         4.6329018e-01,  1.4804647e+00, -4.2031350e+00],
       [ 6.5827179e-01,  9.5819396e-01, -9.1632044e-01, ...,
         5.2813661e-01,  1.0274427e+00, -1.5687137e+00],
       ...,
       [-1.9454038e-02, -1.9204972e-02,  8.3302788e-02, ...,
        -8.2949176e-03, -5.7278134e-02, -8.1093840e-02],
       [-4.5084157e-03, -1.4150353e-02, -3.1951465e-02, ...,
         8.8870479e-04,  2.4251521e-02, -2.6161406e-02],
       [-3.9887622e-02,  5.4721795e-02,  2.4229368e-02, ...,
        -1.7574297e-02, -2.9595951e-03,  3.7285085e-03]], dtype=float32)
```

In [16]:
```
print(f"總共收錄了 {len(model.wv.key_to_index)} 個詞彙")

print("印出 20 個收錄詞彙:")
print(list(model.wv.key_to_index.keys())[:10])
```

```
總共收錄了 1281108 個詞彙
印出 20 個收錄詞彙:
['年', '月', '日', '中', '10', '12', '11', '小行星', '中國', '時']
```

In [17]:
```
vec = model.wv['數學家']
print(vec.shape)
vec
```

```
(300,)
```

```
array([-2.5098159e+00, -8.7425709e-01, -1.1193275e+00, -7.6306146e-01,
        1.5183293e+00, -1.3143593e+00, -3.0370018e+00, -3.9760044e-01,
       -2.3294845e+00, -6.9685298e-01,  6.2689441e-01,  1.4316865e+00,
        3.9469555e-01, -2.8910109e-01, -7.3121977e-01,  4.1757700e-01,
       -2.5718018e-01,  1.3801970e+00, -1.9639119e+00,  7.4829765e-02,
       -1.9182280e+00, -2.0201023e+00,  6.2039506e-01, -1.4086714e+00,
       -2.0132906e+00,  9.5298165e-01,  1.0058908e+00,  1.8032556e+00,
       -1.3946528e+00, -7.1193588e-01, -3.3011141e-01, -1.2556978e-02,
       -1.1751009e+00,  1.7472136e-01, -1.8263913e+00,  2.0147755e+00,
        2.5440696e-01,  1.1868458e+00, -7.6499414e-01, -1.2859404e+00,
       -7.7888030e-01, -4.2789621e+00,  1.1276323e-01,  5.8384013e-01,
       -1.2326813e+00, -9.0584141e-01,  2.7983325e+00, -6.9619542e-01,
       -5.3100312e-01,  2.1726296e+00, -2.2379658e+00, -4.3764052e-01,
       -1.1119894e+00,  1.8412908e+00, -1.3443875e-01,  7.2078747e-01,
       -2.5201950e+00, -3.7151155e-01, -7.5885795e-02,  1.7191498e+00,
        5.8762676e-01,  1.7287215e+00, -3.1106675e-01, -3.0740128e+00,
       -7.7469391e-01,  1.0981706e+00, -7.7425843e-01, -8.0467746e-02,
        2.2519228e-01,  6.3732147e-01,  8.8380104e-01, -1.7376436e+00,
        2.9409137e-01,  8.4651172e-01, -1.5542008e-01,  1.2512927e+00,
       -1.4120048e+00,  6.2775415e-01, -9.4792390e-01, -4.3783218e-01,
       -8.8143331e-01, -3.2813647e-01,  5.4406661e-01, -3.4755824e+00,
       -5.2497974e-03, -2.0792098e+00,  6.5021329e-02, -1.3142005e-01,
       -5.0192219e-01,  5.1159233e-02,  4.3971735e-01, -5.5051732e-01,
       -1.4449131e+00,  2.2459297e+00,  1.9348378e+00,  1.6823611e+00,
        1.4129283e+00, -3.4565011e-01, -1.2944497e+00, -1.9387510e+00,
        1.5396036e+00,  1.0389451e+00,  9.4906360e-01,  1.6002766e+00,
       -5.8514214e-01, -2.7621956e+00, -2.1412592e+00,  2.1607687e+00,
       -2.9859190e+00,  1.8197809e+00,  5.7878196e-01, -4.0863398e-01,
       -2.6183994e+00,  2.1034479e-01, -2.0026236e+00, -1.4798939e+00,
       -9.7133791e-01, -9.9247789e-01, -6.8561591e-02, -1.0578014e+00,
       -8.0379170e-01,  1.0906804e+00, -1.3252856e+00, -7.0287091e-01,
       -2.0765455e+00,  1.0654986e+00, -9.3247640e-01,  3.2667754e+00,
        2.7340227e-01, -1.8300693e+00, -8.6576372e-01,  7.7677542e-01,
       -2.9622028e+00,  1.7873685e+00,  1.0965225e+00,  2.1000713e-01,
        2.1276717e+00, -2.9315794e+00,  1.9664236e+00,  5.0422454e-01,
```

```
-2.6183994e+00,  2.1034479e-01, -2.0026236e+00, -1.4798939e+00,
-9.7133791e-01, -9.9247789e-01, -6.8561591e-02, -1.0578014e+00,
-8.0379170e-01,  1.0906804e+00, -1.3252856e+00, -7.0287091e-01,
-2.0765455e+00,  1.0654986e+00, -9.3247640e-01,  3.2667754e+00,
 2.7340227e-01, -1.8300693e+00, -8.6576372e-01,  7.7677542e-01,
-2.9622028e+00,  1.7873685e+00,  1.0965225e+00,  2.1000713e-01,
 2.1276717e+00, -2.9315794e+00,  1.9664236e+00,  5.0422454e-01,
 1.4087222e+00,  5.9601289e-01, -8.5008425e-01, -4.0505915e+00,
 1.1835086e+00,  2.3517089e+00,  1.7559403e+00, -9.1961044e-01,
-1.6161160e+00, -8.8870454e-01, -1.0705582e-02, -7.8470922e-01,
 2.4660101e+00, -1.6089170e+00,  2.5871605e-01,  3.5639629e+00,
 2.5800874e+00,  1.5180526e+00,  2.2738509e+00, -1.7974727e-01,
-9.0605462e-01,  4.3590224e-01, -1.0028239e+00,  9.2346197e-01,
-1.5424204e+00,  1.6828945e-01,  1.5248320e+00, -8.5174119e-01,
-1.0231490e+00, -2.1322467e+00,  6.6780001e-02, -3.8399644e+00,
-5.8246683e-02, -1.5927006e+00, -1.3300869e+00,  5.8082420e-01,
-1.6367599e+00,  1.8619974e+00, -1.1271410e+00, -2.2339559e+00,
-1.9667931e-01, -1.3902884e+00,  2.3887698e-01, -1.9740051e+00,
 1.8914998e+00, -1.5342224e-01, -2.9233914e-02, -1.6305566e+00,
-6.8159181e-01, -1.2855948e+00, -3.3888325e-01,  2.8311336e-01,
 1.6285813e+00,  1.3949851e+00, -6.3588709e-01, -2.0444918e+00,
-2.2761972e+00, -6.0189545e-01, -9.1646022e-01, -1.3149178e+00,
-1.6722966e+00, -2.5400002e+00,  4.4068119e-01, -3.8710552e-01,
 1.3766977e-01, -7.7309704e-01,  2.7076867e+00, -9.6224910e-01,
 2.0799849e+00,  1.8228353e-03,  3.2291839e-01,  3.0823514e+00,
-2.3910648e-01, -6.7840630e-01,  1.0284235e+00,  2.1988162e-01,
 2.0716677e+00, -7.4077868e-01, -1.4308283e+00,  2.2135131e+00,
-2.3703349e+00,  4.7084403e-01, -6.0994977e-01, -1.3539031e+00,
-8.8549018e-01,  1.1478771e-01,  2.3990755e+00, -1.3474363e+00,
 1.9536786e-01, -2.1184410e-01,  8.1344682e-01,  3.4507101e+00,
-2.5640447e+00,  8.2078540e-01,  2.0369515e+00,  3.8065135e-02,
-1.4829081e+00, -5.2081418e-01,  1.5935873e+00,  1.8186841e+00,
 1.0150800e+00,  6.1693972e-01, -4.9038970e-01, -3.0585778e+00,
 1.8881921e+00,  5.7682824e-01, -1.6601613e+00, -9.0596420e-01,
-8.1696558e-01,  9.4008255e-01, -8.3768928e-01,  1.3375506e+00,
 8.8708532e-01,  7.3253608e-01, -9.2730439e-01, -1.2124244e+00,
-9.5621502e-01, -3.2157729e+00, -5.6441879e-01,  9.8379523e-01,
 1.6980993e+00,  6.5199029e-01,  6.5927112e-01, -9.4362721e-04,
 3.6917147e-01, -6.2483150e-01,  4.3188116e-01, -6.9418067e-01,
-6.6677535e-01,  2.5810044e+00,  2.3590524e+00, -4.8741019e-01,
-1.8116941e+00,  1.5884839e+00,  1.1558408e+00,  1.5452156e+00,
-1.5963442e+00, -8.0355096e-01, -2.1154535e+00,  1.0327097e+00,
 1.0177184e+00, -1.8919615e+00, -2.4699655e-01,  2.9586974e-01,
 6.8665487e-01,  4.1545439e+00, -2.3577856e-01, -1.2149159e+00,
 1.1439668e+00,  2.9635243e+00, -7.8328234e-01, -1.0618507e+00,
 1.2607117e+00,  1.3246002e+00, -9.1961759e-01,  4.2221332e-01,
-1.9433224e-01,  3.9729831e-01,  1.9150944e-01, -2.0348969e+00],
dtype=float32)
```

In [18]:
```
word = "這肉定沒見過 "

# 若強行取值會報錯
try:
    vec = model.wv[word]
except KeyError as e:
    print(e)
```

```
"Key '這肉定沒見過 ' not present"
```

In [19]:
```
model.wv.most_similar("飲料", topn=10)
```

```
[('飲品', 0.809328556060791),
 ('軟飲料', 0.7024866342544556),
 ('含酒精', 0.6904177665710449),
 ('果汁', 0.6858935356140137),
 ('酒類', 0.6661137938499451),
 ('酒精類', 0.6439155340194702),
 ('酒水', 0.6307460069656372),
 ('提神', 0.620352566242218),
 ('罐裝', 0.6196714043617249),
 ('蘇打水', 0.6155136227607727)]
```

In [20]:
```
model.wv.most_similar("car")
```

```
[('truck', 0.6745814681053162),
 ('tikita', 0.669427752494812),
 ('seat', 0.6667279601097107),
 ('limousine', 0.6198944449424744),
 ('motorcycle', 0.6196877360343933),
 ('cab', 0.612440824508667),
 ('chevrolet', 0.6013472080230713),
 ('pickup', 0.598986029624939),
 ('wagon', 0.5982146859169006),
 ('motor', 0.5973161458969116)]
```

In [21]:
```
model.wv.most_similar("facebook")
```

```
[('臉書', 0.8026980757713318),
 ('專頁', 0.7572149038314819),
 ('面書', 0.743718147277832),
 ('instagram', 0.717136025428772),
 ('貼文', 0.6941859126091003),
 ('twitter', 0.6797687411308289),
 ('推特', 0.6733626127243042),
 ('粉專', 0.6730448603630066),
 ('tumblr', 0.6474182605743408),
 ('粉絲團', 0.647148251533508 3)]
```

In [22]:
```
model.wv.most_similar("詐欺")
```

```
[('欺詐', 0.6929510831832886),
 ('詐騙', 0.5871631503105164),
 ('慣犯', 0.5665764808654785),
 ('詐欺罪', 0.5551669001579285),
 ('竊盜', 0.5519699454307556),
 ('委託人', 0.5280893445014954),
 ('敲詐', 0.5202021598815918),
 ('詐財', 0.5067790150642395),
 ('偽證', 0.5061326622962952),
 ('詐騙犯', 0.5058969259262085)]
```

In [23]:
```
model.wv.most_similar("合約")
```

```
[('合同', 0.7794087529182434),
 ('簽約', 0.7035037875175476),
 ('續約', 0.6858285069465637),
 ('簽下', 0.5998603701591492),
 ('租約', 0.5896501541137695),
 ('短約', 0.5811787843704224),
 ('續簽', 0.5810917019844055),
 ('買斷', 0.5799855589866638),
 ('選擇權', 0.5781211853027344),
 ('解約', 0.5772160887718201)]
```

In [24]:
```
model.wv.similarity("連結", "鍵接")
```

```
0.7021598
```

In [25]:
```
model.wv.similarity("連結", "陰天")
```

```
0.008144689
```

In [26]:
```
print(f"Loading {output_model}...")
new_model = word2vec.Word2Vec.load(output_model)
```

```
Loading word2vec.zh.300.model...
```

In [27]:
```
model.wv.similarity("連結", "陰天") == new_model.wv.similarity("連結", "陰天")
```

```
True
```