In [1]:

```
!pip install memory_profiler
```

```
Requirement already satisfied: memory_profiler in c:\users\user\anaconda3
\lib\site-packages (0.61.0)
Requirement already satisfied: psutil in c:\users\user\anaconda3\lib\site-
packages (from memory_profiler) (5.9.0)
```

In [2]:

```
%load_ext memory_profiler
!pip install -q zhconv
```

In [1]:

```
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 4.3.0
jieba 0.42.1
```

In [ ]:

```
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)
```

In [ ]:

```
import urllib.request
url = "https://dumps.wikimedia.org/zhwiki/20230501/zhwiki-20230501-pages-articles.xml.bz2

filename = "zhwiki-20230501-pages-articles.xml.bz2"
urllib.request.urlretrieve(url, filename)
```

In [12]:

```
import os

ZhWiki = r"C:\Users\user\Desktop\notebook\nlp\HW4\zhwiki-20230501-pages-articles.xml.bz2"

print(f"File size: {os.path.getsize(ZhWiki) / (1024*1024):.2f} MB")
```

```
File size: 2509.82 MB
```

In [3]:

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

Out[3]:

```
'這原本是一段簡體中文'
```

In [4]:

```
seg_list = jieba.cut("我来到北京清华大学", cut_all=True)
print("Full Mode: " + "/ ".join(seg_list))  # 全模式

seg_list = jieba.cut("我来到北京清华大学", cut_all=False)
print("Default Mode: " + "/ ".join(seg_list))  # 精確模式
```

```
Building prefix dict from C:\Users\user\Desktop\notebook\nlp\HW4\dict.txt.
big ...
Loading model from cache C:\Users\user\AppData\Local\Temp\jieba.u1860406d2
d6aafb868e1ddf4bccba943.cache
Loading model cost 0.891 seconds.
Prefix dict has been built successfully.

Full Mode: 我/ 来到/ 北京/ 清华/ 清华大学/ 华大/ 大学
Default Mode: 我/ 来到/ 北京/ 清华大学
```

In [5]:

```
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', 'inter
esting', '吧', '?']
```

In [ ]:

In [ ]:

In [ ]:

In [7]:

```
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_wor
print("--")
print(f"英文停用詞 Total={len(nlp_en.Defaults.stop_words)}: {list(nlp_en.Defaults.stop_wor
```

```
--

中文停用詞 Total=1891: ['一个', '总是', '这种', '以致', '一时', '归根到底',
'以后', '即若', '逐步', '反之', '唯有', '`', '仅', '清楚', '却不', '大概',
'专门', '仅仅', '的话', '着呢'] ...
--
英文停用詞 Total=326: ['thence', 'could', 'alone', 'just', 'regarding', 'wh
ether', 'herself', 'meanwhile', 'noone', 'herein', 'something', 'forty',
'last', 'themselves', 'at', 'ever', 'few', 'amongst', 'on', 'is'] ...
```

In [8]:

```
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 [9]:

```
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 [10]:

```
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
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 [13]:

```
print(f"Parsing {ZhWiki}...")
wiki_corpus = WikiCorpus(ZhWiki, token_min_len=1)
```

```
Parsing C:\Users\user\Desktop\notebook\nlp\HW4\zhwiki-20230501-pages-artic
les.xml.bz2...
```

In [14]:

```
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
的作品', '雅典學院', '数学', '是研究數量', '屬於形式科學的一種', '數學利用抽象化
和邏輯推理', '從計數']
['蘇格拉底之死', '由雅克', '路易', '大卫所繪', '年', '哲學', '是研究普遍的',
'基本问题的学科', '包括存在', '知识']
['文學', '在狭义上', '是一种语言艺术', '亦即使用语言文字为手段', '形象化地反映客
观社会生活', '表达主观作者思想感情的一种艺术', '文学不仅强调传达思想观念', '更强
调传达方式的独特性', '且讲究辞章的美感', '文学']
```

In [15]:

```
WIKI_SEG_TXT = "wiki_seg.txt"

generator = wiki_corpus.get_texts()

with open(WIKI_SEG_TXT, "w", encoding='utf-8') as output:
    for texts_num, tokens in enumerate(generator):
        output.write(" ".join(tokens) + "\n")

        if (texts_num + 1) % 100000 == 0:
            print(f"[{str(dt.now()):.19}] 已寫入 {texts_num} 篇斷詞文章")
```

```
[2023-05-14 02:47:24] 已寫入 99999 篇斷詞文章
[2023-05-14 02:51:14] 已寫入 199999 篇斷詞文章
[2023-05-14 02:58:02] 已寫入 299999 篇斷詞文章
[2023-05-14 03:02:42] 已寫入 399999 篇斷詞文章
```

In [17]:

```
%%time

from gensim.models import word2vec
import multiprocessing

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


sentences = word2vec.LineSentence(WIKI_SEG_TXT)

model = word2vec.Word2Vec(sentences, vector_size=word_dim_size, workers=max_cpu_counts)

output_model = f"word2vec.zh.{word_dim_size}.model"
model.save(output_model)
```

```
Use 12 workers to train Word2Vec (dim=300)
CPU times: total: 36min 29s
Wall time: 9min 28s
```

In [19]:

```
!dir word2vec.zh*
```

```
 磁碟區 C 中的磁碟是 OS
 磁碟區序號:  2E58-FFAB

 C:\Users\user\Desktop\notebook\nlp\HW4 的目錄

2023/05/14  上午 03:41        58,888,453 word2vec.zh.300.model
2023/05/14  上午 03:41     1,894,270,928 word2vec.zh.300.model.syn1neg.npy
2023/05/14  上午 03:40     1,894,270,928 word2vec.zh.300.model.wv.vectors.n
py
               3 個檔案   3,847,430,309 位元組
               0 個目錄  233,171,443,712 位元組可用
```

In [21]:

```
print(model.wv.vectors.shape)
model.wv.vectors
```

```
(1578559, 300)
```

Out[21]:

```
array([[-1.6653749e+00,  1.0125446e+00, -2.3497075e-01, ...,
         6.5279901e-01, -6.3151971e-02, -3.9201072e-01],
       [-1.1377993e+00,  3.5012981e-01, -1.2351167e+00, ...,
         2.3135342e-01,  1.4836991e-01, -2.0512626e+00],
       [-1.2004058e+00,  2.7550453e-01, -1.2185031e+00, ...,
        -7.0264214e-01,  2.3253256e-01, -1.2694845e+00],
       ...,
       [-6.8653323e-02,  5.9258785e-02,  2.8251331e-02, ...,
        -3.3067100e-02,  1.9669712e-02,  7.4995020e-03],
       [-2.5511291e-02,  3.4906086e-02,  3.3190895e-03, ...,
        -2.7539186e-02, -6.2455032e-03,  1.2487747e-03],
       [-2.2402661e-02, -4.6018749e-02,  1.7832810e-02, ...,
         8.2145467e-02, -2.6185357e-03,  3.2317400e-02]], dtype=float32)
```

In [22]:

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

```
(300,)
```

Out[22]:

In [23]:

```
word = "這肯定沒見過 "

try:
    vec = model.wv[word]
except KeyError as e:
    print(e)
```

```
"Key '這肯定沒見過 ' not present"
```

In [24]:

```
model.wv.most_similar("飲料", topn=10)
```

Out[24]:

```
[('飲品', 0.8991072773933411),
 ('服飾', 0.8648388981819153),
 ('化妝品', 0.8595173954963684),
 ('零食', 0.8388224840164185),
 ('冰淇淋', 0.8376069664955139),
 ('手錶', 0.8360370993614197),
 ('食品', 0.8342810869216919),
 ('咖啡', 0.8305132389068604),
 ('炸雞', 0.8282167911529541),
 ('家電', 0.8261498212814331)]
```

In [25]:
```
model.wv.most_similar("car")
```

Out[25]:
```
[('truck', 0.7865666747093201),
 ('motor', 0.7244606614112854),
 ('seat', 0.7219462990760803),
 ('wagon', 0.7197807431221008),
 ('saloon', 0.7161923050880432),
 ('convertible', 0.7087476849555969),
 ('cadillac', 0.7031366825103760),
 ('cab', 0.7030538320541382),
 ('coupe', 0.7013782858848572),
 ('volkswagen', 0.6992166042327881)]
```

In [26]:
```
model.wv.most_similar("facebook")
```

Out[26]:
```
[('instagram', 0.8903433416366577),
 ('臉書', 0.8241993676513671),
 ('專頁', 0.8078523445773827),
 ('twitter', 0.7747393103492737),
 ('myspace', 0.7700318550872803),
 ('facebook專頁', 0.7342762863600159),
 ('新浪微博', 0.7309831380844116),
 ('微博', 0.7263548625572205),
 ('blogger', 0.7184987766422717),
 ('推特', 0.7071648659440536)]
```

In [27]:
```
model.wv.most_similar("詐欺")
```

Out[27]:
```
[('盜竊', 0.8782167434692383),
 ('賣淫', 0.8612630963325195),
 ('欺詐', 0.8544232845306396),
 ('洗錢', 0.8529381752014160),
 ('民事訴訟', 0.8512727022171021),
 ('性騷擾', 0.8501027822494507),
 ('解決問題', 0.8416290879249573),
 ('竊盜', 0.8415544629096985),
 ('和理非', 0.8401445150375366),
 ('誇張', 0.8398345708847046)]
```

In [28]:
```
model.wv.most_similar("合約")
```

Out[28]:
```
[('總值', 0.8186979889869690),
 ('年內', 0.8102273669433594),
 ('耗資超過', 0.7741891741752625),
 ('預算為', 0.7729512453079224),
 ('並被罰款', 0.7723879814147949),
 ('億新台幣', 0.7714751362800598),
 ('據了解', 0.7709003388554382),
 ('花費', 0.7679263353347778),
 ('被罰款', 0.7666801810264587),
 ('萬美金', 0.7616029977798462)]
```

In [29]:
```
model.wv.similarity("連結", "鏈結")
```

Out[29]:
```
0.5345687
```

In [30]:
```
model.wv.similarity("連結", "陰天")
```

Out[30]:
```
0.3341626
```

In [31]:
```
print(f"Loading {output_model}...")
new_model = word2vec.Word2Vec.load(output_model)
```

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

In [32]:
```
model.wv.similarity("連結", "陰天") == new_model.wv.similarity("連結", "陰天")
```

Out[32]:
```
True
```

In [ ]:
```

```