多 媒 體 系 統
MULTIMEDIA SYSTEM

張家瑋 博士

助理教授

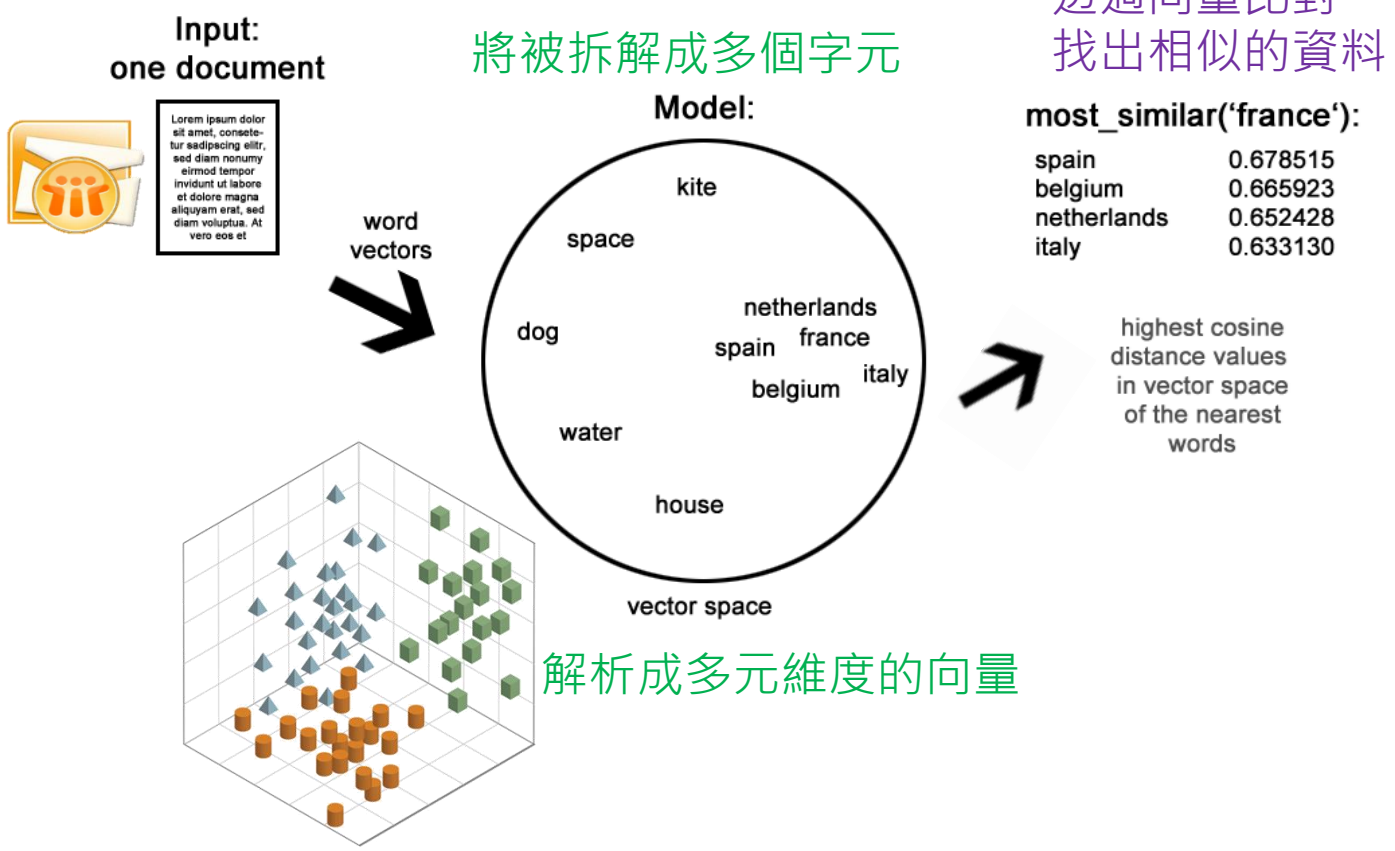
國立臺中科技大學資訊工程系

字詞語意向量實作
WORD2VEC EXERCISE

文字檔案

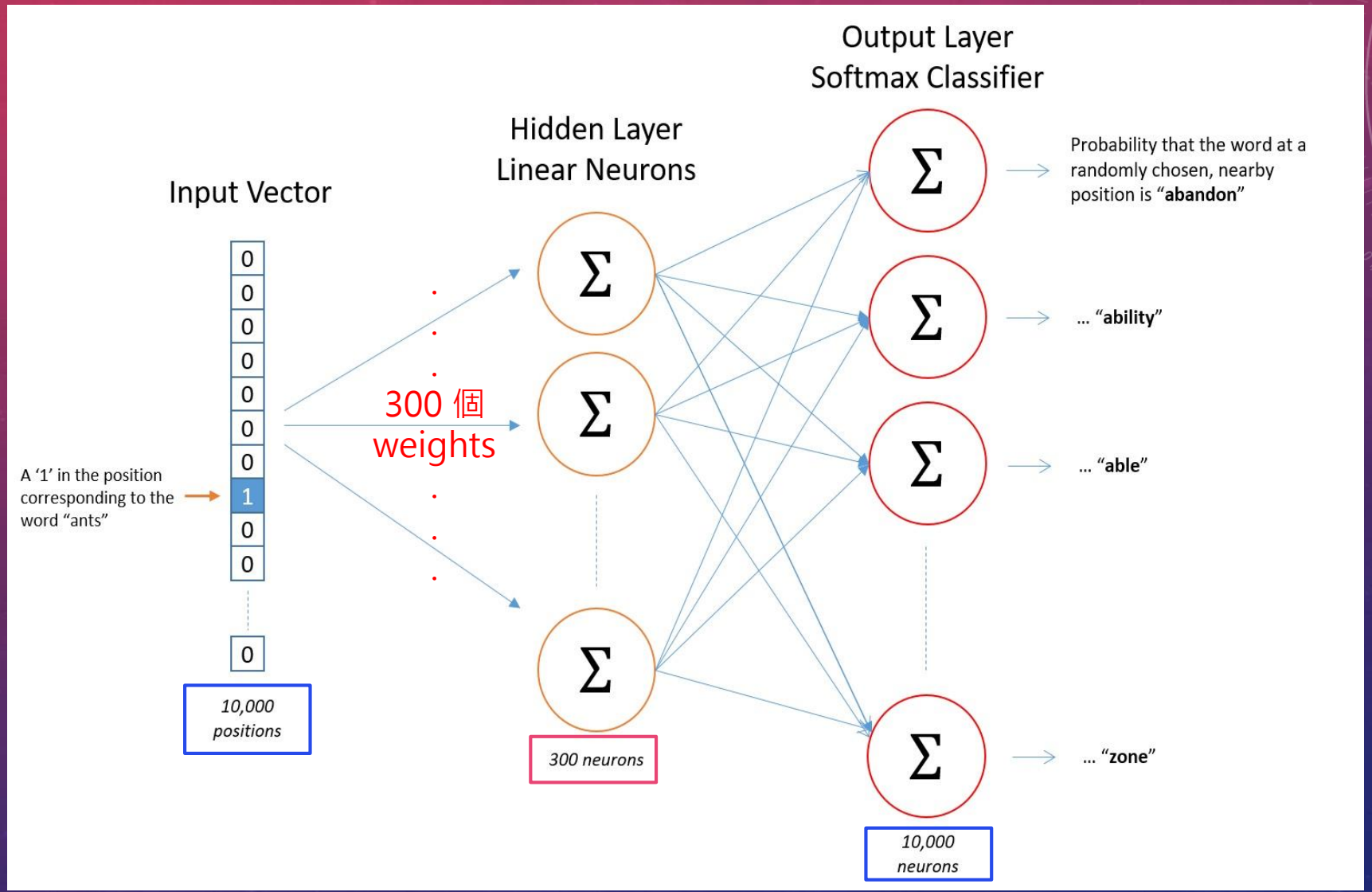
word2vec

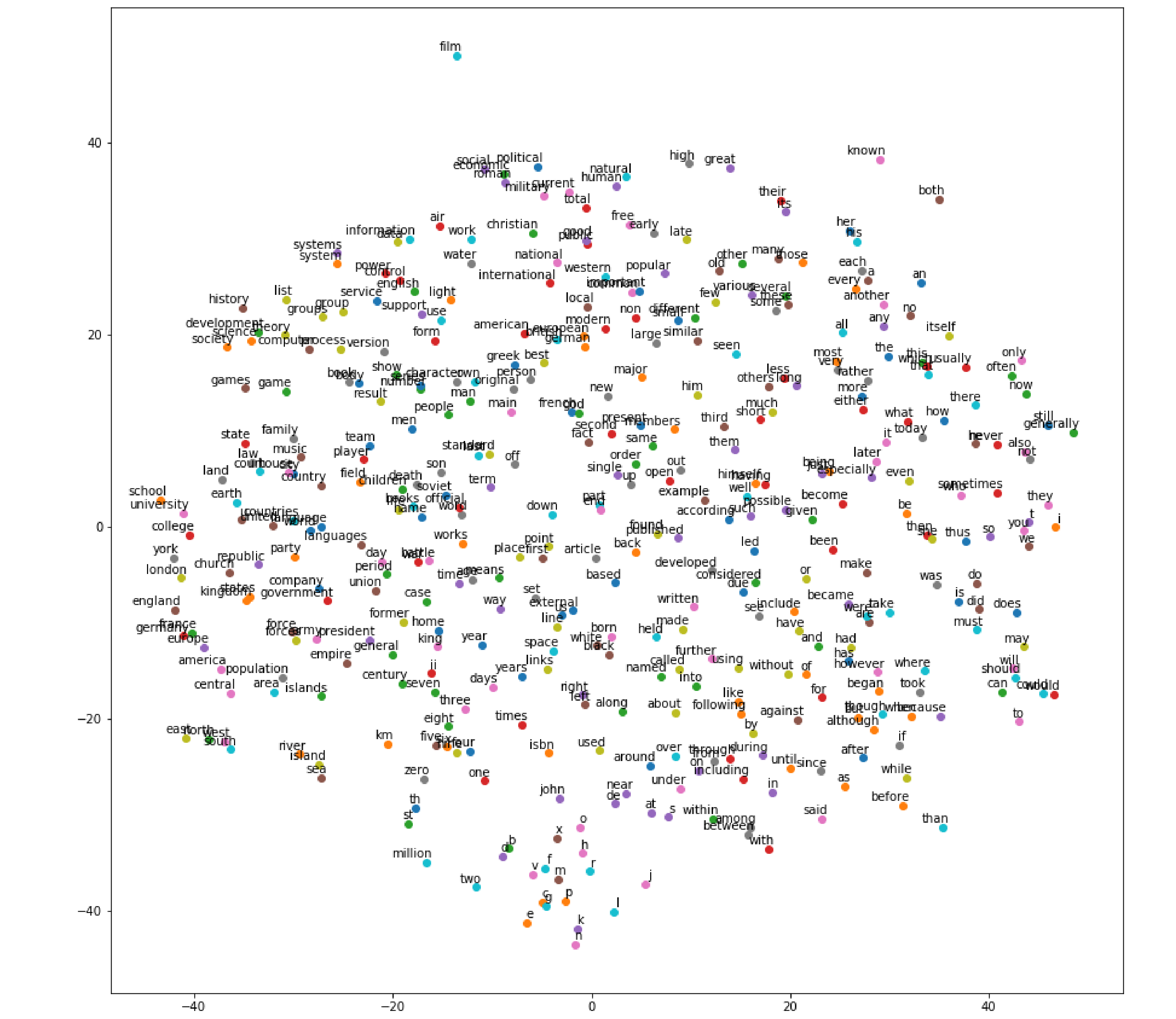
透過向量比對
找出相似的資料



將被拆解成多個字元

解析成多元維度的向量


Output Layer
Softmax Classifier
Hidden Layer
Linear Neurons
Input Vector
Probability that the word at a randomly chosen, nearby position is "abandon"
... "ability"
... "able"
... "zone"
A '1' in the position corresponding to the word "ants"
0
0
0
0
0
0
0
1
0
0
0
10,000 positions
300 個 weights
300 neurons
10,000 neurons


40
20
0
−20
−40
−40
−20
0
20
40
film
known
both
high
great
social
political
economic
roman
military
current
natural
human
total
their
its
her
his
air
work
christian
information
data
systems
system
water
national
western
popular
old
other
many
those
each
a
every
an
another
power
control
english
international
history
list
service
light
local
various
several
few
these
some
no
any
itself
development
theory
science
society
computer
process
version
group
groups
support
use
american
european
british
german
modern
non
different
small
large
similar
seen
all
the
this
which
that
usually
only
often
now
games
game
book
body
show
number
character
own
original
greek
best
person
major
him
most
very
rather
less
long
others
more
either
result
people
man
main
french
old
new
much
short
what
how
there
still
generally
men
state
family
team
player
standard
present
second
members
third
fact
same
them
it
today
never
also
not
law
music
land
society
country
field
death
son
children
soviet
term
off
single
order
out
open
up
himself
having
later
especially
being
even
sometimes
who
they
school
university
earth
official
world
books
name
down
part
end
example
well
possible
become
according
such
given
be
you
t
i
college
countries
united
language
languages
works
point
first
place
article
back
found
published
led
been
then
see
thus
so
we
york
republic
party
day
battle
war
london
church
means
time
period
union
company
developed
considered
due
or
make
was
do
england
kingdom
states
government
case
set
way
external
written
include
became
is
did
does
germany
france
europe
force
forces
army
president
former
home
line
born
held
made
have
were
are
take
must
king
year
space
white
black
further
using
and
had
has
may
america
population
empire
general
links
years
called
named
without
of
however
where
will
should
central
area
islands
century
seven
days
right
left
into
like
for
began
took
can
could
would
three
along
about
following
against
though
when
but
because
although
to
east
north
west
south
eight
times
by
if
km
five
six
four
isbn
used
river
island
sea
over
through
during
from
around
on
including
until
since
after
while
zero
one
under
as
john
near
de
in
at
s
within
among
between
with
said
before
than
th
st
o
x
b
d
h
million
f
r
v
m
two
j
c
g
p
e
l
k
n

INSTALL PACKAGES

- pip install genism
- pip install jieba
- pip install hanziconv

Word2Vec: https://github.com/Alex-CHUN-YU/Word2vec

步驟

1. 將 wiki 的 xml 轉換成 txt (wiki_xml2txt.ipynb)
2. 將簡體轉繁體，並且斷詞去除廢字 (segmentation.ipynb)
3. 訓練Word2Vec模型 (train.ipynb)
4. 使用Word2Vec模型 (WordVec_LoadPretrainModel.ipynb)

下載: https://reurl.cc/vdnee

結果

```
1.輸入一個詞彙會找出前5名相似
2.輸入兩個詞彙會算出兩者之間相似度
3.輸入三個詞彙爸爸之於老公,如媽媽之於老婆
```

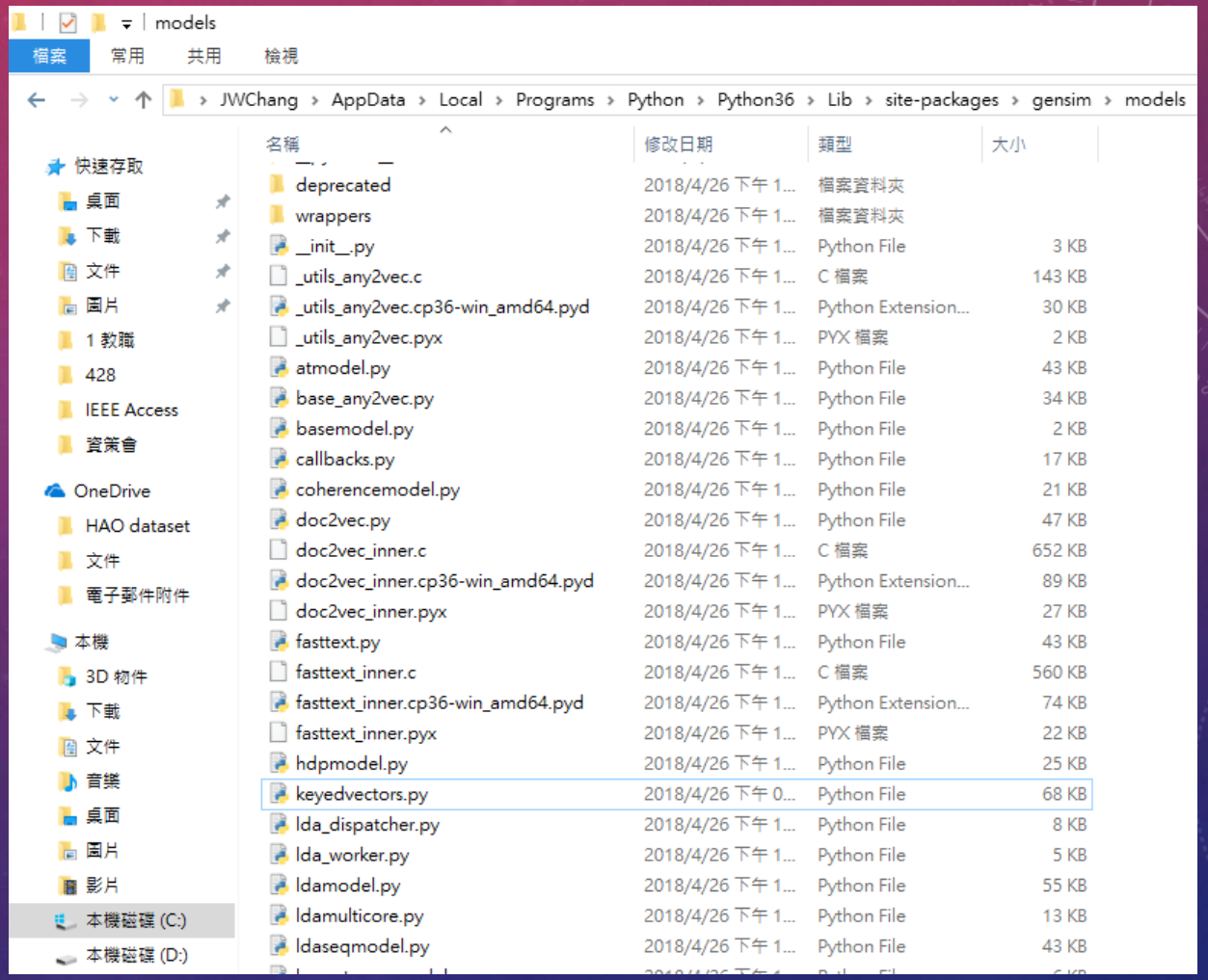
```
輸入格式( Ex: 爸爸,媽媽,....註:最多三個詞彙)
老師
詞彙相似詞前 5 排序
班導,0.6360481977462769
班導師,0.6360464096069336
代課,0.6358826160430908
級任,0.6271134614944458
班主任,0.6270170211791992
```

```
輸入格式( Ex: 爸爸,媽媽,....註:最多三個詞彙)
爸爸,媽媽
計算兩個詞彙間 Cosine 相似度
0.780765200371
```

```
輸入格式( Ex: 爸爸,媽媽,....註:最多三個詞彙)
爸爸,老公,媽媽
爸爸之於老公，如媽媽之於
老婆,0.5401346683502197
蠢萌,0.5245970487594604
夠枰,0.5059393048286438
駭命,0.4888317286968231
孔爵,0.4857243597507477
```

修改 GENSIM

C:\Users\user\AppData\Local\Programs\Python\Python36\Lib\site-packages\gensim\models\keyedvectors.py



修改 GENSIM

```
193 class WordEmbeddingsKeyedVectors(BaseKeyedVectors):
194     """Class containing common methods for operations over word vectors."""
195
196     def __init__(self, vector_size):
197         super(WordEmbeddingsKeyedVectors, self).__init__(vector_size=vector_size)
198         self.vectors_norm = None
199         self.index2word = []
200         self.zeroVec = []
201         for i in range(300):
202             self.zeroVec.append(0.0)
```

修改 GENSIM

```
255     def word_vec(self, word, use_norm=False):
256         """
257         Accept a single word as input.
258         Returns the word's representations in vector space, as a 1D numpy array.
259 
260         If `use_norm` is True, returns the normalized word vector.
261 
262         Examples
263         --------
264         >>> trained_model['office']
265         array([ -1.40128313e-02, ...])
266 
267         """
268         if word in self.vocab:
269             if use_norm:
270                 result = self.vectors_norm[self.vocab[word].index]
271             else:
272                 result = self.vectors[self.vocab[word].index]
273 
274             result.setflags(write=False)
275             return result
276         else:
277             return array(self.zeroVec)
278             #raise KeyError("word '%s' not in vocabulary" % word)
```

THANK YOU