Korean Word2Vec

> 네이버 영화 리뷰 데이터

```
import warnings
warnings.filterwarnings('ignore')
```

I. Install & Import Packages

- Install KoNLPy

```
!pip install konlpy
```

```
Collecting konlpy
  Downloading https://files.pythonhosted.org/packages/85/0e/f385566fec837c0b83f216b2da65db999
     |████████████████████████████████| 19.4MB 29.1MB/s
Requirement already satisfied: numpy>=1.6 in /usr/local/lib/python3.7/dist-packages (from kon
Collecting beautifulsoup4==4.6.0
  Downloading https://files.pythonhosted.org/packages/9e/d4/10f46e5cfac773e22707237bfcd51bbff
     |████████████████████████████████| 92kB 6.4MB/s
Collecting JPype1>=0.7.0
  Downloading https://files.pythonhosted.org/packages/cd/a5/9781e2ef4ca92d09912c4794642c1653a
     |████████████████████████████████| 460kB 42.6MB/s
Requirement already satisfied: tweepy>=3.7.0 in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: lxml>=4.1.0 in /usr/local/lib/python3.7/dist-packages (from ko
Collecting colorama
  Downloading https://files.pythonhosted.org/packages/44/98/5b86278fbbf250d239ae0ecb724f8572a
Requirement already satisfied: typing-extensions; python_version < "3.8" in /usr/local/lib/py
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: six>=1.10.0 in /usr/local/lib/python3.7/dist-packages (from tw
Requirement already satisfied: requests[socks]>=2.11.1 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/pyth
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (from r
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: PySocks!=1.5.7,>=1.5.6; extra == "socks" in /usr/local/lib/pyt
Installing collected packages: beautifulsoup4, JPype1, colorama, konlpy
  Found existing installation: beautifulsoup4 4.6.3
    Uninstalling beautifulsoup4-4.6.3:
      Successfully uninstalled beautifulsoup4-4.6.3
Successfully installed JPype1-1.2.1 beautifulsoup4-4.6.0 colorama-0.4.4 konlpy-0.5.2
```

- Import Packages

```
import pandas as pd
import matplotlib.pyplot as plt
```

II. Data Preprocessing

1) naverRatings.zip

- Google Drive Mount

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
!ls -l /content/drive/MyW Drive/ColabW Notebooks/datasets/naverRatings.zip
```

```
-rw------- 1 root root 7903524 May  4  2020 '/content/drive/My Drive/Colab Notebooks/datasets
```

```
!unzip /content/drive/MyW Drive/ColabW Notebooks/datasets/naverRatings.zip
```

```
Archive:  /content/drive/My Drive/Colab Notebooks/datasets/naverRatings.zip
  inflating: naverRatings.txt
```

```
!ls -l naverRatings.txt
```

```
-rw-r--r-- 1 root root 19515078 May  4  2020 naverRatings.txt
```

2) 데이터 읽어오기

- Label : '1'(긍정), '0'(부정)

```
train_data = pd.read_table('naverRatings.txt')

train_data[:5]
```

	id	document	label
0	8112052	어릴때보고 지금다시봐도 재밌어요ㅋㅋ	1
1	8132799	디자인을 배우는 학생으로, 외국디자이너와 그들이 일군 전통을 통해 발전해 가는 문화산...	1
2	4655635	폴리스스토리 시리즈는 1부터 뉴까지 버릴께 하나도 없음.. 최고.	1
3	9251303	와.. 연기가 진짜 개쩔구나.. 지루할거라고 생각했는데 몰입해서 봤다.. 그래 이런...	1

- 네이버 영화 리뷰 개수

```
print(len(train_data))
```

```
200000
```

3) 데이터 정제(Cleaning)

- NULL값 존재 확인

```
print(train_data.isnull().values.any())
```

```
True
```

- NULL값 존재 행 제거 후 재확인

```
train_data = train_data.dropna(how = 'any')

print(train_data.isnull().values.any())
```

```
False
```

- NULL값 제거 후 데이터 개수

```
print(len(train_data))
```

```
199992
```

4) 정규표현식을 통한 한글 외 문자 제거

```
train_data['document'] = train_data['document'].str.replace('[^ㄱ-ㅎㅏ-ㅣ가-힣 ]', '')
```

- 처리 결과 확인

```
train_data[:5]
```

	id	docur
0	8112052	어릴때보고 지금다시봐도 재밌어요
1	8132799	디자인을 배우는 학생으로 외국디자이너와 그들이 일군 전통을 통해 발전해가는 문화산
2	4655635	폴리스스토리 시리즈는 부터 뉴까지 버릴께 하나도 없음
3	9251303	와 연기가 진짜 개쩔구나 지루할거라고 생각했는데 몰입해서 봤다 그래 이런게 진짜 영
4	10067386	안개 자욱한 밤하늘에 떠 있는 초승달 같은

```
train_data['document'].shape
```

```
(199992,)
```

5) 불용어(Stopword) 지정

```
stopwords = ['의','가','이','은','들','는','좀','잘','걍',W
             '과','도','를','으로','자','에','와','한','하다']
```

6) Okt( )를 활용한 토큰화 및 불용어 제거

- 10분 소요

```
%%time

from konlpy.tag import Okt

okt = Okt()
tokenized_data = []

for sentence in train_data['document']:
    temp_X = okt.morphs(sentence,
                        stem = True)
    temp_X = [word for word in temp_X if not word in stopwords]
    tokenized_data.append(temp_X)
```

```
CPU times: user 13min 35s, sys: 5.59 s, total: 13min 41s
Wall time: 14min 2s
```

```
len(tokenized_data)
```

```
199992
```

III. 리뷰 데이터 분포 시각화

1) 리뷰 길이 확인

```
print('리뷰의 최대 길이 :',max(len(l) for l in tokenized_data))
print('리뷰의 평균 길이 :',sum(map(len, tokenized_data))/len(tokenized_data))
```

```
리뷰의 최대 길이 : 72
리뷰의 평균 길이 : 10.716703668146726
```

2) 리뷰 길이 시각화

```
plt.hist([len(s) for s in tokenized_data], bins = 50)
plt.xlabel('length of samples')
plt.ylabel('number of samples')
plt.show()
```

IV. Word2Vec 수행

1) 임베딩 학습

- Vector 차원 : 100
- Window 크기 : 5
- size = 워드 벡터의 특징 값. 즉, 임베딩 된 벡터의 차원.
- window = 컨텍스트 윈도우 크기
- min_count = 단어 최소 빈도 수 제한 (빈도가 적은 단어들은 학습하지 않는다.)
- workers = 학습을 위한 프로세스 수
- sg = 0은 CBOW, 1은 Skip-gram

```
from gensim.models import Word2Vec

model = Word2Vec(sentences = tokenized_data,
                 size = 100,
                 window = 5,
                 min_count = 5,
                 workers = 4,
                 sg = 1)
```

2) 학습된 임베딩 매트릭스 크기 확인

```
model.wv.vectors.shape
```

```
(16477, 100)
```

V. 임베딩 결과 테스트

```
model.wv.most_similar('이병헌')
```

```
[('공리', 0.8348996639251709),
```

```
('최민식', 0.8100616931915283),
('심은하', 0.8070361614227295),
('주진모', 0.805631697177887),
('유다인', 0.8006829619407654),
('최수종', 0.7995752692222595),
('김명민', 0.7947758436203003),
('정려원', 0.7944632768630981),
('고소영', 0.7922888994216919),
('김창완', 0.7905289530754089)]
```

```
model.wv.most_similar('액션')
```

```
[('액션씬', 0.7082537412643433),
 ('격투씬', 0.7056965827941895),
 ('볼거리', 0.7043814659118652),
 ('무술', 0.6873269081115723),
 ('격투', 0.6828569173812866),
 ('디도', 0.6745415925979614),
 ('추격', 0.6702743768692017),
 ('레이싱', 0.6695525646209717),
 ('액션영화', 0.6640169620513916),
 ('스릴러물', 0.6607005596160889)]
```

VI. 사전훈련된 Word2Vec

1) Google Drive Mount

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/c
```

```
!ls -l /content/drive/MyW Drive/ColabW Notebooks/datasets/ko_w2v.zip
```

```
-rw------- 1 root root 80596565 May  4  2020 '/content/drive/My Drive/Colab Notebooks/dataset
```

2) Unzip 'ko_w2v.zip'

```
!unzip /content/drive/MyW Drive/ColabW Notebooks/datasets/ko_w2v.zip
```

```
Archive:  /content/drive/My Drive/Colab Notebooks/datasets/ko_w2v.zip
  inflating: ko.bin
  inflating: ko.tsv
```

```
!ls -l ko.bin
```

```
-rw------- 1 root root 50697568 Dec 21  2016 ko.bin
```

3) Word2Vec 가져오기

```
import gensim
```

```
model = gensim.models.Word2Vec.load('ko.bin')
```

4) Word2Vec Test

```
model.wv.most_similar('금융')
```

```
[('감독원', 0.6556380391120911),
 ('신용', 0.6269841194152832),
 ('은행', 0.6236893534660339),
 ('외환', 0.6192121505737305),
 ('중소기업', 0.6051731705665588),
 ('중앙은행', 0.6050782799720764),
 ('증권', 0.5907014608383179),
 ('거래', 0.5898198485374451),
 ('투자', 0.5844753384590149),
 ('경영', 0.5692520141601562)]
```

```
model.wv.most_similar('은행')
```

```
[('씨티', 0.6328796148300171),
 ('금융', 0.6236892938613892),
 ('농협', 0.6008170247077942),
 ('본점', 0.5930641889572144),
 ('한국은행', 0.5903059840202332),
 ('지점장', 0.5847948789596558),
 ('은행장', 0.5830198526382446),
 ('거래소', 0.5792121887207031),
 ('증권', 0.5775279998779297),
 ('외환', 0.5623738765716553)]
```

#

#

#

The End

#

#

#