Ozetleme

November 25, 2021

[1]:
```
import re
import time
import numpy as np
import pandas as pd
from gensim.models import KeyedVectors
from sklearn.metrics.pairwise import cosine_similarity
```

word2vec Turkish: https://github.com/akoksal/Turkish-Word2Vec

[2]:
```
start = time.time()

word_vectors = KeyedVectors.load_word2vec_format('trmodel', binary=True)

print("total time: ", time.time() - start)
```

```
total time:  4.104509115219116
```

Turkce stopwords lerden kurtul : https://github.com/sgsinclair/trombone/blob/master/src/main/resources/org/v
lucene.txt

Kelime kokune in

[3]:
```
url ='https://raw.githubusercontent.com/sgsinclair/trombone/master/src/main/
 ↪resources/org/voyanttools/trombone/keywords/stop.tr.turkish-lucene.txt'
data = pd.read_csv(url)
```

[4]:
```
data
```

[4]:
```
                    # Turkish stopwords from LUCENE-559
0    # merged with the list from "Information Retri…
1    #   (http://www.users.muohio.edu/canf/papers/J…
2                                               acaba
3                                              altmış
4                                                altı
..                                                …
206                                             yine
207                                            yirmi
208                                            yoksa
209                                              yüz
```

```
210                                                    zaten

[211 rows x 1 columns]
```

[5]:
```
stpwrds = set(data[2:].iloc[:, 0].values)
```

[6]:
```
def vectorize(cumle, stpwrds = stpwrds):
    kelimeler = cumle.split()
    vc = np.zeros((1, 400))
    n = 0
    for v in kelimeler:
        if v in stpwrds:
            continue
        try:
            vc += word_vectors[v].reshape(1,400)
            n +=1
        except:
            continue
    vc /= n
    return vc
```

1 Ozetleme

[7]:
```
sentences = ['Fenerbahçe spor klubü ciddi başarılar kazandı',
             'Damlayan su, taşı deler. Ovidius',
             'Hazine, eziyet çekene gözükür.',
             'Fenerbahçe vurdu gol oldu',
             'Galatasaray voleybol takımı kazanır']

clean_sentences = [re.sub('[;!@#’‘?.,\'$]', '', s).lower() for s in sentences]
clean_sentences
```

[7]:
```
['fenerbahçe spor klubü ciddi başarılar kazandı',
 'damlayan su taşı deler ovidius',
 'hazine eziyet çekene gözükür',
 'fenerbahçe vurdu gol oldu',
 'galatasaray voleybol takımı kazanır']
```

[8]:
```
vectors = [vectorize(s) for s in clean_sentences]
```

[9]:
```
n = len(sentences)
X = np.concatenate(vectors).reshape(n,1,400)
X.shape
```

[9]:
```
(5, 1, 400)
```

[10]:
```
sim_mat = np.zeros((n,n))
for i in range(n):
    for j in range(n):
        sim_mat[i,j] = cosine_similarity(X[i], X[j])[0,0]
```

[11]:
```
import networkx as nx

nx_graph = nx.from_numpy_array(sim_mat)
scores = nx.pagerank(nx_graph)
```

[12]:
```
ranked_sentences = sorted(((scores[i],s) for i,s in enumerate(sentences)),␣
 ↪reverse=True)
# Specify number of sentences to form the summary
sn = 3

# Generate summary
for i in range(sn):
    print(ranked_sentences[i][1])
```

```
Fenerbahçe vurdu gol oldu
Galatasaray voleybol takımı kazanır
Fenerbahçe spor klubü ciddi başarılar kazandı
```

[13]:
```
scores
```

[13]:
```
{0: 0.19640726226766067,
 1: 0.19608643612516974,
 2: 0.1959043809471848,
 3: 0.20764884375481069,
 4: 0.2039530769051744}
```

[ ]:
```

```