산업인공지능 개론

Home Work #2
Knowledge Graph

학과 : 산업인공지능학과
학번 : 2024254022
이름 : 정현일

2024.03.31.

KnowledgeGraph.py (1/2)

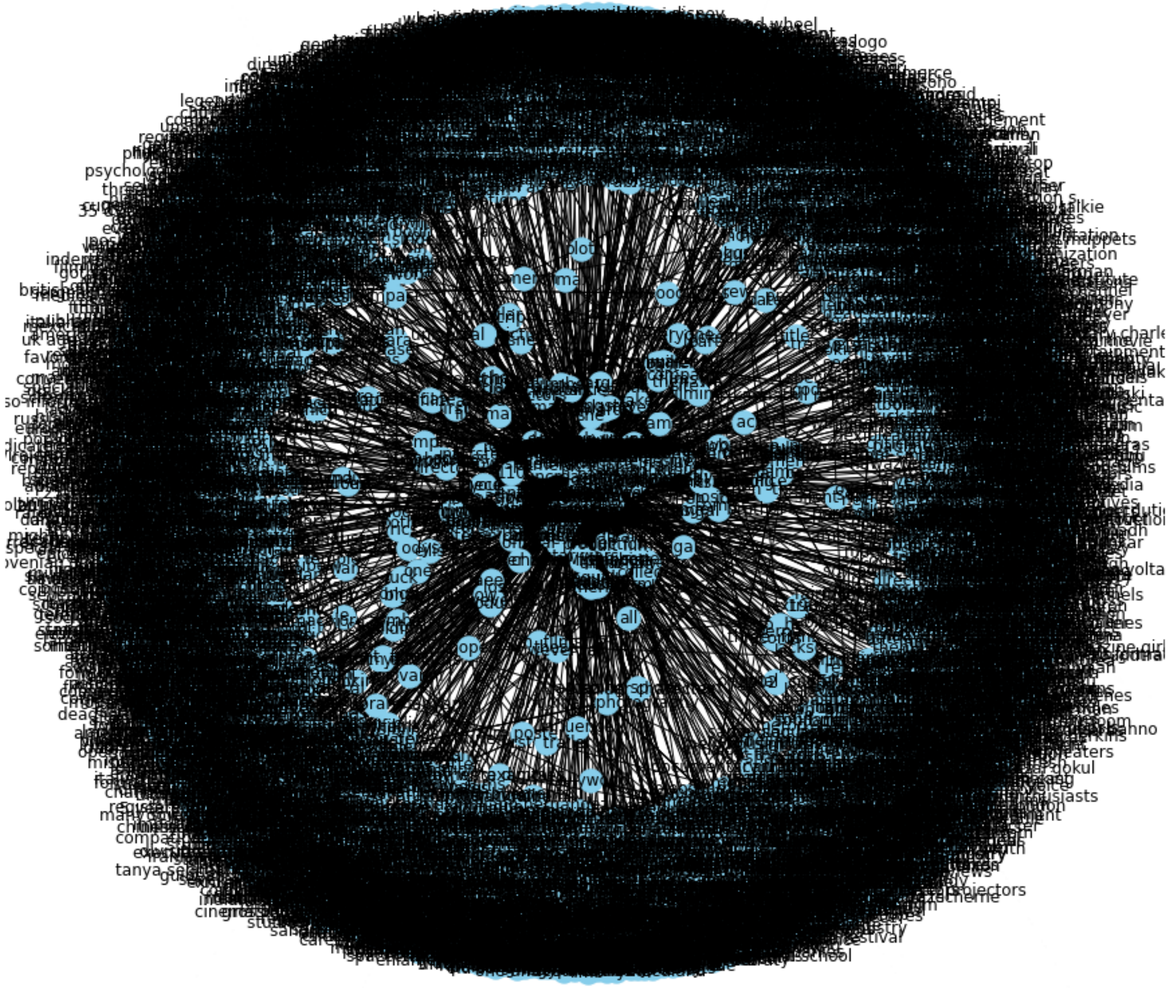
```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
산업인지능 개론

HW2
강의노트 3.2장에서 Knowledge Graph를 구성하는 코드를 실행하고, 구성된 knowledge graph에서
   written_by나 composed_by가 아닌 2개의 관계를 선택하여 해당되는 정보를 추출

학번 : 2024254022
이름 : 정현일

@author: chohi
"""


import re
import pandas as pd
import bs4
import requests
import spacy

from spacy import displacy
nlp = spacy.load('en_core_web_sm')

from spacy.matcher import Matcher
from spacy.tokens import Span

import networkx as nx

import matplotlib.pyplot as plt
from tqdm import tqdm


pd.set_option('display.max_colwidth', 200)
%matplotlib inline


candidate_sentences = pd.read_csv('wiki_sentences_v2.csv')
print(candidate_sentences.shape)
print(candidate_sentences)

entiry_pairs = []


def get_entities(sent):
    ent1 = ""
    ent2 = ""
    prv_tok_dep = ""
    prv_tok_text = ""
    prefix = ""
    modifier = ""
    for tok in nlp(sent):
        # 토큰이 구두점(punctuation mark)이면 다음 토큰으로 이동
        if tok.dep_ != "punct":
            if tok.dep_ == "compound": # 토큰이 복합어인 경우
                prefix = tok.text
                if prv_tok_dep == "compound": # 직전 토큰이 복합어이면 현재 토큰과 결합
                    prefix = prv_tok_text + " " + tok.text
            if tok.dep_.endswith("mod") == True: # 토큰이 수식어(modifier)인 경우
                modifier = tok.text
                if prv_tok_dep == "compound": # 직전 톡ㄴ이 수식어이면 현재 토큰을 결합
                    modifier = prv_tok_text + " " + tok.text

            if tok.dep_.find("subj") == True: # 주어(subject)인 경우,
                ent1 = modifier + " " + prefix + " " + tok.text # 수식어와 현재 토큰 결합 => 개체명 생성
                prefix = ""
                modifier = ""
                prv_tok_dep = ""
                prv_tok_text = ""
            if tok.dep_.find("obj") == True: # 목적어인 경우
                ent2 = modifier + " " + prefix + " " + tok.text # 수식어와 현재 토큰 결합 => 객체명 생성

            prv_tok_dep = tok.dep_
            prv_tok_text = tok.text
    return [ent1.strip(), ent2.strip()] # 식별된 개체명 반환



def get_relation(sent):
    doc = nlp(sent)
    matcher = Matcher(nlp.vocab)

    # 패턴 정의
    pattern = [{'DEP' : 'ROOT'},
               {'DEP' : 'prep', 'OP' : "?"},
               {'DEP' : 'agnet', 'OP' : "?"},
               {'POS' : 'ADJ', 'OP' : "?"}]

    matcher.add('matching_1', [pattern])

    matches = matcher(doc)
    print('matches : ', matches)
    k = len(matches) - 1

    span = doc[matches[k][1]:matches[k][2]]

    return(span.text)
```

KnowledgeGraph.py (2/2)

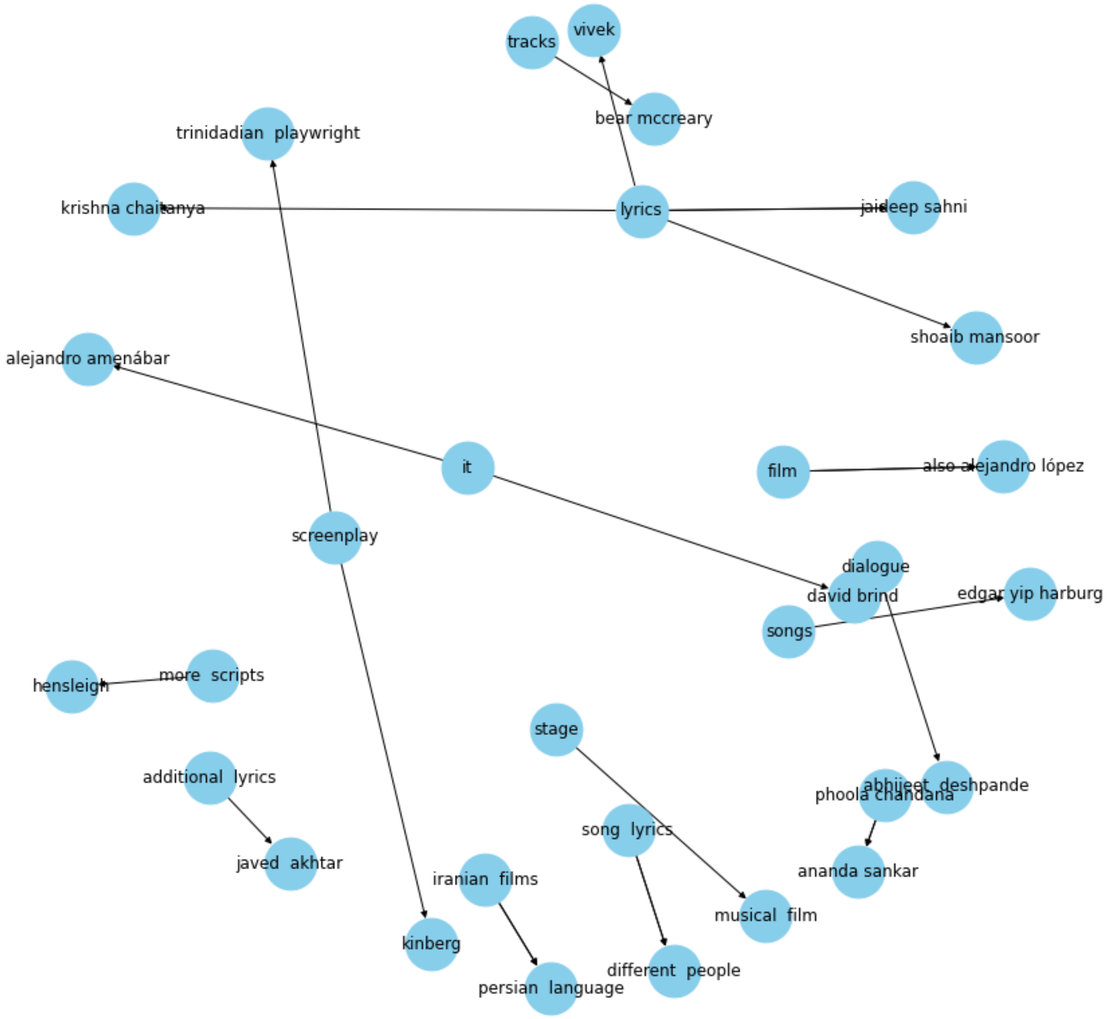
```
get_relation("John completed the task")
relations = [get_relation(i) for i in tqdm(candidate_sentences['sentence'])]



for i in tqdm(candidate_sentences["sentence"]):
    entiry_pairs.append(get_entities(i))

entiry_pairs[10:20]


# 주어(subject) 추출
source = [i[0] for i in entiry_pairs]


# 목적어(object) 추출
target = [i[1] for i in entiry_pairs]

kg_df = pd.DataFrame({'source' : source, 'target' : target, 'edge' : relations})



# 방향 그래프 생성
G = nx.from_pandas_edgelist(kg_df, "source", "target",
                            edge_attr=True, create_using=nx.MultiDiGraph())

# 그래프 그리기
plt.figure(figsize=(12,12))
pos = nx.spring_layout(G)
nx.draw(G, with_labels=True, node_color='skyblue', edge_cmap=plt.cm.Blues, pos=pos)
plt.show()



## written
G=nx.from_pandas_edgelist(kg_df[kg_df['edge']=="written"], "source", "target",
                          edge_attr=True, create_using=nx.MultiDiGraph())

plt.figure(figsize=(12,12))
pos = nx.spring_layout(G, k = 0.5)
nx.draw(G, with_labels=True, node_color='skyblue', node_size=1500, edge_cmap=plt.cm.Blues, pos = pos)
plt.show()



## include
B = nx.from_pandas_edgelist(kg_df[kg_df['edge'] == "include"], "source", "target",
                            edge_attr=True, create_using=nx.MultiDiGraph())
```

```
plt.figure(figsize=(12,12))
pos = nx.spring_layout(G, k=0.5) # k refulates the distance between nodes
nx.draw(G, with_labels=True, node_color='skyblue', node_size=1500, edge_cmap=plt.cm.Blues, pos = pos)
plt.show()
```

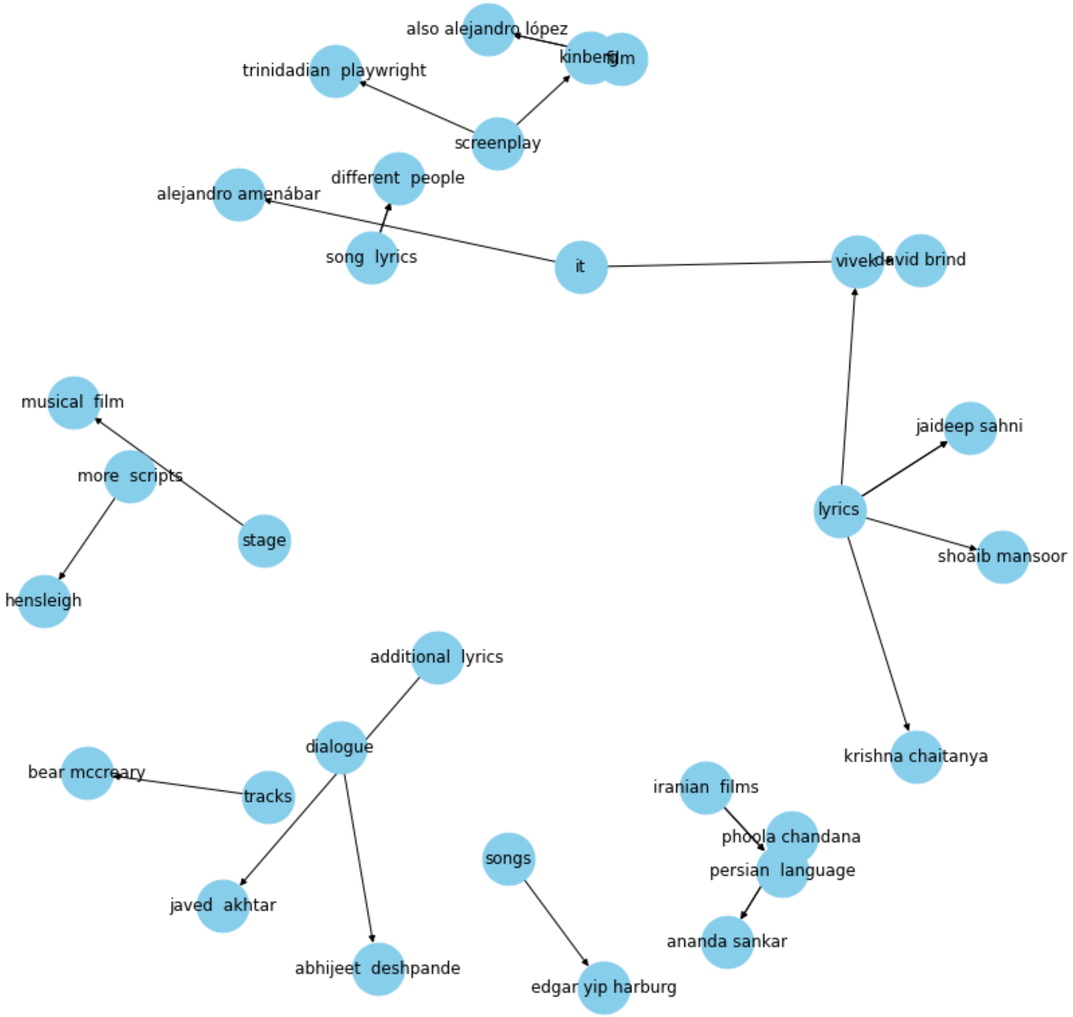
실행 결과



실행 결과


tracks
vivek
bear mccreary
trinidadian playwright
krishna chaitanya
lyrics
jaideep sahni
shoaib mansoor
alejandro amenábar
it
film
also alejandro lópez
screenplay
dialogue
david brind
edgar yip harburg
songs
hensleigh
more scripts
stage
additional lyrics
abhijeet deshpande
phoola chandana
song lyrics
javed akhtar
ananda sankar
iranian films
musical film
kinberg
different people
persian language

실행 결과


also alejandro lópez
kinberg
film
trinidadian playwright
screenplay
different people
alejandro amenábar
song lyrics
it
vivek
david brind
musical film
more scripts
stage
hensleigh
jaideep sahni
lyrics
shoaib mansoor
krishna chaitanya
additional lyrics
dialogue
bear mccreary
tracks
javed akhtar
abhijeet deshpande
songs
edgar yip harburg
iranian films
phoola chandana
persian language
ananda sankar