**Location Sentiment Analysis**

동국대학교 컴퓨터공학과 / 인공지능협업센터

2021.04.12 작성

서상현 박사과정

|  |
| --- |
| 1. 실행 가이드 및 분석 절차 2. 분석 결과 (예시) 3. 활용방법 |

1. 실행 가이드 및 분석 절차
   1. 구글 드라이브에 파일이동

<https://drive.google.com/drive/u/1/folders/1GVHsdoW8msQo49cIUe6SpZj18CSx0XTV>

**텍스트이(가) 표시된 사진

자동 생성된 설명**

공유된 google drive에는 a tale of two cities, the overcoat, father goriot 등의 각 소설에 대한 txt파일과 해당 텍스트 파일을 이용하여 감성분석을 수행하는 .ipynb 파일이 있음. Scatter\_plot 폴더에는 분석에 대한 시각화 자료가 저장되며, word2vec\_model 폴더에는 소설 텍스트 데이터로부터 학습된 워드임베딩 모델이 저장됨

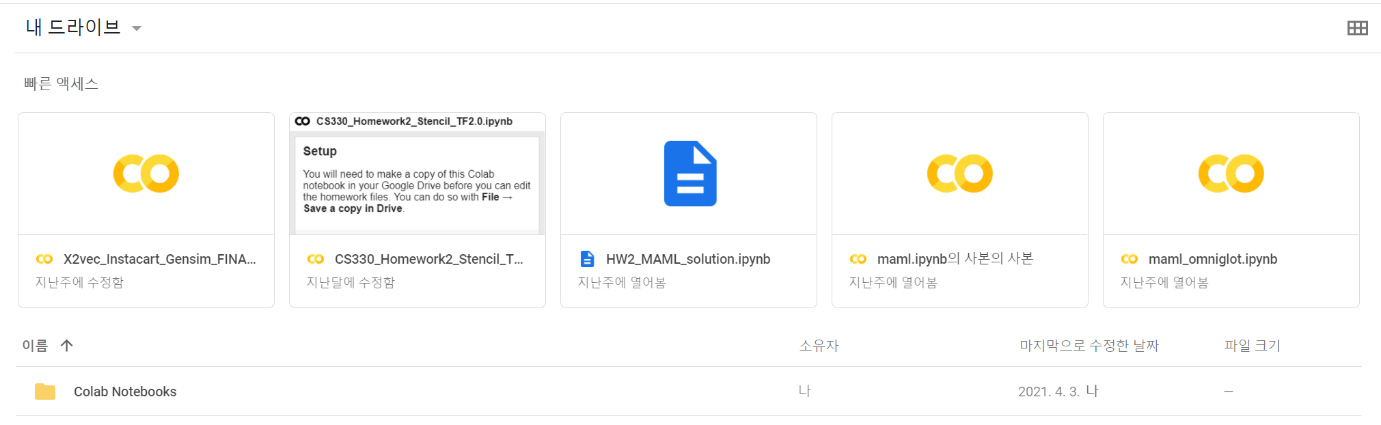
* 전체 드래그 🡪 우클릭 🡪 다운로드  
  텍스트이(가) 표시된 사진

  자동 생성된 설명
* 다운로드 진행 🡪 파일 압축 해제

텍스트이(가) 표시된 사진

자동 생성된 설명

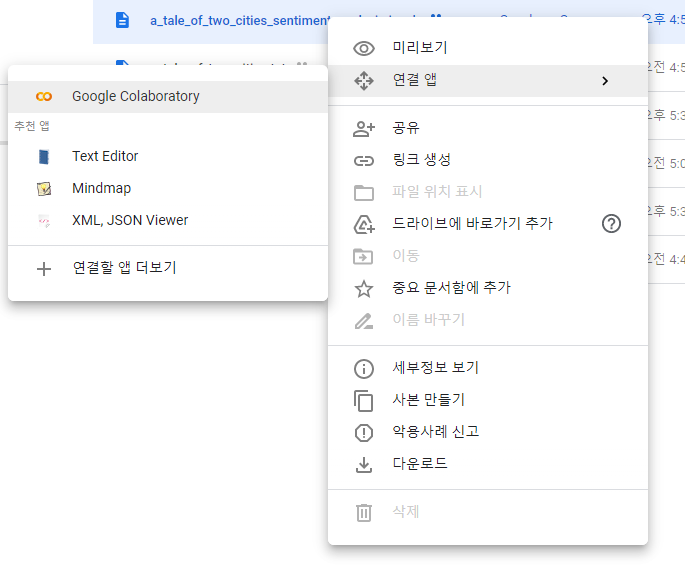
* 본인 구글 계정의 구글 드라이브 이동 🡪 “Colab Notebooks” 폴더 생성 후 해당 폴더로 이동



* Novel\_sentiment\_analysis 폴더 생성 후 해당 폴더로 이동 🡪 다운로드 한 파일 업로드(파일을 드래그 하면 업로드 가능)  
  텍스트이(가) 표시된 사진

  자동 생성된 설명
  1. 전체코드 실행

1. **.ipynb 파일 실행시 google colaboratory로 실행**



1. 런타임 🡪 모두 실행 클릭

텍스트이(가) 표시된 사진

자동 생성된 설명

1. 출력된 url 클릭 🡪 계정 선택(해당 계정의 구글 드라이브 이용) 🡪 구글 계정 엑세스 허용

텍스트, 스크린샷, 모니터이(가) 표시된 사진

자동 생성된 설명

1. Authorization code 복사 🡪 실행 셀에 붙여넣기 🡪 엔터

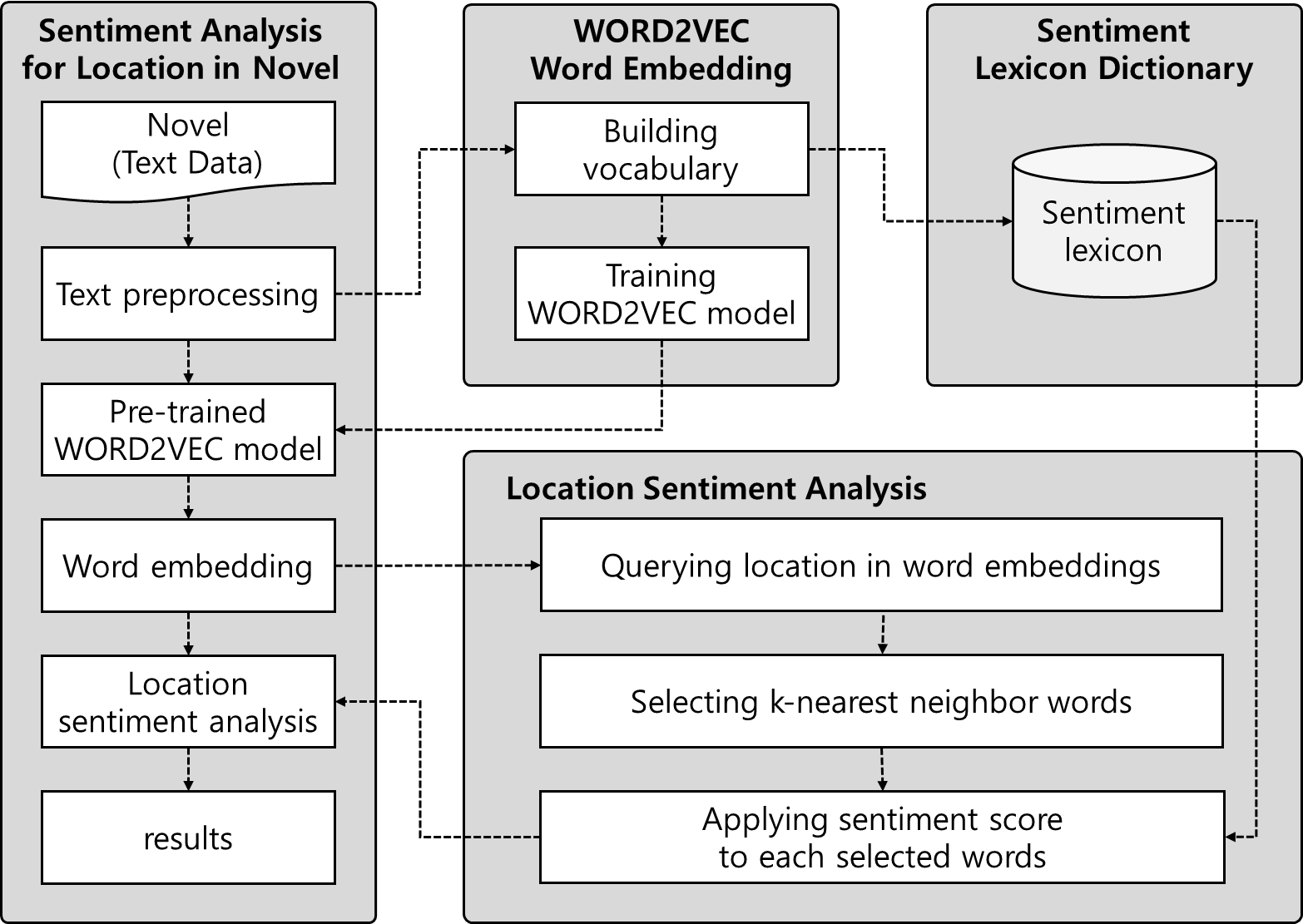
텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

1. 전체 스크립트가 실행되며, 소설 텍스트 데이터에 대한 워드임베딩 모델 학습 및 이를 기반으로 한 감성분석 수행

개념적으로 감성분석은 아래의 도표와 같이 진행되며, 프로그래밍도 이 절차에 따라서 진행됨



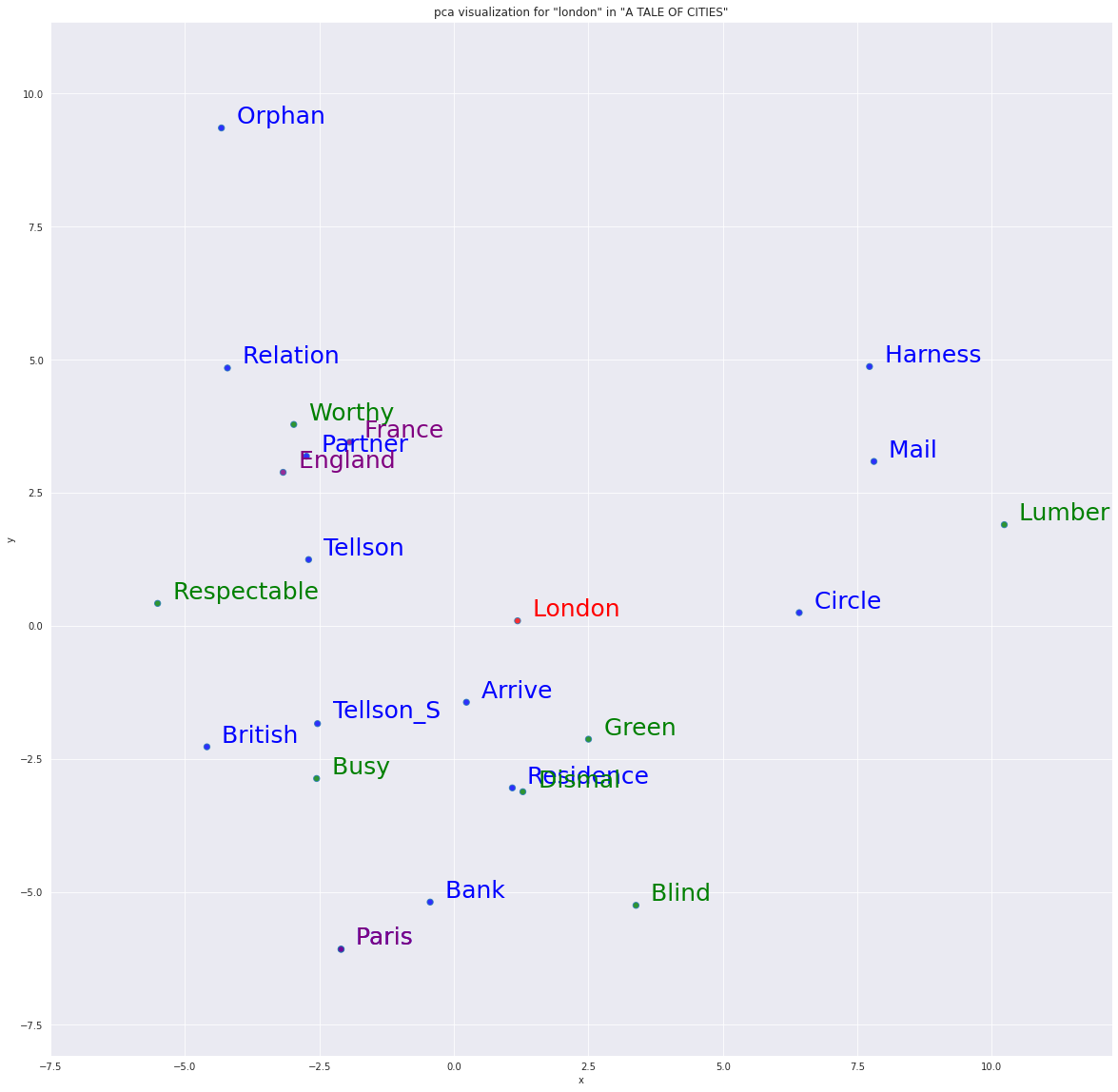
<.ipynb 코드 실행 순서>

1. Google Drive Amount: 실행 파일과 구글 드라이브를 연동하고, 작업 디렉토리를 셋팅
2. Import Library: 감성분석에 필요한 주요 라이브러리를 Importing
3. Loading Text File: 학습 대상이 되는 소설 텍스트 데이터를 작업 가능하도록 로드
4. Checking Text Data Frame: 텍스트 데이터를 Dataframe 형식으로 변형하며, 문장 단위로 작업을 수행하도록 변형
5. Preprocessing Text Data: 대소문자로 구성된 텍스트를 모두 대문자로 바꾸고, 알파벳 이외의 모든 특수기호를 제거, 단어의 개수가 2 이상인 문장만 남김(단어의 개수가 2 이하인 문장은 모델의 학습에 불필요)
6. Training Word2Vec Model: 소설 텍스트 데이터를 기반으로 word2vec 모델을 학습 수행, 소설 내에서 최소 5번 이상 등장한 단어에 대해서만 학습 진행, 총 1000번의 반복 학습 진행, 하나의 단어를 300차원의 벡터로 표현
7. Word Embedding: 학습된 word2vec 모델에서 총 단어의 수와 embedding된 word vector의 출력 예시 확인
8. Location Sentiment Analysis: 공간 감성분석 수행,   
   1) 도시(예: ‘london’) 단어를 입력  
   2) 가장 유사한 k개의 이웃단어를 검색 (예제에서는 k를 20으로 설정)  
   3) 검색된 각각의 이웃 단어를 SentiWordNet의 감성사전에 입력하여 동의어 집합인 SynSet의 모든 단어들을 확인  
   4) SynSet의 단어들에 있는 Positive Score, Negative Score, Objective Score를 확인(score는 0이 최소치, 1이 최대치를 의미)  
   5) 2-(Positive Score + Negative Score)의 수식으로 계산되는 객관성 점수(objective score)가 높은 단어들을 제외하고, 감성을 나타내는데 유의미한 단어들만을 실제 감성분석에 사용 (예제에서는 objective score가 사용 기준치는 0.25보다 낮을 때에만 분석에 활용)  
   6) 각 이웃단어들이 SynSet에서 objective score가 0.25보다 낮은 감성단어들만을 활용하여 positive score와 negative score를 모두 고려한 평균값으로 최종 감성점수를 결정  
   7) 도시 단어의 이웃 단어들 중 감성단어로 선택된 단어들의 최종 감성점수의 평균값을 통해 해당 도시 단어의 감성점수를 확정
9. Visualization: 분석한 결과로서 아래의 네 종류의 단어벡터들을 2차원 시각화  
   1) 도시 단어의 벡터(red color)  
   2) 도시 단어의 이웃 단어의 벡터(blue color)  
   3) 이웃단어들 중 감성단어의 벡터(green color)  
   4) 기타 내가 관계를 확인하고 싶은 단어의 벡터 (purple color)
10. 분석 결과

2-1) A Tale of Two Cities 분석 결과 (세부내용 .ipynb 파일 참조)

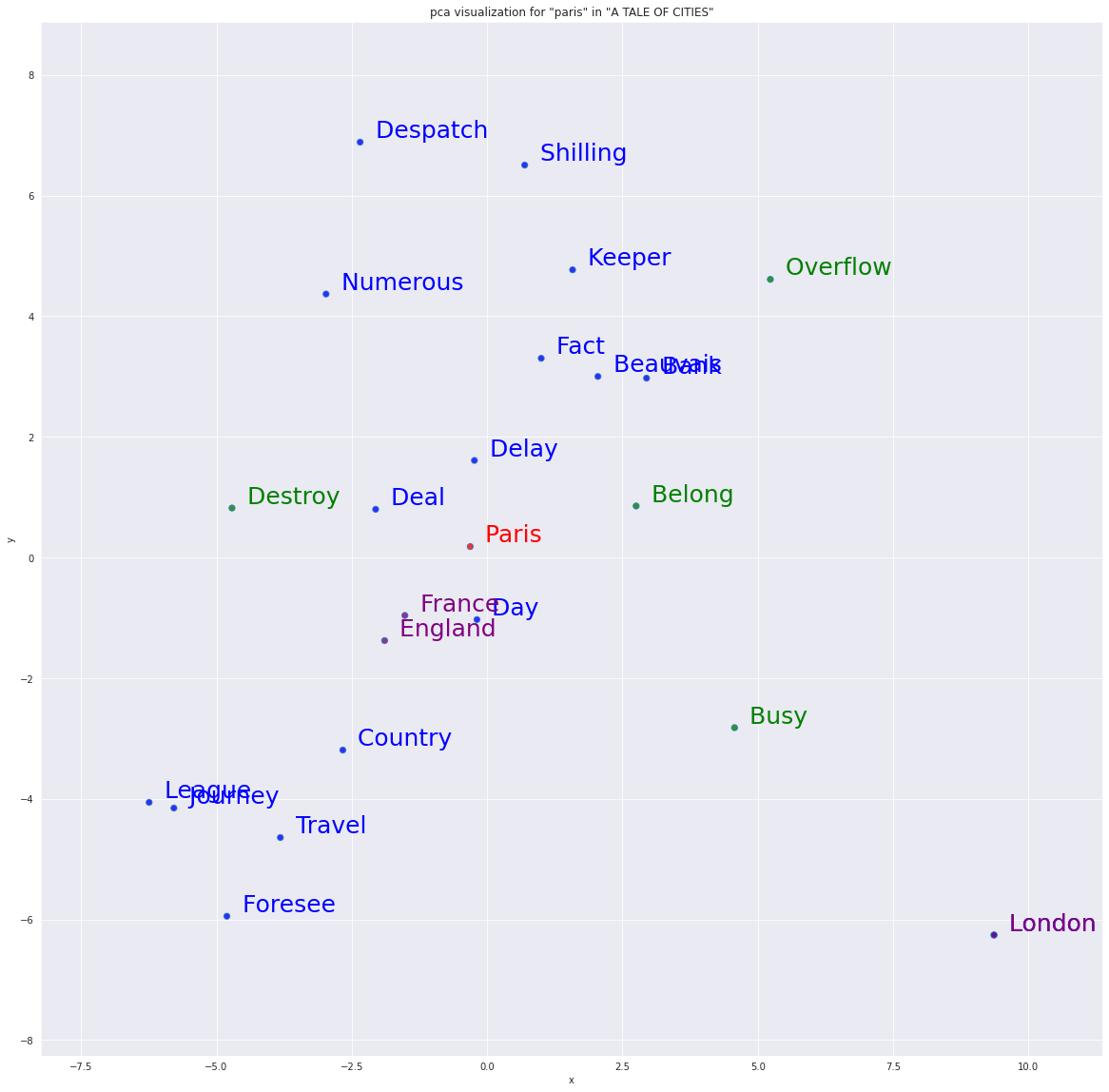
도시 단어: london  
선택된 이웃 단어: ['tellson', 'mail', 'partner', 'busy', 'tellson\_s', 'green', 'arrive', 'blind', 'paris', 'worthy', 'harness', 'circle', 'british', 'lumber', 'relation', 'residence', 'orphan', 'respectable', 'bank', 'dismal']  
이웃단어들 중 감성단어: ['busy', 'green', 'blind', 'worthy', 'lumber', 'respectable', 'dismal']  
이웃단어들의 감성점수: [-0.3125, -0.375, -0.625, 0.6875, -0.875, 0.75, -0.625]

final sentiment score of 'london': -0.19642857142857142



도시 단어: paris  
선택된 이웃 단어: ['journey', 'foresee', 'despatch', 'busy', 'london', 'day', 'travel', 'league', 'deal', 'country', 'beauvais', 'numerous', 'delay', 'shilling', 'overflow', 'keeper', 'fact', 'destroy', 'belong', 'bank']  
이웃단어들 중 감성단어: ['busy', 'overflow', 'destroy', 'belong']  
이웃단어들의 감섬점수: [-0.3125, -0.5, -0.5, 0.75]

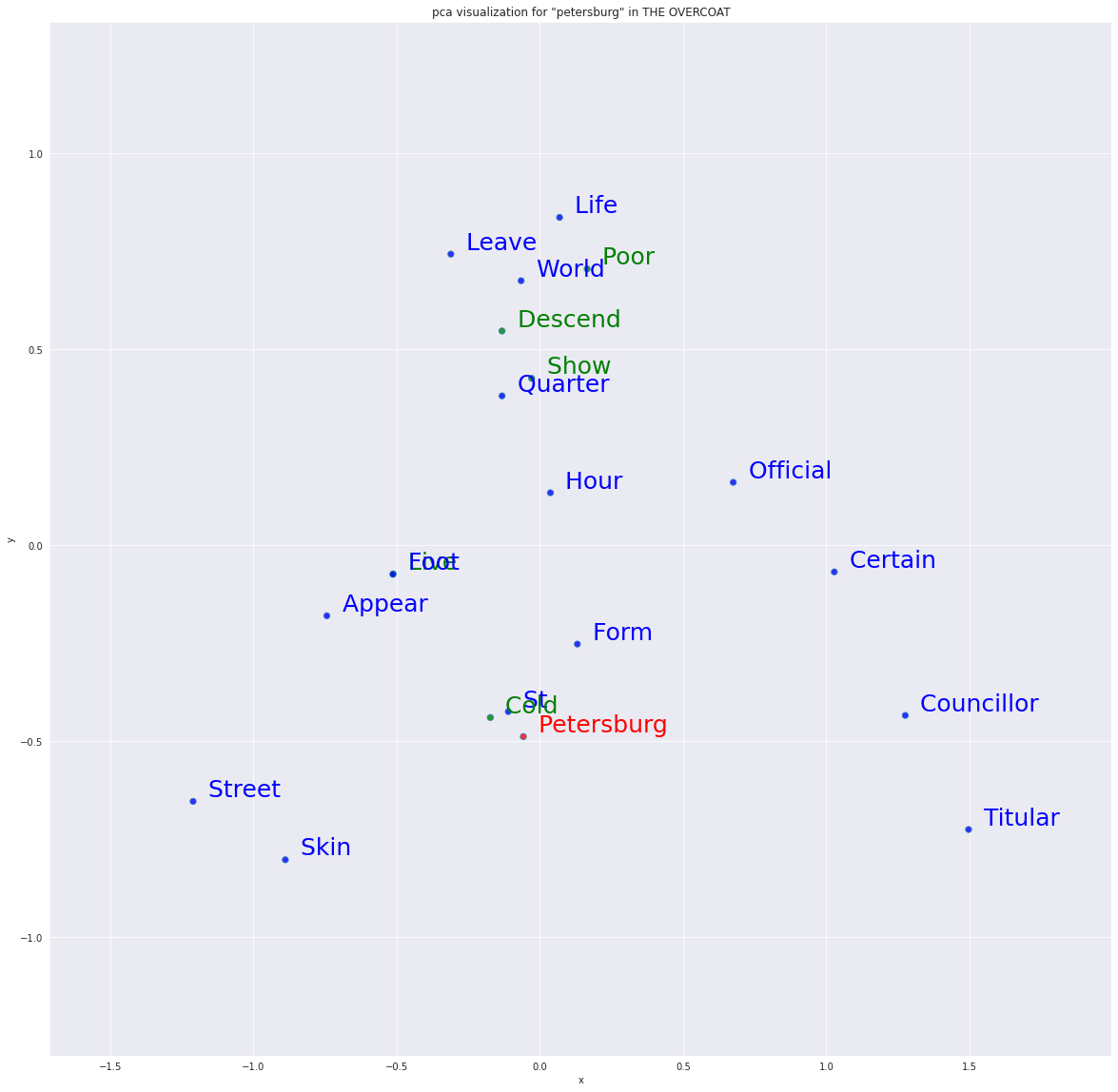
final sentiment score of 'paris': -0.140625



2-2) The Overcoat 분석 결과 (세부내용 .ipynb 파일 참조)

도시 단어: petersburg  
선택된 이웃 단어: ['st', 'official', 'form', 'street', 'live', 'descend', 'hour', 'appear', 'cold', 'skin', 'poor', 'life', 'certain', 'leave', 'foot', 'quarter', 'titular', 'councillor', 'world', 'show']  
이웃단어들 중 감성단어: ['live', 'descend', 'cold', 'poor', 'show']  
이웃단어들의 감성점수: [0.5416666666666666, 0.5, -0.375, -0.6875, 0.5]

final sentiment score of 'petersburg': 0.0958333333333333



2-3) Father Goriot 분석 결과 (세부내용 .ipynb 파일 참조)

도시 단어: paris  
선택된 이웃 단어: ['salon', 'house', 'meet', 'fashionable', 'sensation', 'fashion', 'willing', 'gain', 'remote', 'prompt', 'charming', 'queen', 'high', 'wealthy', 'dust', 'fancy', 'beauseant', 'enjoy', 'present', 'respectable']  
이웃단어들 중 감성단어: ['meet', 'fashionable', 'fashion', 'willing', 'gain', 'charming', 'high', 'fancy', 'enjoy', 'respectable']  
이웃단어들의 감성점수: [0.625, 0.375, 0.5, -0.0625, 0.875, 0.6875, 0.375, -0.0625, 0.6666666666666666, 0.75]

final sentiment score of 'paris': 0.4729166666666666



1. 활용방법

* 본 모델은 학습이나 분석에 필요한 하이퍼파라미터를 설정하는 방식에 따라 결과가 달라질 수 있음
* 전체 분석 프로세스를 실행한 뒤, query, num\_k, object\_threshold의 값을 변경하면 다른 분석 결과를 얻을 수 있음
* 예) father goriot 분석에서 query는 paris로 두고, num\_k를 100개, obj\_trhreshold를 0.5로 설정한 결과

텍스트이(가) 표시된 사진

자동 생성된 설명

