[TA 2-4]

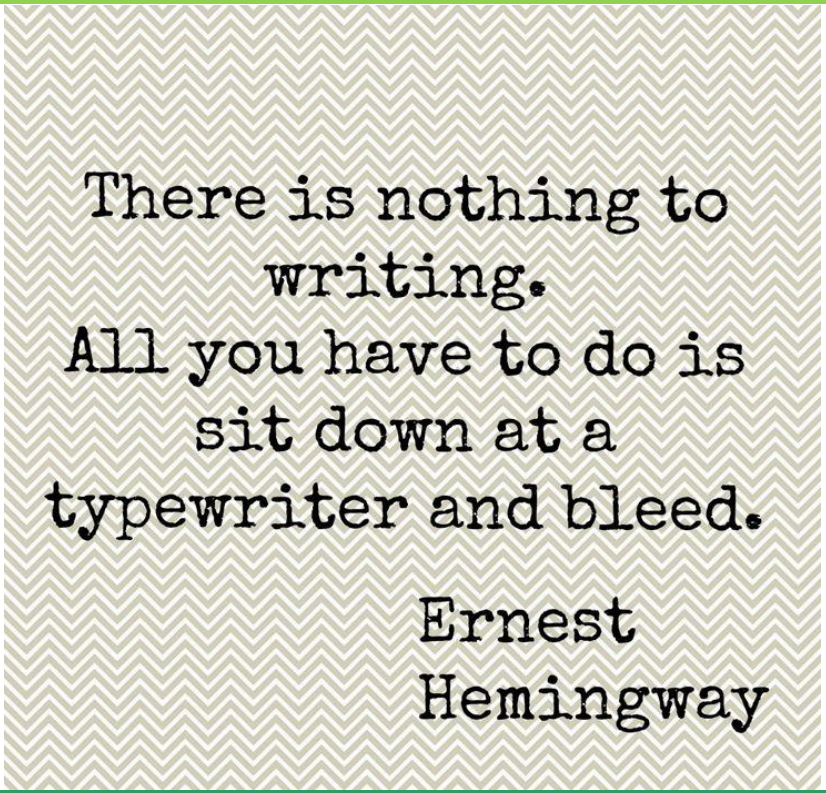
Text Mining - Basics

빅데이터 프로그래밍 기초

박진수 교수

Big Data Institute,
Seoul National University

실습 1 – 영어 텍스트 전처리하기



실습 2-4-1-1. 영어 문장 토큰화하기

· 파이썬의 nltk 모듈을 활용하여, 아래 문장들을 토큰화(Tokenization)한 결과를 출력한다

- Sentence 1: “My only regret in life is that I did not drink more wine.”

- Sentence 2: “I drink to make other people more interesting.”

- Sentence 3: “An intelligent man is sometimes forced to be drunk to spend time with his fools.”

· TIPS

- nltk 패키지의 word_tokenize 함수를 사용한다

실습 2-4-1-1. 영어 문장 토큰화하기

· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 1-1-1.py
Tokens of Sentence 1:
['My', 'only', 'regret', 'in', 'life', 'is', 'that', 'I', 'did', 'not', 'drink',
 'more', 'wine', '.']
Tokens of Sentence 2:
['I', 'drink', 'to', 'make', 'other', 'people', 'more', 'interesting', '.']
Tokens of Sentence 3:
['An', 'intelligent', 'man', 'is', 'sometimes', 'forced', 'to', 'be', 'drunk', '
to', 'spend', 'time', 'with', 'his', 'fools', '.']

C:₩Users₩USER₩seoul>
```

실습 2-4-1-2. 영어 문장 품사 태깅(POS tagging)하기

· 아래 문장들의 품사를 태깅해 출력한다

- Sentence 1: “My only regret in life is that I did not drink more wine.”

- Sentence 2: “I drink to make other people more interesting.”

- Sentence 3: “An intelligent man is sometimes forced to be drunk to spend time with his fools.”

· TIPS

- nltk 패키지의 pos_tag 함수를 사용한다

- 토큰화 된 결과를 활용한다

실습 2-4-1-2. 영어 문장 품사 태깅(POS tagging)하기

· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 1-1-2.py
POS tagging Sentence 1:
[('My', 'PRP$'), ('only', 'JJ'), ('regret', 'NN'), ('in', 'IN'), ('life', 'NN'),
 ('is', 'VBZ'), ('that', 'IN'), ('I', 'PRP'), ('did', 'VBD'), ('not', 'RB'), ('d
rink', 'VB'), ('more', 'JJR'), ('wine', 'NN'), ('.', '.')]
POS tagging Sentence 2:
[('I', 'PRP'), ('drink', 'VBP'), ('to', 'TO'), ('make', 'VB'), ('other', 'JJ'),
('people', 'NNS'), ('more', 'RBR'), ('interesting', 'JJ'), ('.', '.')]
POS tagging Sentence 3:
[('An', 'DT'), ('intelligent', 'JJ'), ('man', 'NN'), ('is', 'VBZ'), ('sometimes'
, 'RB'), ('forced', 'VBN'), ('to', 'TO'), ('be', 'VB'), ('drunk', 'VBN'), ('to',
 'TO'), ('spend', 'VB'), ('time', 'NN'), ('with', 'IN'), ('his', 'PRP$'), ('fool
s', 'NNS'), ('.', '.')]

C:₩Users₩USER₩seoul>
```

실습 2-4-1-2. 영어 문장 품사 태깅(POS tagging)하기

· 참고: nltk의 POS 태그 리스트 (source: https://pythonprogramming.net/part-of-speech-tagging-nltk-tutorial/)

태그	품사	번역
CC	Coordinating conjunction	등위접속사
CD	Cardinal digit	기수
DT	Determiner	한정사
EX	Existential	There (is), there (exists)
FW	Foreign word	외래어
IN	Preposition/subordinating conjunction	전치사/종속 접속사
JJ	Adjective	형용사
JJR	Adjective (comparative)	형용사(비교급)
JJS	Adjective (superlative)	형용사(최상급)
LS	List marker	리스트 마커(1), 2), 3) 등)
MD	Modal	조동사
NN	Noun (singular)	명사(단수)
NNS	Noun (plural)	명사(복수)
NNP	Proper noun (singular)	고유 명사(단수)
NNPS	Proper noun (plural)	고유 명사(복수)
PDT	Predeterminer	전치 한정사
POS	Possessive ending	소유격 ‘s

실습 2-4-1-2. 영어 문장 품사 태깅(POS tagging)하기

· 참고: nltk의 POS 태그 리스트 – 계속 (source: https://pythonprogramming.net/part-of-speech-tagging-nltk-tutorial/)

태그	품사	번역
PRP	Personal noun	인칭대명사
PRP$	Possessive pronoun	소유격
RB	Adverb	부사
RBR	Adverb (comparative)	부사(비교급)
RBS	Adverb (superlative)	부사(최상급)
RB	Particle	불변화사
TO	To	To부정사
UH	Interjection	감탄사
VB	Verb (Base form)	동사(기본형)
VBD	Verb (Past tense)	동사(과거형)
VBN	Verb (Past participle)	동사(과거 분사)
VBP	Verb (Singular, present)	동사(현재 단수형)
VBZ	Verb (3rd person, singular, present)	동사(3인칭 현재 단수형)
WDT	Wh-determiner	한정사(which)
WP	Wh-pronoun	관계대명사(who, what)
WRB	Wh-adverb	관계부사(where, when)

실습 2-4-1-3. 단어의 기본형 찾기(Lemmatization)

- Lemmatization: 단어의 기본형 찾기(source: https://en.wikipedia.org/wiki/Lemmatisation#Description)
  - 한국어와는 달리 영어는 단어의 기본형을 찾기가 상대적으로 용이하다
  - Lemmatization은 단어의 어근에 기반하여 단어의 기본형(lemma)를 찾아준다
  - 예시
    - better → good
    - cactus → cacti
    - apples → apple

실습 2-4-1-3. 단어의 기본형 찾기(Lemmatization)

· 아래 세 문장을 lemmatize해 그 결과를 출력해 본다

- Sentence 1: “My only regret in life is that I did not drink more wine.”

- Sentence 2: “I drink to make other people more interesting.”

- Sentence 3: “An intelligent man is sometimes forced to be drunk to spend time with his fools.”

· TIPS

- nltk 패키지의 wordnet.WordNetLemmatizer 함수를 활용한다

실습 2-4-1-3. 단어의 기본형 찾기(Lemmatization)

· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 1-1-3.py
Lemmatizing Sentence 1:
['My', 'only', 'regret', 'in', 'life', 'is', 'that', 'I', 'did', 'not', 'drink',
 'more', 'wine', '.']
Lemmatizing Sentence 2:
['I', 'drink', 'to', 'make', 'other', 'people', 'more', 'interesting', '.']
Lemmatizing Sentence 3:
['An', 'intelligent', 'man', 'is', 'sometimes', 'forced', 'to', 'be', 'drunk', '
to', 'spend', 'time', 'with', 'his', 'fool', '.']

C:₩Users₩USER₩seoul>
```

실습 2-4-1-4. Stopwords 제거하기

· Stopwords: 문맥상 유의하지 않아 전처리 과정에서 흔히 분석의 효율성을 위해 필터링하는 단어들

- 대표적인 영어의 stopwords로는 the, is, at, which, on 등이 있음

- nltk 패키지의 stopwords로는 다음을 참고한다: http://www.nltk.org/book/ch02.html

실습 2-4-1-4. Stopwords 제거하기

· 아래 세 문장에서 stopwords를 제거한 결과를 출력해 본다

- Sentence 1: “My only regret in life is that I did not drink more wine.”

- Sentence 2: “I drink to make other people more interesting.”

- Sentence 3: “An intelligent man is sometimes forced to be drunk to spend time with his fools.”

· TIPS

- nltk.corpus의 stopwords를 불러온다

- nltk 패키지의 stopwords list는 모두 소문자로 이루어져 있음을 유의한다

실습 2-4-1-4. Stopwords 제거하기

· 출력 결과

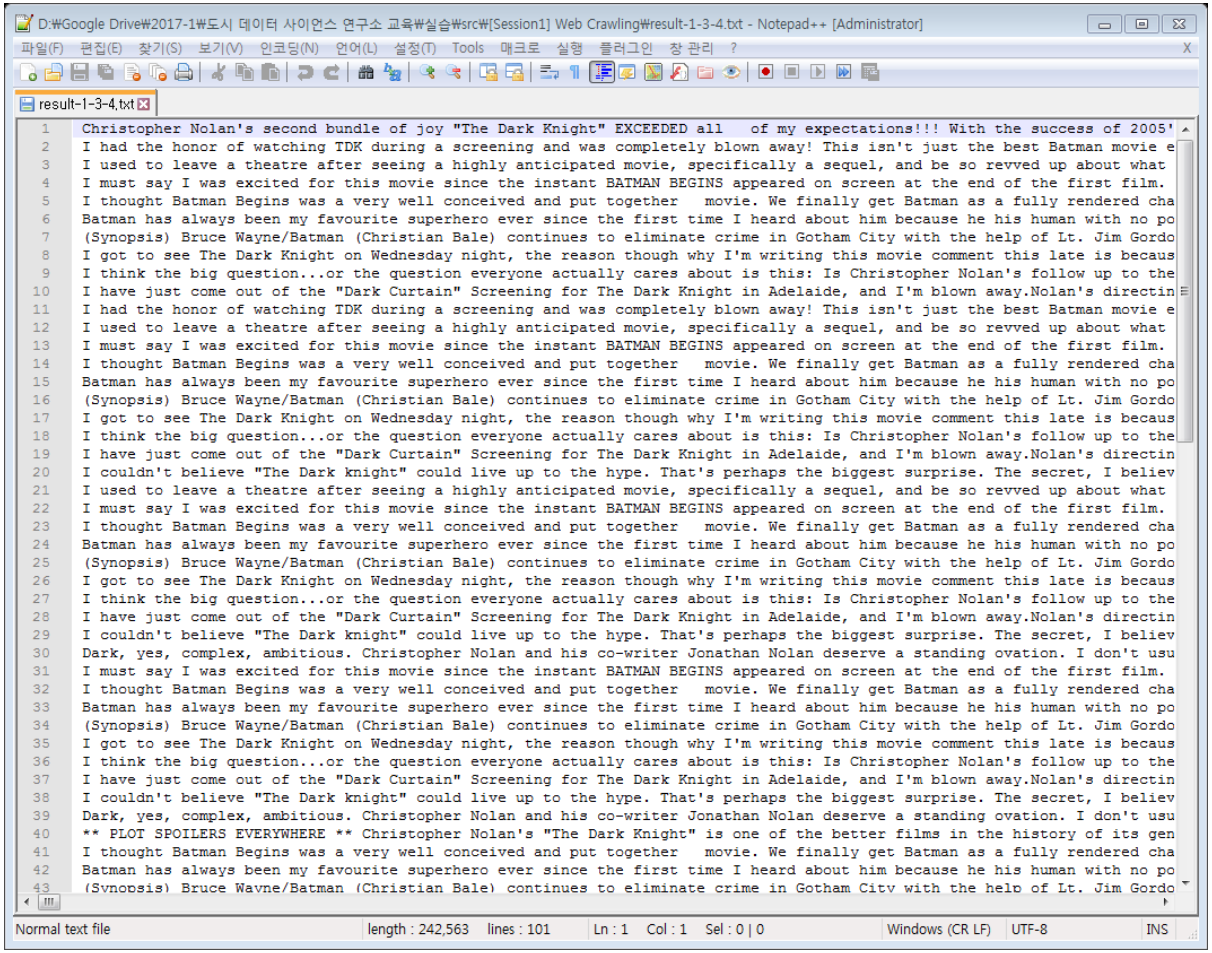
관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 1-1-4.py
Removing stopwords in Sentence 1:
['regret', 'life', 'drink', 'wine', '.']
Removing stopwords in Sentence 2:
['drink', 'make', 'people', 'interesting', '.']
Removing stopwords in Sentence 3:
['intelligent', 'man', 'sometimes', 'forced', 'drunk', 'spend', 'time', 'fools',
 '.']

C:₩Users₩USER₩seoul>
```

실습 2 – IMDb 리뷰 데이터 전처리하기

실습 2-4-2-1. 첫 번째 리뷰 전처리하기 (1)

· 웹 크롤링 실습에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰를 불러와 그 중 첫 번째 리뷰 텍스트를 토큰화하고 품사 태깅을 해 그 결과를 출력한다



실습 2-4-2-1. 첫 번째 리뷰 전처리하기 (1)

· TIPS

- readlines() 함수를 활용해 리뷰 데이터를 리스트로 받아온다

- 파일을 열 때 인코딩 설정을 꼭 ‘utf-8’으로 한다

실습 2-4-2-1. 첫 번째 리뷰 전처리하기 (1)

· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 2-2-1.py
Tokens for first review:
['Christopher', 'Nolan', "'s", 'second', 'bundle', 'of', 'joy', '``', 'The', 'Da
rk', 'Knight', "''", 'EXCEEDED', 'all', 'of', 'my', 'expectations', '!', '!', '!
', 'With', 'the', 'success', 'of', '2005', "'s", 'reboot', 'of', 'the', 'Batman'
, 'franchise', ',', 'they', 'took', 'what', 'was', 'already', 'established', 'an
d', 'expanded', 'it', ',', 'amped', 'it', 'up', ',', 'and', 'gave', 'a', 'deeper
', ',', 'darker', 'and', 'brooding', 'story', 'that', 'is', 'more', 'gripping',
'and', 'the', 'suspense', 'is', 'likely', 'to', 'catch', 'you', 'of', 'guard', '
several', 'times', 'throughout', '.', 'Christian', 'Bale', 'delves', 'more', 'de
eper', 'into', 'Batman', ',', 'sworn', 'to', 'fight', 'evil', 'and', 'injustice'
, ',', 'though', 'also', 'quite', 'reluctant', 'and', 'uncertain', 'if', 'his',
'crusade', 'can', 'ever', 'end', 'and', 'cleanse', 'his', 'inner', 'turmoil', 'f
rom', 'his', 'fractured', 'soul', 'due', 'to', 'the', 'murder', 'of', 'his', 'be
loved', 'parents', '.', 'But', 'with', 'the', 'help', 'of', 'his', 'trusted', 'b
utler/ally', 'Alfred', '(', 'played', 'superbly', 'by', 'Michael', 'Cane', ')',
'grounds', 'him', ',', 'gives', 'him', 'moral', 'support', ',', 'and', 'keeps',
'him', 'in', 'check', '.', 'But', 'the', 'real', 'star', 'of', 'the', 'show', 'i
s', 'Heath', 'Ledger', 'as', 'Batman', "'s", 'most', 'deadly', 'enemy', ',', 'Th
e', 'Joker', '.', 'I', 'can', 'HONESTLY', 'tell', 'you', 'that', ':', 'as', 'goo
d', 'as', 'Jack', 'Nicholson', 'was', 'in', "Batman'89", 'he', 'is', 'CHILD', "'
S", 'PLAY', 'compared', 'to', 'this', 'Joker', '.', 'He', 'is', 'sadistic', ',',
 'psychotic', ',', 'and', 'downright', 'SCARIER', 'and', 'PSYCHOLOGICALLY', 'dis
turbing', 'than', 'the', 'previous', 'incarnation', 'of', 'The', 'Clown', 'Princ
e', 'of', 'Crime', 'and', 'Ledger', 'gives', 'it', 'his', 'all', 'to', 'do', 'hi
m', 'justice', '.', 'Along', 'with', 'the', 'original', 'cast', 'comes', 'some',
 'fresh', 'faces', 'such', 'as', 'Aaron', 'Eckhart', ',', 'Maggie', 'Gyllenhaal'
, 'and', 'more', '.', 'I', 'must', 'say', 'though', 'I', 'liked', 'Katie', 'Holm
es', ',', 'Gyllenhaal', 'gives', 'a', 'much', 'better', 'performance', 'and', 'i
s', 'a', 'far', 'cry', 'from', 'the', '``', 'damsel-in-distress', "''", 'stereot
ype', '(', 'though', 'there', "'s", 'a', 'little', 'of', 'it', ',', 'THANKFULLY'
, ')', 'that', "'s", 'common', 'in', 'films', '.', 'Bale', 'and', 'Gyllenhall',
'have', 'MUCH', 'better', 'chemistry', 'this', 'time', 'around', 'more', 'so', '
than', 'Holmes', '.', 'Even', 'better', ',', 'the', 'fight', 'sequences', 'are',
 'vastly', 'improved', 'and', 'feature', 'more', 'brutal', 'and', 'bone', 'crush
ing', 'combat', 'than', '``', 'Begins', "''", 'in', 'addition', 'to', 'new', 'te
chnology', 'at', 'Batman', "'s", 'disposal', '.', 'Also', 'worth', 'mentioning',
 'is', 'screenwriter', 'Jonathan', 'Nolan', ',', 'who', 'gives', 'the', 'film',
```

<토큰>

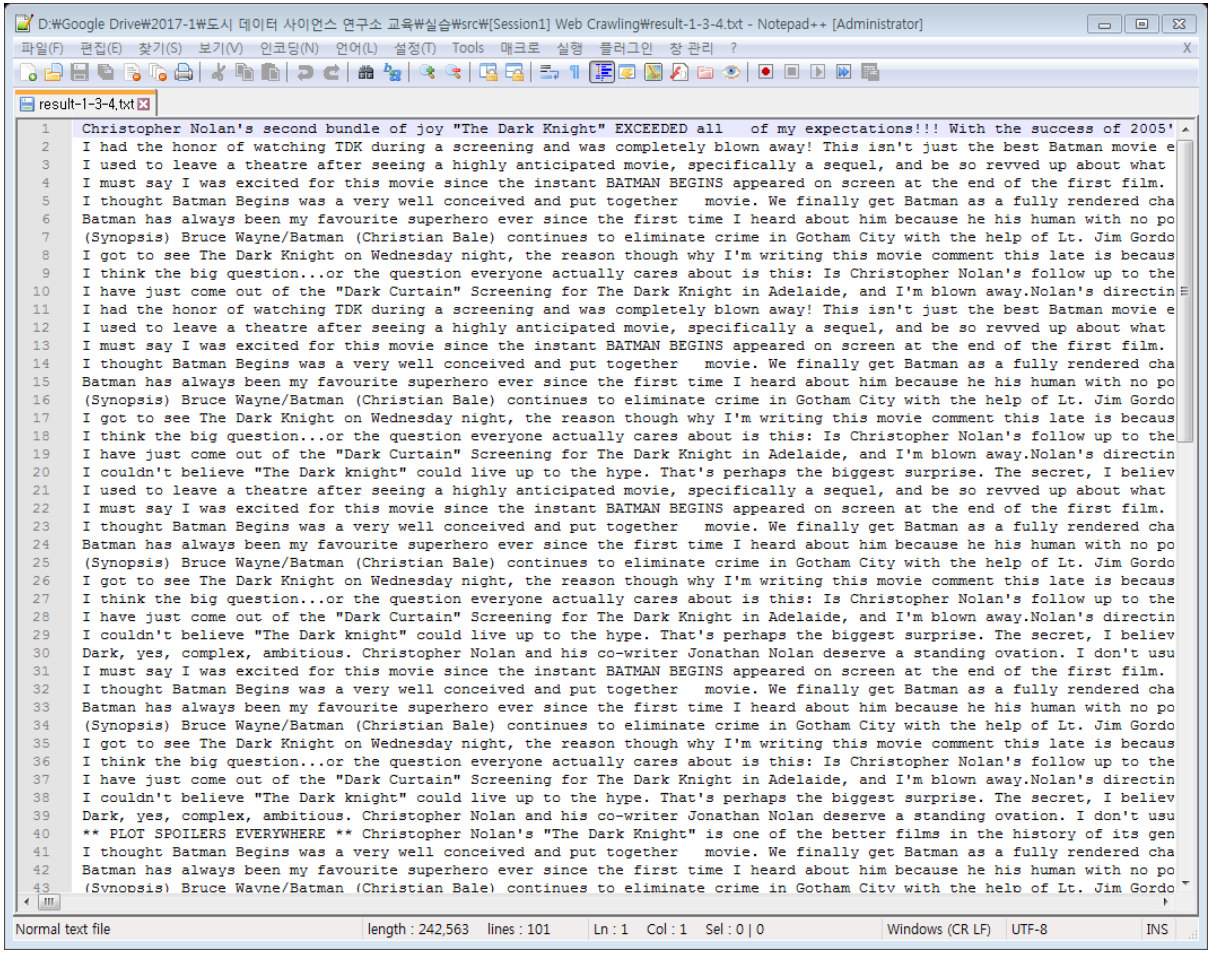
관리자: C:₩Windows₩system32₩cmd.exe

```
POS tags for first review:
[('Christopher', 'NNP'), ('Nolan', 'NNP'), ("'s", 'POS'), ('second', 'JJ'), ('bu
ndle', 'NN'), ('of', 'IN'), ('joy', 'NN'), ('``', '``'), ('The', 'DT'), ('Dark',
 'NNP'), ('Knight', 'NNP'), ("''", "''"), ('EXCEEDED', 'NNP'), ('all', 'DT'), ('
of', 'IN'), ('my', 'PRP$'), ('expectations', 'NNS'), ('!', '.'), ('!', '.'), ('!
', '.'), ('With', 'IN'), ('the', 'DT'), ('success', 'NN'), ('of', 'IN'), ('2005'
, 'CD'), ("'s", 'POS'), ('reboot', 'NN'), ('of', 'IN'), ('the', 'DT'), ('Batman'
, 'NNP'), ('franchise', 'NN'), (',', ','), ('they', 'PRP'), ('took', 'VBD'), ('w
hat', 'WP'), ('was', 'VBD'), ('already', 'RB'), ('established', 'VBN'), ('and',
'CC'), ('expanded', 'VBN'), ('it', 'PRP'), (',', ','), ('amped', 'VBD'), ('it',
'PRP'), ('up', 'RP'), (',', ','), ('and', 'CC'), ('gave', 'VBD'), ('a', 'DT'), (
'deeper', 'NN'), (',', ','), ('darker', 'NN'), ('and', 'CC'), ('brooding', 'VBG'
), ('story', 'NN'), ('that', 'WDT'), ('is', 'VBZ'), ('more', 'JJR'), ('gripping'
, 'JJ'), ('and', 'CC'), ('the', 'DT'), ('suspense', 'NN'), ('is', 'VBZ'), ('like
ly', 'JJ'), ('to', 'TO'), ('catch', 'VB'), ('you', 'PRP'), ('of', 'IN'), ('guard
', 'JJ'), ('several', 'JJ'), ('times', 'NNS'), ('throughout', 'IN'), ('.', '.'),
 ('Christian', 'JJ'), ('Bale', 'NNP'), ('delves', 'VBZ'), ('more', 'JJR'), ('dee
per', 'JJ'), ('into', 'IN'), ('Batman', 'NNP'), (',', ','), ('sworn', 'VBN'), ('
to', 'TO'), ('fight', 'VB'), ('evil', 'NN'), ('and', 'CC'), ('injustice', 'NN'),
 (',', ','), ('though', 'RB'), ('also', 'RB'), ('quite', 'RB'), ('reluctant', 'J
J'), ('and', 'CC'), ('uncertain', 'JJ'), ('if', 'IN'), ('his', 'PRP$'), ('crusad
e', 'NN'), ('can', 'MD'), ('ever', 'RB'), ('end', 'VB'), ('and', 'CC'), ('cleans
e', 'VB'), ('his', 'PRP$'), ('inner', 'JJ'), ('turmoil', 'NN'), ('from', 'IN'),
('his', 'PRP$'), ('fractured', 'JJ'), ('soul', 'NN'), ('due', 'JJ'), ('to', 'TO'
), ('the', 'DT'), ('murder', 'NN'), ('of', 'IN'), ('his', 'PRP$'), ('beloved', '
JJ'), ('parents', 'NNS'), ('.', '.'), ('But', 'CC'), ('with', 'IN'), ('the', 'DT
'), ('help', 'NN'), ('of', 'IN'), ('his', 'PRP$'), ('trusted', 'VBN'), ('butler/
ally', 'RB'), ('Alfred', 'NNP'), ('(', '('), ('played', 'VBN'), ('superbly', 'RB
'), ('by', 'IN'), ('Michael', 'NNP'), ('Cane', 'NNP'), (')', ')'), ('grounds', '
VBZ'), ('him', 'PRP'), (',', ','), ('gives', 'VBZ'), ('him', 'PRP'), ('moral', '
JJ'), ('support', 'NN'), (',', ','), ('and', 'CC'), ('keeps', 'VBZ'), ('him', 'P
RP'), ('in', 'IN'), ('check', 'NN'), ('.', '.'), ('But', 'CC'), ('the', 'DT'), (
'real', 'JJ'), ('star', 'NN'), ('of', 'IN'), ('the', 'DT'), ('show', 'NN'), ('is
', 'VBZ'), ('Heath', 'NNP'), ('Ledger', 'NNP'), ('as', 'IN'), ('Batman', 'NNP'),
 ("'s", 'POS'), ('most', 'JJS'), ('deadly', 'RB'), ('enemy', 'NN'), (',', ','),
('The', 'DT'), ('Joker', 'NNP'), ('.', '.'), ('I', 'PRP'), ('can', 'MD'), ('HONE
STLY', 'VB'), ('tell', 'VB'), ('you', 'PRP'), ('that', 'WDT'), (':', ':'), ('as'
, 'RB'), ('good', 'JJ'), ('as', 'IN'), ('Jack', 'NNP'), ('Nicholson', 'NNP'), ('
```

<품사 태그>

실습 2-4-2-2. 첫 번째 리뷰 전처리하기 (2)

· 웹 크롤링 실습에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰를 불러와 그 중 첫 번째 리뷰 텍스트에서 stopwords를 제거하고 lemmtization을 수행해 그 결과를 출력한다



실습 2-4-2-2. 첫 번째 리뷰 전처리하기 (2)

· TIPS

- readlines() 함수를 활용해 리뷰 데이터를 리스트로 받아온다

- 파일을 열 때 인코딩 설정을 꼭 ‘utf-8’으로 한다

실습 2-4-2-2. 첫 번째 리뷰 전처리하기 (2)

· 출력 결과

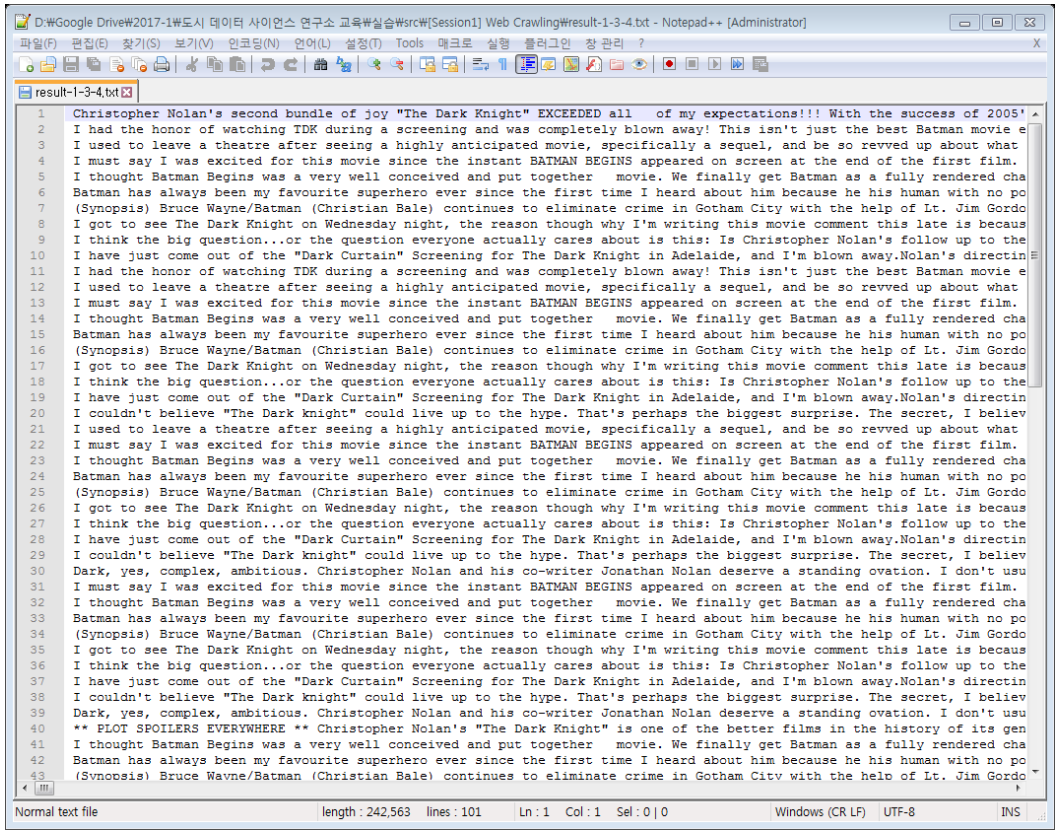
관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 2-2-2.py
Lemmas of the first review:
['Christopher', 'Nolan', "'s", 'second', 'bundle', 'joy', '``', 'Dark', 'Knight'
, "''", 'EXCEEDED', 'expectation', '!', '!', '!', 'success', '2005', "'s", 'rebo
ot', 'Batman', 'franchise', ',', 'took', 'already', 'established', 'expanded', '
,', 'amped', ',', 'gave', 'deeper', ',', 'darker', 'brooding', 'story', 'grippin
g', 'suspense', 'likely', 'catch', 'guard', 'several', 'time', 'throughout', '.'
, 'Christian', 'Bale', 'delf', 'deeper', 'Batman', ',', 'sworn', 'fight', 'evil'
, 'injustice', ',', 'though', 'also', 'quite', 'reluctant', 'uncertain', 'crusad
e', 'ever', 'end', 'cleanse', 'inner', 'turmoil', 'fractured', 'soul', 'due', 'm
urder', 'beloved', 'parent', '.', 'help', 'trusted', 'butler/ally', 'Alfred', '(
', 'played', 'superbly', 'Michael', 'Cane', ')', 'ground', ',', 'give', 'moral',
 'support', ',', 'keep', 'check', '.', 'real', 'star', 'show', 'Heath', 'Ledger'
, 'Batman', "'s", 'deadly', 'enemy', ',', 'Joker', '.', 'HONESTLY', 'tell', ':',
 'good', 'Jack', 'Nicholson', "Batman'89", 'CHILD', "'S", 'PLAY', 'compared', 'J
oker', '.', 'sadistic', ',', 'psychotic', ',', 'downright', 'SCARIER', 'PSYCHOLO
GICALLY', 'disturbing', 'previous', 'incarnation', 'Clown', 'Prince', 'Crime', '
Ledger', 'give', 'justice', '.', 'Along', 'original', 'cast', 'come', 'fresh', '
face', 'Aaron', 'Eckhart', ',', 'Maggie', 'Gyllenhaal', '.', 'must', 'say', 'tho
ugh', 'liked', 'Katie', 'Holmes', ',', 'Gyllenhaal', 'give', 'much', 'better', '
performance', 'far', 'cry', '``', 'damsel-in-distress', "''", 'stereotype', '(',
 'though', "'s", 'little', ',', 'THANKFULLY', ')', "'s", 'common', 'film', '.',
'Bale', 'Gyllenhall', 'MUCH', 'better', 'chemistry', 'time', 'around', 'Holmes',
 '.', 'Even', 'better', ',', 'fight', 'sequence', 'vastly', 'improved', 'feature
', 'brutal', 'bone', 'crushing', 'combat', '``', 'Begins', "''", 'addition', 'ne
w', 'technology', 'Batman', "'s", 'disposal', '.', 'Also', 'worth', 'mentioning'
, 'screenwriter', 'Jonathan', 'Nolan', ',', 'give', 'film', 'added', 'frosting',
 'already', 'delicious', 'cake', '.', 'Simply', 'put', ',', 'Dark', 'Knight', 't
otally', 'bad', 'as', '``', 'Begins', '.', "''", 'action', 'great', ',', 'plot',
 'deeper', 'engrossing', '.', 'applaud', 'Christopher', 'Nolan', ',', 'Christian
', 'Bale', ',', 'especially', 'Heath', 'Ledger', '(', 'sadly', 'passed', 'away',
 'earlier', 'year', ')', 'aboard', 'believing', 'Mr.', 'Nolan', "'s", 'talent',
'second', 'installment', '.', 'Although', 'may', 'feel', 'bit', 'melancholy', 'L
edger', "'s", 'death', ',', 'final', 'note', 'say', 'sincerely', 'heart', ':', '
Remember', 'Heath', 'Ledger', 'honor', 'mind', 'heart', 'performance', ',', 'hum
an', 'father', 'daughter', 'Matilda', 'Ledger', '.', 'May', 'issue', 'best', 'wi
sh', 'family', 'friend', 'daughter', 'year', 'come', '.', 'Remember', '.', '.',
'.', 'Honor', 'role', 'past', 'role', ',', 'incredible', 'individual', 'talented
```

실습 2-4-2-3. 전체 리뷰 데이터 전처리하기(1)

· 웹 크롤링 실습에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰를 불러와 각각의 리뷰(총 100개)를 전처리해 새로운 텍스트 파일에 저장한다

- 토큰화와 stopwords 제거와 lemmatization을 한 후 저장한다



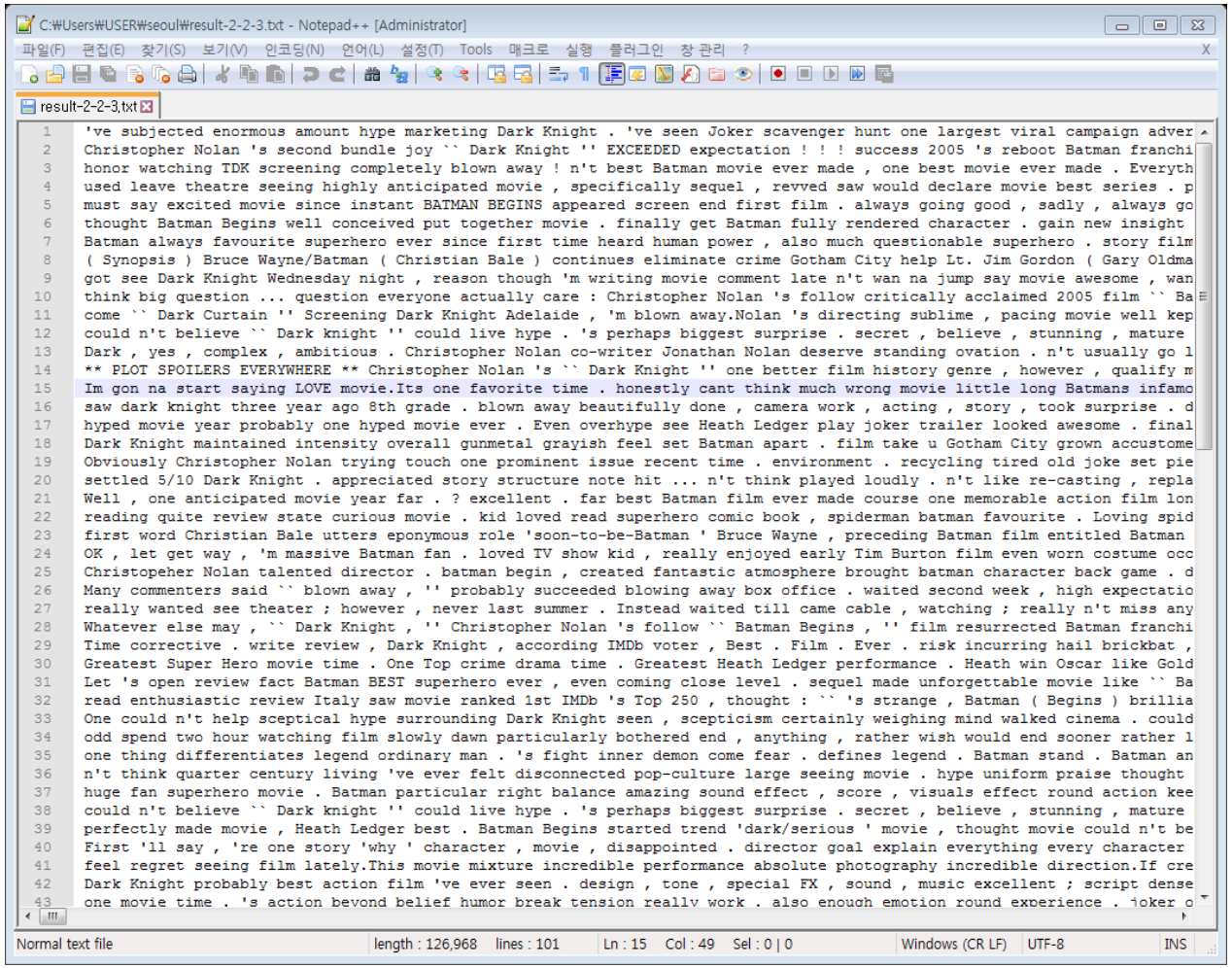
실습 2-4-2-3. 전체 리뷰 데이터 전처리하기(1)

· TIPS

- Input 파일과 같은 100줄 짜리 텍스트 파일이 결과로 나오지만 각각의 줄을 보면 lemmatize 된 결과가 나와야 한다

실습 2-4-2-3. 전체 리뷰 데이터 전처리하기(1)

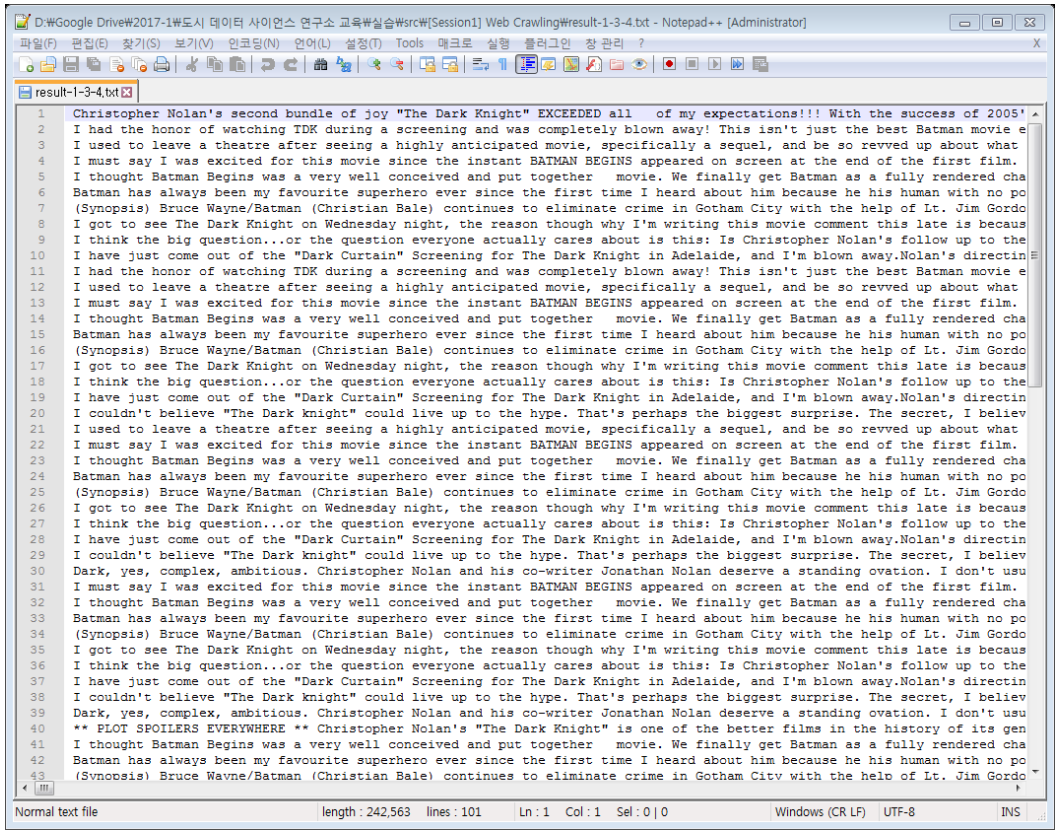
· 저장 결과



실습 2-4-2-4. 전체 리뷰 데이터 전처리하기(2)

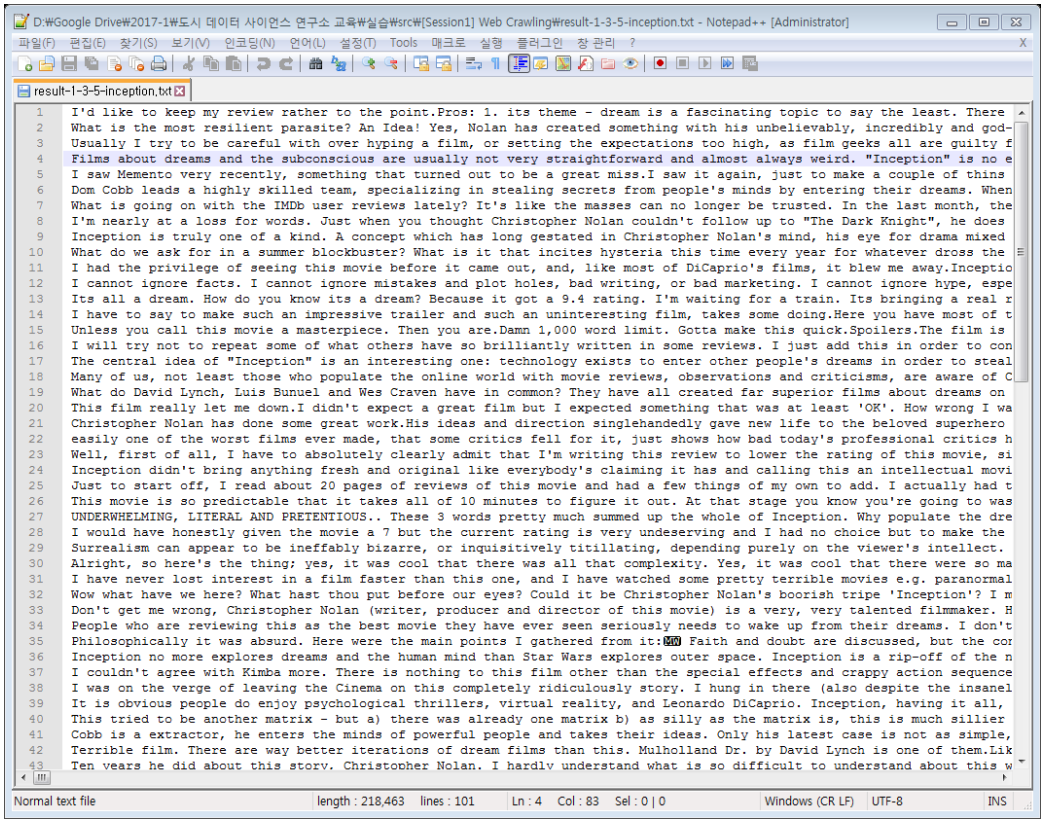
· 웹 크롤링 실습에서 수집하였던 다른 영화 리뷰들을 이전 실습과 같이 전처리해 각각 다른 텍스트 파일에 저장해 보자

- 토큰화와 stopwords 제거와 lemmatization을 한 후 저장한다



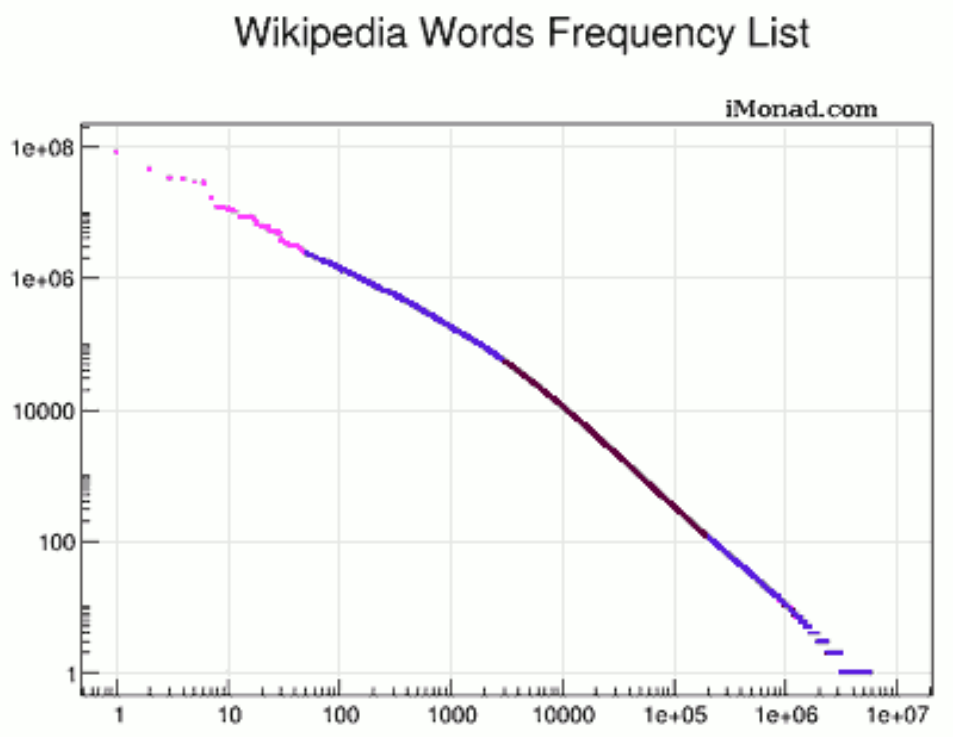
실습 2-4-2-4. 전체 리뷰 데이터 전처리하기(2)

· 저장 결과



result-1-3-5-inception	2017-06-22 오후...	텍스트 문서	214KB
result-1-3-5-old_boy	2017-06-22 오후...	텍스트 문서	155KB
result-1-3-5-whiplash	2017-06-22 오후...	텍스트 문서	173KB

실습 3 – 가장 많이 등장하는 단어 추출하기



실습 2-4-3-1. 리뷰에서 많이 등장하는 명사 추출하기

· 웹 크롤링 실습에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰를 불러와 가장 많이 사용된 명사 10개를 추출해 출력해 본다

· TIPS

- 텍스트를 lower() 함수를 사용해 소문자로 변환 후 비교한다

- 파이썬의 collections 패키지의 Counter 함수를 사용해 많이 사용된 명사를 추출한다

- POS tag 가 **‘NN’, ‘NNS’, ‘NNP, ‘NNPS**’인 것을 추출한다

실습 2-4-3-1. 리뷰에서 많이 등장하는 명사 추출하기

· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-1-1.py
[('movie', 341), ('batman', 298), ('film', 242), ('joker', 217), ('ledger', 130)
, ('knight', 129), ('dark', 126), ('heath', 110), ('time', 97), ('action', 83)]

C:₩Users₩USER₩seoul>
```

실습 2-4-3-1. 리뷰에서 많이 등장하는 명사 추출하기

· 영화 '다크 나이트' 리뷰에서 자주 등장한 명사들

단어	빈도수	단어	빈도수
movie	341	knight	129
batman	298	dark	126
film	242	heath	110
joker	217	time	97
ledger	130	action	83

· 대부분 영화 및 영화 내용과 관련된 단어임을 알 수 있다

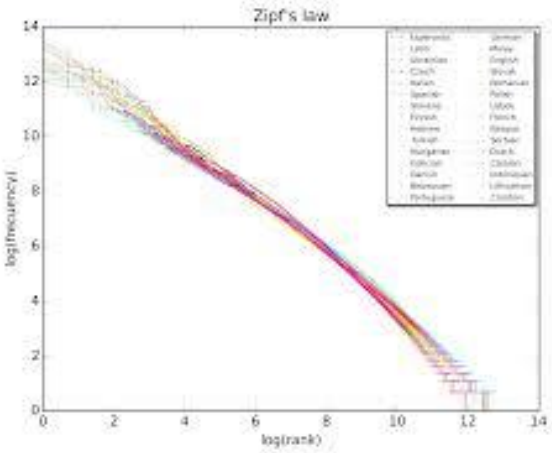
실습 2-4-3-1. 리뷰에서 많이 등장하는 명사 추출하기

· 참고: 지프의 법칙(Zipf's Law) [source: https://en.wikipedia.org/wiki/Zipf%27s_law]

- 20/80의 법칙이라고도 하며 우리가 사용하는 일상어에서 소수의 단어들이 빈도수의 대부분을 차지한다는 내용의 법칙이다

- 예를 들어, 미국 표준 영어에서 "the"가 가장 많이 쓰이며 전체 문서에서 약 7%의 빈도를 보여준다고 한다. 다음으로 많이 사용되는 단어는 "of"와 "and"로 각각 3.5%와 3%의 빈도를 보여준다

· 즉, 지프의 법칙에 따르면 텍스트에서 자주 사용되는 상위 단어 몇 개(즉, 키워드)만 보더라도 텍스트의 내용을 대략적으로 파악할 수 있다



실습 2-4-3-2. 리뷰에서 많이 등장하는 형용사 추출하기

· 웹 크롤링 실습에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰를 불러와 가장 많이 사용된 형용사 10개를 추출해 출력해 본다

· TIPS

- 텍스트를 lower() 함수를 사용해 소문자로 변환 후 비교한다

- 파이썬의 collections 패키지의 Counter 함수를 사용해 많이 사용된 명사를 추출한다

- POS tag 가 **‘JJ’, ‘JJR’, ‘JJS’** 인 것을 추출한다

실습 2-4-3-2. 리뷰에서 많이 등장하는 형용사 추출하기

· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-1-2.py
[('good', 146), ('great', 77), ('many', 53), ('other', 47), ('more', 43), ('much
', 39), ('comic', 39), ('better', 33), ('christian', 29), ('real', 29)]

C:₩Users₩USER₩seoul>
```

실습 2-4-3-2. 리뷰에서 많이 등장하는 형용사 추출하기

· 영화 '다크 나이트' 리뷰에서 자주 등장한 형용사들

단어	빈도수	단어	빈도수
good	146	much	39
great	77	comic	39
many	53	better	33
other	47	christian	29
more	43	real	29

· 형용사는 주로 영화에 대한 감정, 혹은 평가와 관련된 단어들이 많은 것을 알 수 있다

· 29번 등장한 **'Christian':** 주연 배우인 크리스천 배일(Christian Bale)의 이름이라 자주 등장한 것인데 컴퓨터는 이를 형용사로 인식함 => 컴퓨터 인식의 한계이므로 사람의 개입이 필요한 부분

실습 2-4-3-3. 리뷰에서 많이 등장하는 단어 추출하기

· 다른 영화들의 리뷰에서 자주 등장하는 단어를 추출해 본다

- 자신이 관심 있는 품사의 단어들을 자유롭게 추출해 출력해 본다

· TIPS

- 영어 문장 POS 태그를 참고한다

실습 2-4-3-3. 리뷰에서 많이 등장하는 단어 추출하기

· 출력 결과: 영화 ‘인셉션’ 리뷰에서 자주 등장하는 명사들

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-1-3-inception.py
[('movie', 311), ('film', 291), ('dream', 264), ('nolan', 148), ('inception', 14
2), ('plot', 90), ('people', 89), ('cobb', 88), ('dreams', 83), ('time', 80)]

C:₩Users₩USER₩seoul>
```

실습 2-4-3-3. 리뷰에서 많이 등장하는 단어 추출하기

· 출력 결과: 영화 ‘위플래시’ 리뷰에서 자주 등장하는 동사들

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-1-3-whiplash.py
[('is', 609), ('be', 203), ('was', 179), ("'s", 167), ('have', 158), ('are', 154
), ('do', 97), ('has', 82), ('does', 64), ('get', 61)]

C:₩Users₩USER₩seoul>
```

실습 2-4-3-3. 리뷰에서 많이 등장하는 단어 추출하기

· 출력 결과: 영화 ‘올드보이’ 리뷰에서 자주 등장하는 부사들

관리자: C:₩Windows₩system32₩cmd.exe

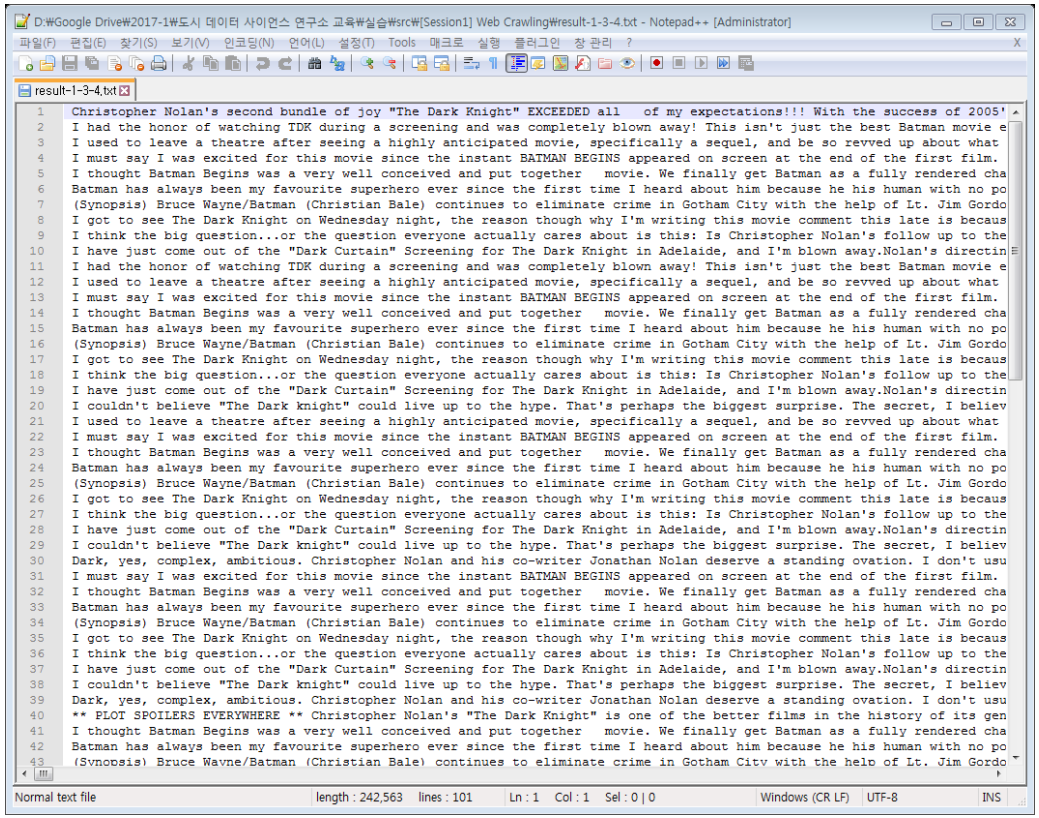
```
C:₩Users₩USER₩seoul>python 3-1-3-old_boy.py
[('not', 237), ("n't", 174), ('so', 109), ('just', 79), ('only', 66), ('even', 6
3), ('also', 50), ('very', 48), ('really', 47), ('then', 46)]

C:₩Users₩USER₩seoul>
```

실습 4 – IMDb 리뷰 데이터 탐색하기

실습 2-4-4-1. 전체 리뷰의 토큰 개수 출력하기

· 웹 크롤링 실습에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰를 불러와 전체 리뷰를 토큰화하고 토큰의 개수를 출력해 본다

- 전체 토큰의 개수와 중복되지 않는(unique) 토큰의 개수를 출력해 보자



실습 2-4-4-1. 전체 리뷰의 토큰 개수 출력하기

· TIPS

- 리뷰를 토큰화할 때 lower 함수를 사용해 소문자로 변환한 뒤 토큰화한다

- Nltk의 Text 함수를 사용한다

  ■ nltk.Text()는 전체 문서(corpus)를 쉽고 간단하게 탐색할 수 있도록 도와준다

- Unique한 토큰의 개수를 출력하기 위해서는 리스트를 셋(set)으로 변환한다

- 분석의 효율성을 위해 Stopwords를 제거한다

실습 2-4-4-1. 전체 리뷰의 토큰 개수 출력하기

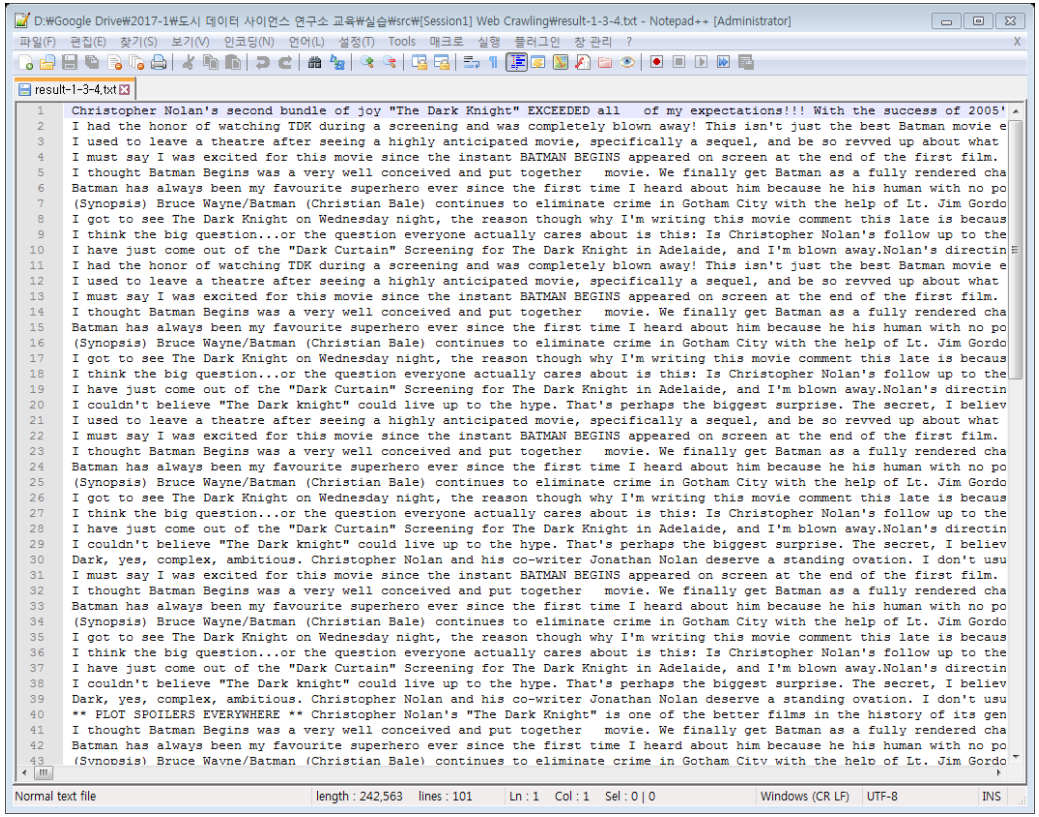
· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-2-1.py
21504 ⟶ 전체 토큰의 개수
4636  ⟶ Unique한 토큰의 개수

C:₩Users₩USER₩seoul>
```

실습 2-4-4-2. 토큰의 등장 횟수 시각화하기

· 웹 크롤링 실습에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰를 불러와 토큰화한 뒤 가장 많이 등장하는 토큰 50개의 등장 횟수를 그래프로 시각화해본다



실습 2-4-4-2. 토큰의 등장 횟수 시각화하기

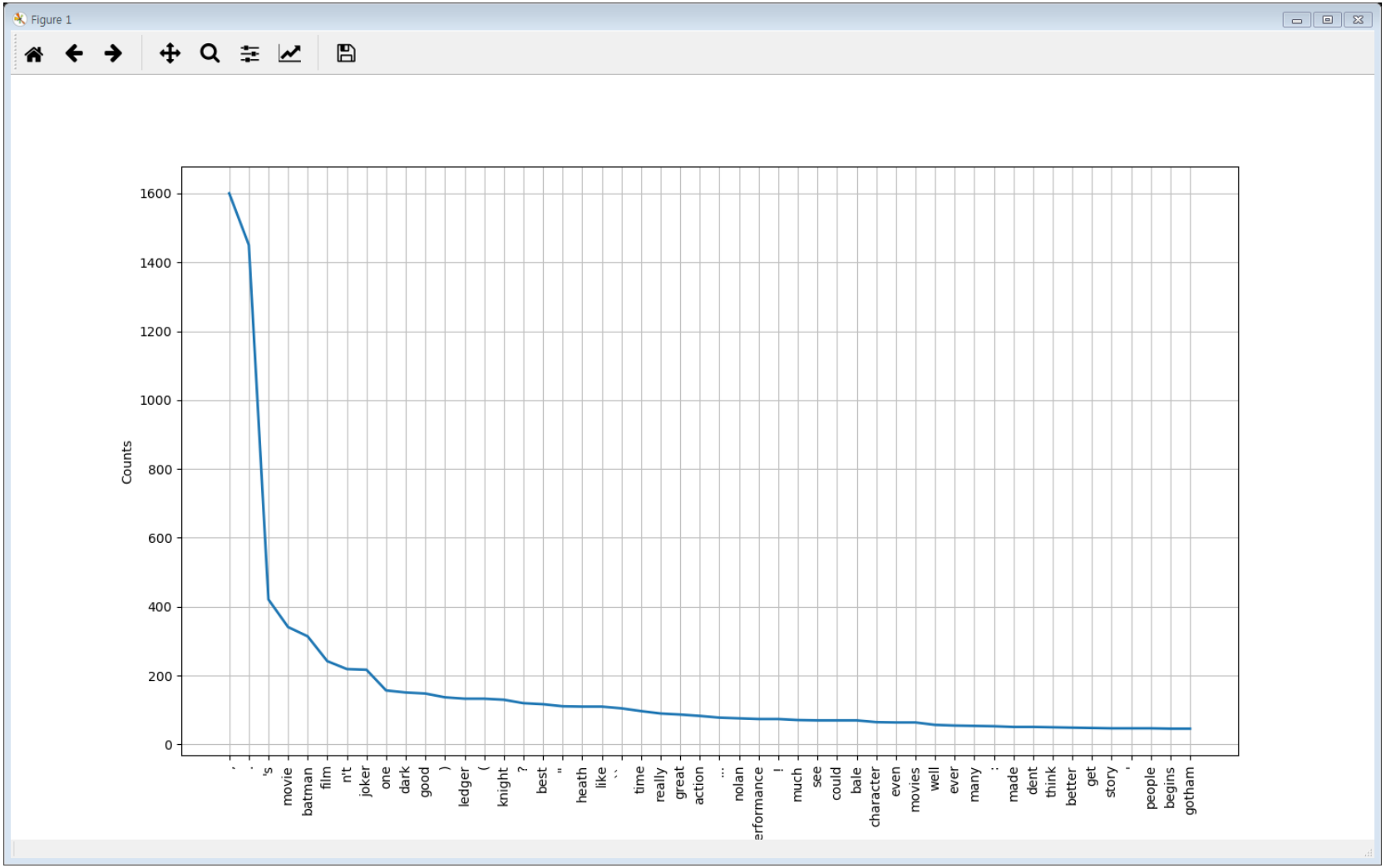
· TIPS

- nltk의 Text 함수를 사용해 토큰화한 결과를 저장한다

- plot 함수를 사용해 그래프를 시각화해본다

실습 2-4-4-2. 토큰의 등장 횟수 시각화하기

· 출력 결과



실습 2-4-4-3. 문맥상 유사한 단어 출력하기

· 웹 크롤링 실습에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰를 불러와 텍스트 문맥상 ‘Batman’과 ‘Joker’와 유사한(similar) 단어를 출력해 본다

· TIPS

- similar 함수를 사용해 유사한 단어를 출력한다

실습 2-4-4-3. 문맥상 유사한 단어 출력하기

· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-2-3.py
==================================================
Similar words with Batman:
superhero film action character better iconic movies seen movie actors
heath performance modern difficult villain second end good comes films
==================================================
Similar words with Joker:
time performance jack fan sequel would watch kill recommend penguin
trailer
==================================================

C:₩Users₩USER₩seoul>
```

실습 2-4-4-4. 텍스트의 연어(collocation) 출력하기

· 연어(collocation) [source: https://en.wikipedia.org/wiki/Collocation]

- 문장 내에서 유의미하게 자주 함께 등장하는 단어의 쌍

- 예를 들어, ‘dark’와 ‘black’이 거의 유사한 의미를 지니지만, ‘black chocolate’ 보다는 ‘dark chocolate’이라는 단어의 조합이 훨씬 선호된다. 이 경우 ‘dark’와 ‘chocolate’ 사이에 연어 관계가 있다고 볼 수 있다

- 한국어에도 수많은 연어가 존재한다. 예를 들어, 일상적으로 ‘장갑을 끼다’, ‘옷을 입다’, ‘신발을 신다’ 라고 표현하지 ‘장갑을 입다’, ‘옷을 신다’, ‘신발을 끼다’ 라고 표현하지 않는다

· 실습 1-3-4에서 수집하였던 영화 ‘다크 나이트(The Dark Knight)’ 리뷰 텍스트 내의 연어를 출력해 본다

· TIPS

- collocations 함수를 사용해 연어를 출력한다

실습 2-4-4-4. 텍스트의 연어(collocation) 출력하기

· 출력 결과

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-2-4.py
Collocations for reviews of "The Dark Knight":
dark knight; heath ledger; christian bale; harvey dent; comic book;
christopher nolan; bruce wayne; aaron eckhart; gary oldman; morgan
freeman; batman begins; maggie gyllenhaal; rachel dawes; michael
caine; two face; gotham city; special effects; tim burton; district
attorney; jack nicholson

C:₩Users₩USER₩seoul>
```

실습 2-4-4-5. 다른 리뷰 텍스트 탐색하기

· 2-1부터 2-4까지의 실습 내용을 웹 크롤링 실습에서 수집한 다른 영화 리뷰 데이터에 적용해 보자

- 앞의 예제의 내용에 구애 받지 않고 자유롭게 적용해 본다

실습 2-4-4-5. 다른 리뷰 텍스트 탐색하기

· 출력 결과(인셉션)

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-2-5-inception.py
25458
5525
==================================================
Similar words with Dream:
dreaming plot several know movie character people felt film nothing
layers multiple complex skilled part die saying within super nature
==================================================
Collocations for reviews of "Inception":
special effects; christopher nolan; ellen page; dark knight; shutter
island; cillian murphy; joseph gordon-levitt; within dream; dream
within; leonardo dicaprio; michael caine; tom hardy; dream world;
marion cotillard; hans zimmer; action sequences; blah blah; space
odyssey; 've seen; dom cobb

C:₩Users₩USER₩seoul>
```

실습 2-4-4-5. 다른 리뷰 텍스트 탐색하기

· 출력 결과(올드보이)

관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-2-5-old_boy.py
18301
4313
==================================================
Similar words with old boy:
No matches
==================================================
Collocations for reviews of "Old boy":
main character; old boy; min-sik choi; live octopus; held captive;
park chan-wook; mr. vengeance; fight scenes; eating live; lady
vengeance; monte cristo; sympathy mr.; south korean; fall love; cut
tongue; count monte; chan wook; drama queen; choi min-sik; min sik

C:₩Users₩USER₩seoul>
```

실습 2-4-4-5. 다른 리뷰 텍스트 탐색하기

· 출력 결과(위플래시)

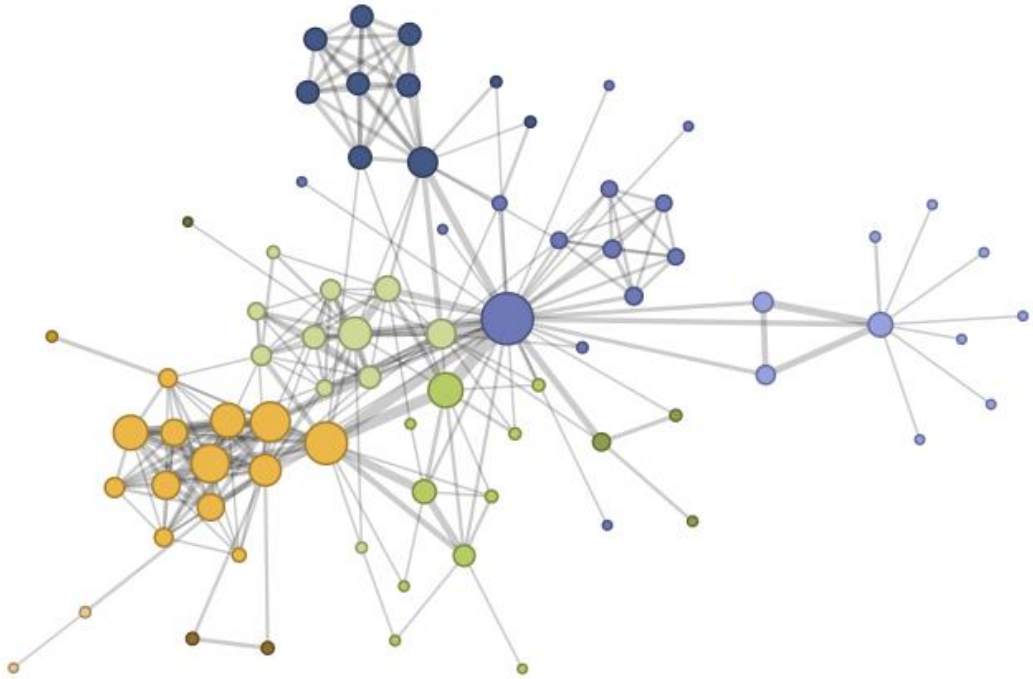
관리자: C:₩Windows₩system32₩cmd.exe

```
C:₩Users₩USER₩seoul>python 3-2-5-whiplash.py
19894
4932
==================================================
Similar words with Drum:
drummer hitting fishing
==================================================
Collocations for reviews of "Whiplash":
miles teller; j.k. simmons; damien chazelle; charlie parker; metal
jacket; drill sergeant; buddy rich; lee ermey; music school; full
metal; car accident; melissa benoist; even though; young drummer;
supporting actor; 21st century; new york; could n't; main character;
boot camp

C:₩Users₩USER₩seoul>
```

실습 2-4-4-6. 연관 단어 그래프 만들기(advanced)

· 실습 2-4-2-3 에서 결과로 출력한 lemmatize된 영화 다크 나이트의 리뷰를 바탕으로, 연관된(같은 문장에서 등장하는) 명사들 끼리 연결된 그래프를 만들어 출력해 본다

- 구현하기 너무 어려우면 완성된 코드를 참고만 해 연관 단어 그래프 만드는 법을 익힌다



실습 2-4-4-6. 연관 단어 그래프 만들기(advanced)

· TIPS

- Unique한 명사들만 추출해 이를 리스트로 만든다

- (총 문장의 개수)를 행의 수로, (unique한 명사의 개수)를 열의 수로 하는 행렬을 만들어 이를 자신의 전치 행렬 (Transpose matrix)와 곱한 co-occurrence matrix를 만든다

- 비가중 무향 그래프(Unweighted, undirected graph)를 만들어 명사들 간의 연관 관계를 표현한다

- numpy와 networkx, 그리고 matplotlib 패키지를 활용한다

실습 2-4-4-6. 연관 단어 그래프 만들기(advanced)

· 참고: co-occurrence matrix

- 특정 문장에서 두 단어가 얼마나 자주 함께 등장하는지를 표현하기 위한 행렬로, occurrence matrix를 자신의 전치 행렬과 곱(matrix multiplication)해 계산한다

- 예시: 문장 2개(문장1, 문장2)와 명사 3개(고양이, 강아지, 치타)가 있고 각 단어가 각 문장에서 등장하는 횟수는 아래와 같다고 가정하자

	고양이	강아지	치타
문장1	1	0	1
문장2	0	1	0

■ occurrence matrix는 아래와 같다

$$M = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \; and \; M^T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

실습 2-4-4-6. 연관 단어 그래프 만들기(advanced)

· 참고: co-occurrence matrix

■ co-occurrence matrix는 다음과 같이 쉽게 계산할 수 있다

$$M^T M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \boxed{\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}}$$

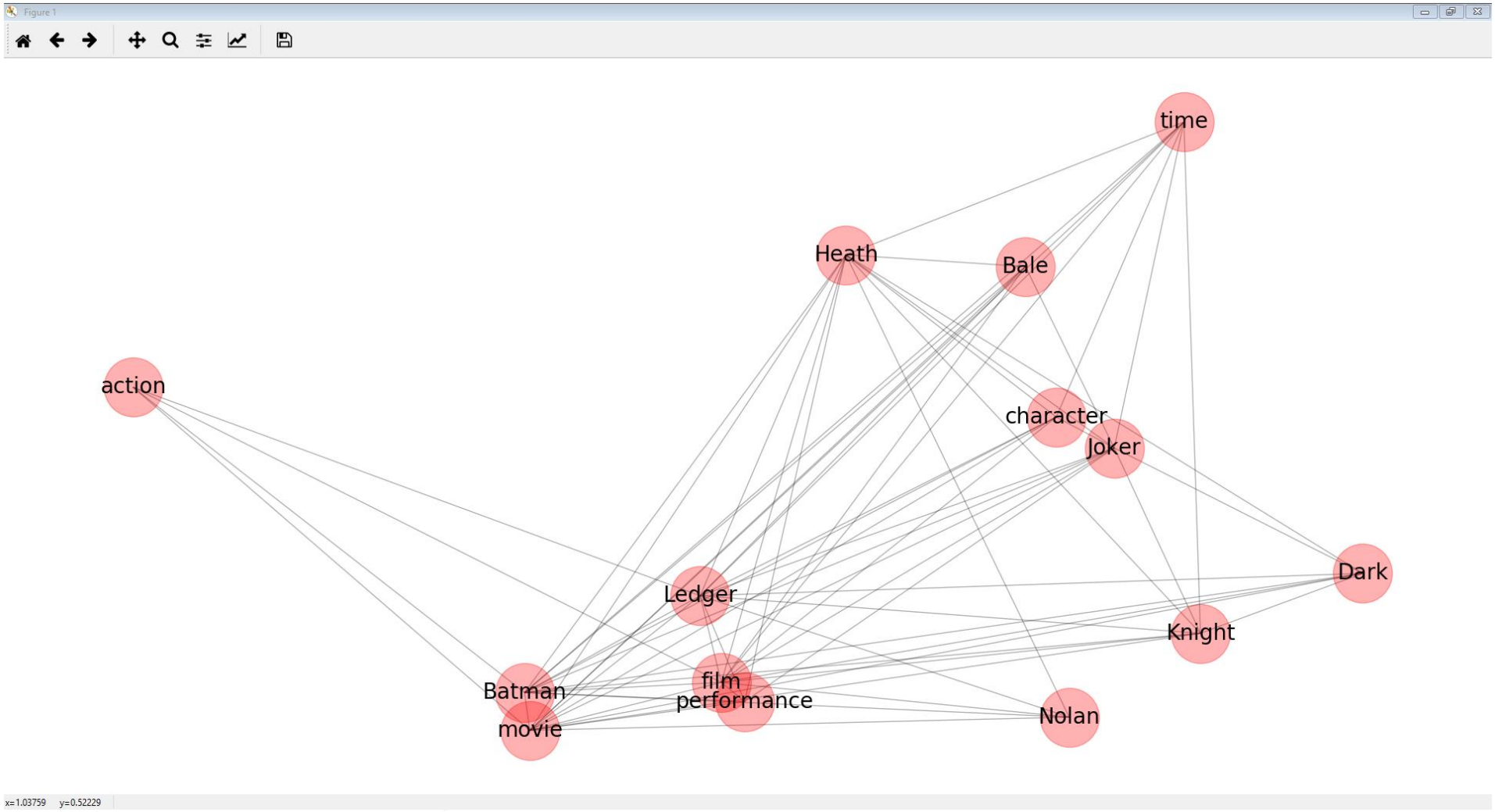
	고양이	강아지	치타
고양이	**1**	**0**	**1**
강아지	**0**	**1**	**0**
치타	**1**	**0**	**1**

■ 고양이와 치타가 문장1에서 함께 같이 등장하므로 co-occurrence matrix의 (1,3) 그리고 (3,1) 원소가 1이다

■ co-occurrence matrix는 무조건 대칭행렬(symmetric matrix)이고 대각선(diagonal) 원소의 값은 1이다 – 이유는?

실습 2-4-4-6. 연관 단어 그래프 만들기(advanced)

· 출력 결과

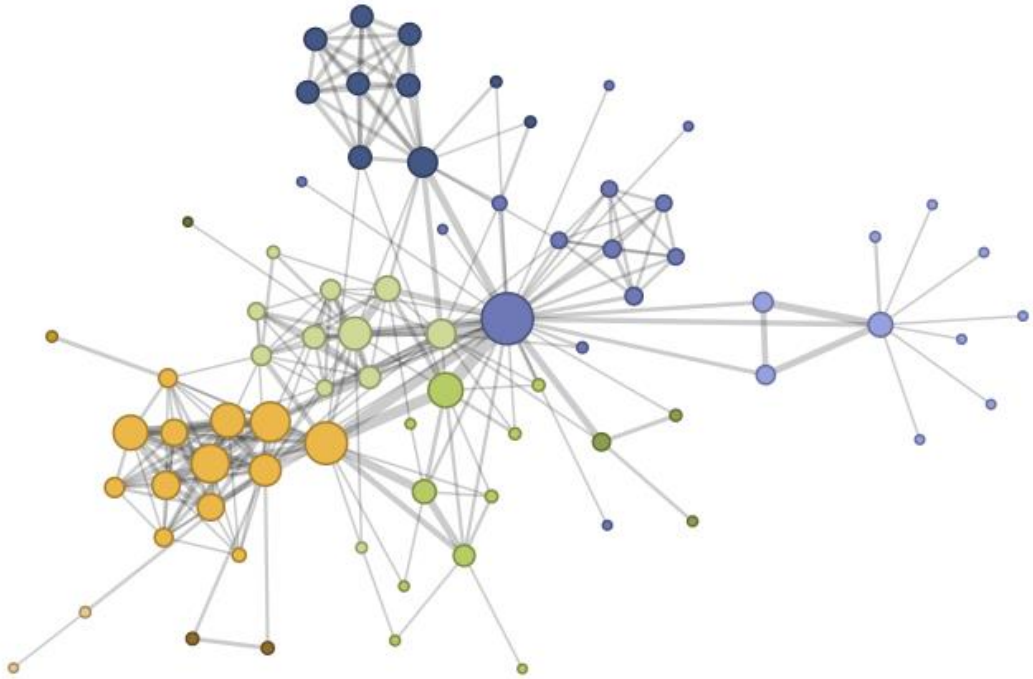


실습 2-4-4-7. 연관 단어 그래프 만들기(advanced) (2)

· 앞선 실습에서 전처리한 다른 영화 리뷰들로도 연관 단어 그래프를 만들어 본다

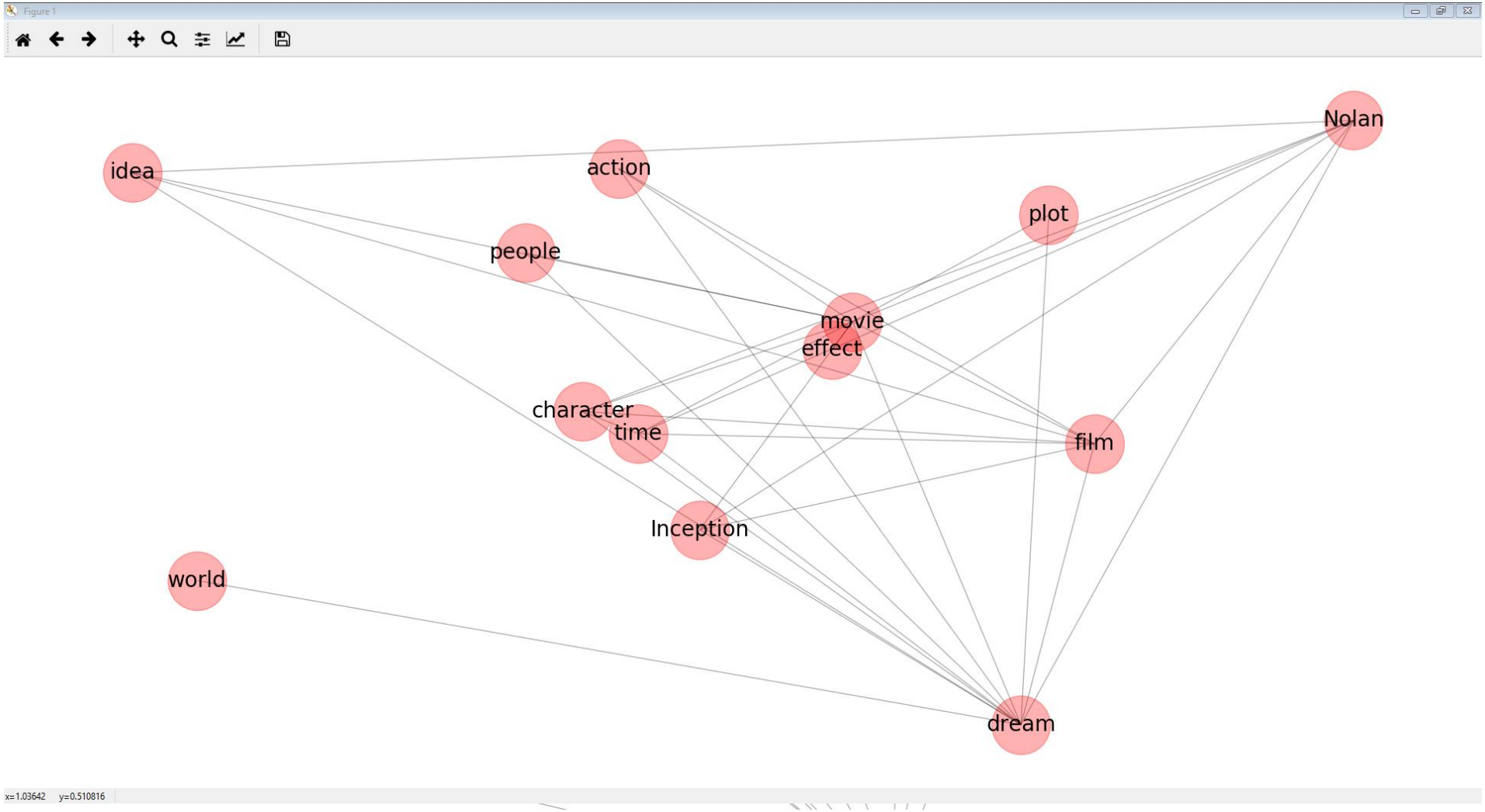
- 구현하기 너무 어려우면 완성된 코드를 참고만 해 연관 단어 그래프 만드는 법을 익힌다

- 출력 결과는 데이터와 파라미터 세팅에 따라 얼마든지 달라질 수 있다



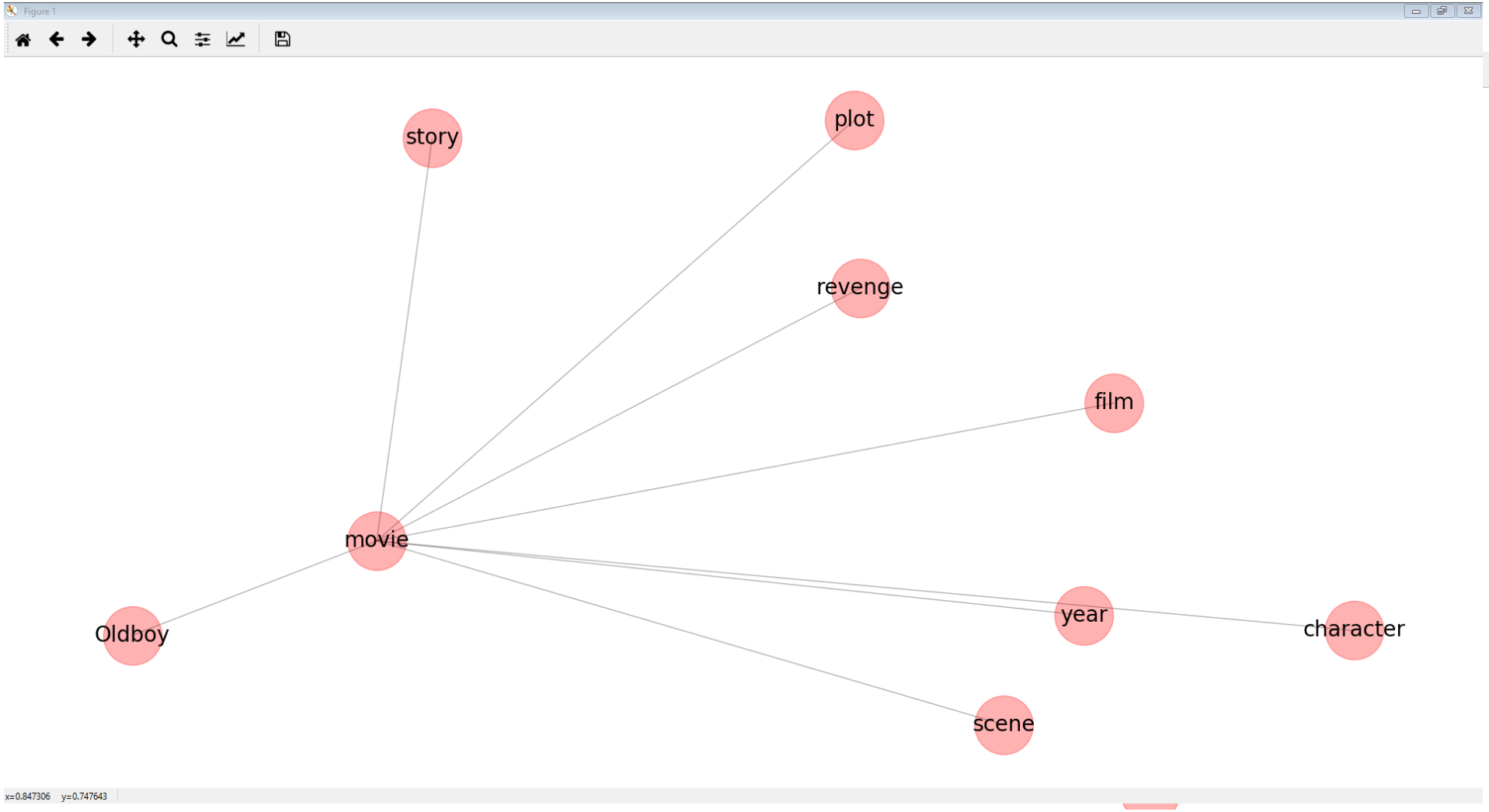
실습 2-4-4-7. 연관 단어 그래프 만들기(advanced) (2)

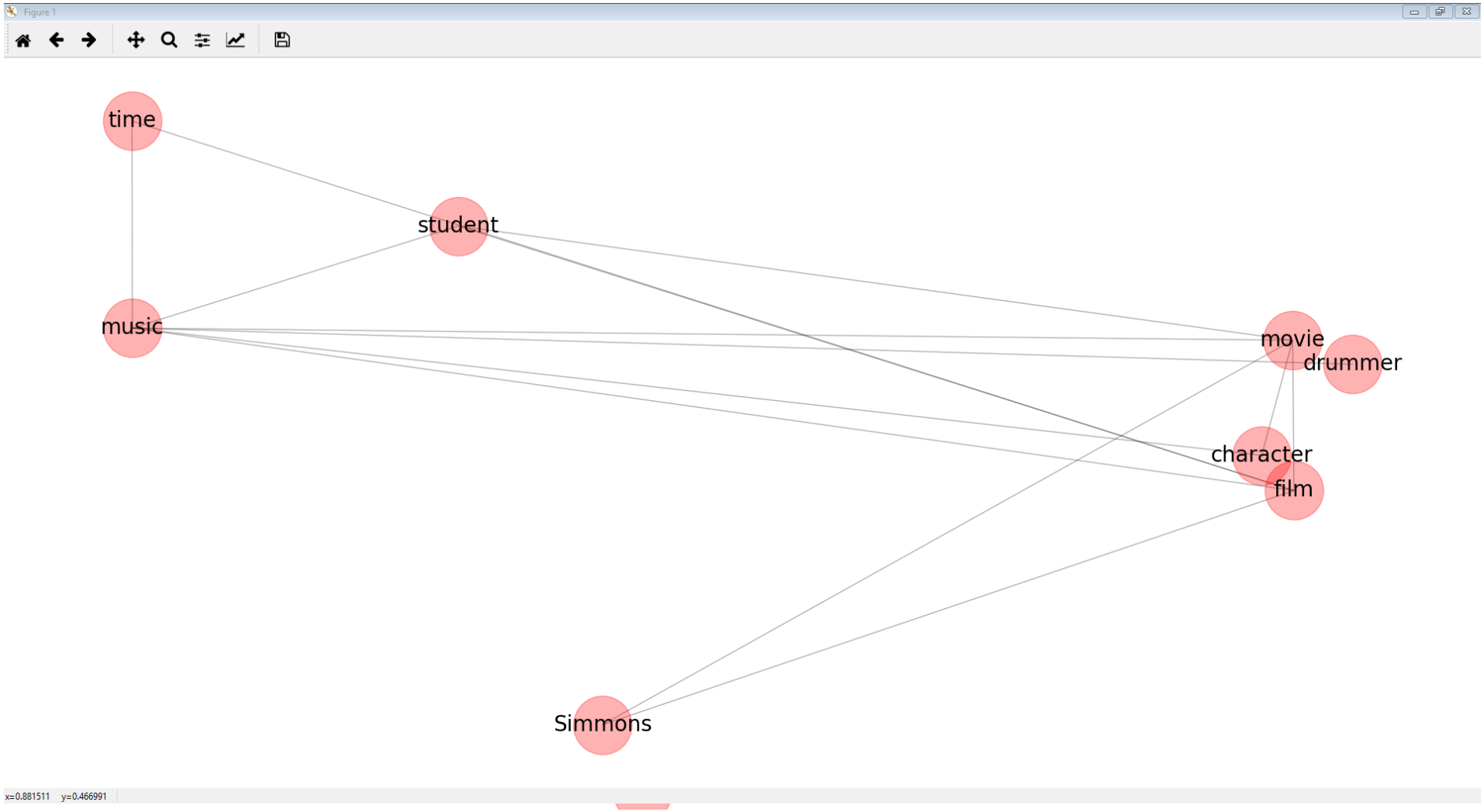
· 출력 결과(인셉션)



실습 2-4-4-7. 연관 단어 그래프 만들기(advanced) (2)

· 출력 결과(올드보이)



실습 2-4-4-7. 연관 단어 그래프 만들기(advanced) (2)

· 출력 결과(위플래시)