

텍스트 데이터 분석 기말 과제

강민수(20182786)

CONTENTS

1. 정릉시장 맛집 데이터 크롤링

1.1 데이터 크롤링 설명

2.1 데이터 크롤링

2. 토큰나이징, 불용어 처리, 말뭉치 생성, 빈도 계수

3. LSA, LDA 설명

4. LSA

5. LDA

6. 결론

1.1 데이터 크롤링 설명



● 정적 크롤링 vs 동적 크롤링

	정적 크롤링	동적 크롤링
연속성	주소를 통해 단발적으로 접근	브라우저를 사용하여 연속적으로 접근
수집 능력	수집 데이터의 한계가 존재	수집 데이터의 한계가 없음
속도	빠름	느림
라이브러리	requests, BeautifulSoup	selenium, chromedriver

- 데이터 크롤링을 할 때 **selenium**을 사용하는 동적 크롤링을 진행하였다.

이유는 음식점의 경우는 중간에 폐업하기도 하고 맛집은 계속 바뀌기 때문에 **VIEW**검색 데이터는 검색하는 시점에 따라서 다를 수 있다.

- 정적 크롤링을 통하여 연구를 진행하면 추후에 비슷한 연구를 진행할 때 코드가 전혀 쓸모없어지기 때문에 동적 크롤링을 사용하는게 맞다고 판단하여 동적 크롤링으로 진행하였다.

네이버 VIEW 검색에서 정릉 근처 맛집 Crawling

'VIEW 검색'은 블로그+카페+포스트+주제별리뷰 검색결과가 통합되어 노출
다양한 출처의 경험, 의견, 리뷰 콘텐츠 등을 모아 볼 수 있어 사용자들의 생생한 이야기를 들을 수 있습니다.

- '정릉시장맛집' VIEW 검색 Crawling

```
1 browser = webdriver.Chrome() #Chromedriver가 없는경우는 다운받아서 파일위치 있는곳에 넣어줄것
2 browser.get('https://naver.com')
3 browser.find_element_by_id("query").click()
4 browser.find_element_by_id("query").send_keys("정릉시장맛집")
5 browser.find_element_by_class_name("ico_search_submit").click()
6 browser.find_element_by_link_text("VIEW").click()
7
8 import time
9 SCROLL_PAUSE_SEC = 1
10
11 # 스크롤 높이 가져올
12 last_height = browser.execute_script("return document.body.scrollHeight")
13
14 while True:
15     # 끝까지 스크롤 다운
16     browser.execute_script("window.scrollTo(0, document.body.scrollHeight);")
17
18     # 1초 대기
19     time.sleep(SCROLL_PAUSE_SEC)
20
21     # 스크롤 다운 후 스크롤 높이 다시 가져올
22     new_height = browser.execute_script("return document.body.scrollHeight")
23     if new_height == last_height:
24         break
25     last_height = new_height
```

1.2 데이터 크롤링

N

정릉시장맛집

dar098

18

99+

통합

VIEW

이미지

지식IN

인플루언서

동영상

쇼핑

뉴스

어학사전

지도

...

공유

전체

블로그

카페

전체

정릉맛집

카페글

VIEW에서 #1일차를 검색해보세요!

허나월드

인플루언서


2022.01.27.

정릉시장 내돈내먹 맛집 모음 (도이칠란드박, 남기남호떡, 슬로카...

정릉시장 맛집모음 안녕하세요 허나짱입니다 ㄹ! 정릉동으로 이사를 온지도 반년 정도가 되었네요. 오고... 골목식당, 함박 메뉴 단 두개! 둘 다 너무 너무 맛있음.....

#정릉시장맛집

#정릉맛집



어느날 고양이

말했다

2022.06.09.

정릉시장 떡갈비 맛집 딱좋은 반찬 고고상회

반찬으로 쓰기 딱 좋으며 사러가자 하길래 정릉시장 반찬 맛집(떡갈비맛집) 고고상회로 이동했다. 정릉시장의... 진짜 육즙이 팡팡 가득한 떡갈비라서,, 음,,, 굳이 맛...


#고고상회

#떡갈비

#떡갈비맛집

#떡갈비맛있는곳

#정릉시장맛집



데일리 미쓰O

2020.09.25.

정릉 시장 가성비맛집 양가네칼국수 수제비


양가네 손칼국수인데 안에 손님도 많더라고요~ 정릉시장 맛집 같아보였어요. 무엇보다.. 나이 드신분들이... 양가네손칼국수 #정릉시장가성비맛집 #정릉시장수제비...

#정릉시장맛집

#정릉시장양가네칼국수

#양가네칼국수

#양가네손칼국수




유랑꼬망스

2022.05.23.

[서울 맛집] 정릉 순대국 추천! 정릉 전통시장 안에... 내돈내산 후...

울 뭉이, 언니랑 오빠 밥 먹는 동안 암전히 기다려줘서 고마워 #서울순대국 #정릉순대국 #정릉시장맛집 #정릉시장 #정릉맛집 #기차순대국 #정릉기차순대국 #서울...



연관 검색어

정릉시장

정릉시장 카페

정릉시장 술집

코로나19

확진현황 및 백신접종 정보

>

신속항원검사

내 주변 검사기관 찾기

>

코로나19 팩트체크

코로나백신 결과 보기

>

잔여백신예약

내 주변 잔여백신 찾기

>

1.2 데이터 크롤링



```
In [9]: 1 view_1_1 = browser.find_elements_by_css_selector('div.total_area > a') # Crawling한 VIEW 제목
2 for i in range(len(view_1_1)):
3     best_store_list.append(view_1_1[i].text)
4     print(i+1, view_1_1[i].text)
```

```
<ipython-input-9-952ee846a89e>:1: DeprecationWarning: find_elements_by_css_selector is deprecated. Please use find_elements(by=By.CSS_SELECTOR, value=css_selector) instead
view_1_1 = browser.find_elements_by_css_selector('div.total_area > a') # Crawling한 VIEW 제목
```

```
1 정릉시장 내돈내먹 맛집 모음 (도미칠란드박, 남기남호역, 슬로카페달팽이, 우리맛곱창, 청수장, 마뽕함박...)
2 정릉시장 맛집 처음 가본 도미칠란드박
3 정릉 시장 가성비맛집 양가네칼국수 수제비
4 정릉시장 칼국수, 수제비 맛집 : 양가네 손칼국수 (feat. 정릉시장 핫플)
5 [정릉역맛집] 정릉시장 개존맛 술집 도미칠란드박 (잡동보르 샌드위치, 킬바사 소시지 플래터)
6 정릉시장 족발 맛집, 한방족발 포장 후기
7 파스타펍 정릉시장 맛집으로 추천
8 [정릉시장 짜꾸미맛집] 정릉짜꾸미를 소개합니다!
9 배우의 꿈을 접고골목식당 출연한 사장님의 사연
10 정릉시장맛집 홍두깨 손 칼국수 손맛이 장난이 아니여유
11 [술샘로/정릉] 정릉시장에 이런곳이, 킬바사맛집, "도미칠란드박"
12 파스타 맛집 / 파스타펍 / 정릉 맛집 / 정릉시장 맛집 / 파스타 배달
13 정릉 맛집 | 정릉시장 중국집 짜장면, 짬뽕, 유린기 맛집 "라이판" 후기
14 [정릉 맛집 / 정릉시장 맛집] 여수 다찌가 생각이 나는 푸짐한 스게다시가 인상적인 정릉 현지인 추천 맛집...
15 정릉시장 맛집 꿀꿀미생일날(고기존맛집) 오늘은 갈매기살이랑 돼지갈비 뿌시기)
16 정릉 마린한시장의 골목집 - 소하마리한
```

```
In [10]: 1 view_1_2 = browser.find_elements_by_css_selector('div.total_dsc_wrap > a') #Crawling한 VIEW 미리보기
2 for i in range(len(view_1_2)):
3     best_store_list.append(view_1_2[i].text)
4     print(i+1, view_1_2[i].text)
```

```
1 정릉시장 맛집모음 안녕하세요 허나짱입니다 ☺ ! 정릉동으로 이사를 온지도 반년 정도가 되었네요. 오고... 골목식당 , 함박 메
뉴 단 두개 ! 둘 다 너무 너무 맛있음... 음식 점수 : ★★★★★ 6. 황궁 위치 : 정릉시장 내...
2 요즘 정릉시장 맛집으로 급부상하고 있다는 도미칠란드 박입니다. 자주 방문하는 곳이 아니다 보니 친구가 알려줘서 알았네요.
정릉시장 쪽 골목 골짜리에 위치해있었습니다. 앞에 사람들이 많아서 눈에 바로 띄었어...
3 양가네 손칼국수인데 안에 손님도 많더라구요~ 정릉시장 맛집 같아보였어요. 무엇보다 .. 나이 드신분들이... 양가네손칼국수 #
정릉시장가성비맛집 #정릉시장수제비 #정릉가성비식당 성북구 술샘로6길 59 02-911-0322 ****-****-****
4 오늘은 점심으로 칼국수가 너무 애크더라구요~ 그래서 찾게된 성북구 정릉시장의 맛집인 양가네 손칼국수를... 즐겨요~~ #서울맛
집 #정릉시장맛집 #정릉맛집 #정릉혼밥 #성북구맛집 #성북구혼밥 #양가네손칼국수 #칼국수...
5
```

```
<ipython-input-10-aec23edb44d5>:1: DeprecationWarning: find_elements_by_css_selector is deprecated. Please use find_elements(by=By.CSS_SELECTOR, value=css_selector) instead
view_1_2 = browser.find_elements_by_css_selector('div.total_dsc_wrap > a') #Crawling한 VIEW 미리보기
```

```
2 개나 주셔서 내가 다 썼다. 담요 정말 두껍고 따뜻함 뉴ㅋㅋㅋㅋㅋ 덕분에 잘 즐기다 갔어요 마디오스~! #정릉역맛집 #정릉역
술집 #정릉시장술집 #정릉시장맛집 #독일음식 #독일소시지 #소시지플래터 #잡동보르샌드위치
6 ㅋㅋ 정릉 시장 한방족발! 아주 대대대대대 만족입니다! 재방문 의사 100% 200% 라 다음에는 저희 동네에서 배달 시켜 먹어보
거나 족발 나오는 시간에 맞춰서 포장하러 가볼거예요! 정말 많은 것 같은데 정작...
7 정릉시장에서 처음 등장했던 파스타집인데 오픈 초반부터 몇 년째 단골로 찾아가는 곳 입니다! 경기도로 이사갔지만 파스타가 먹
고 싶으면 여기까지 종종 찾아가는 나는야 찐단골! 저번에 방문할때는 홈 메뉴가...
```



1.2 데이터 불러오기 및 확인



Data Loading

- Portal_News_Crawler에서 수집한 데이터에서 Text 정보만 불러옴

```
1 # 데이터 불러오기
2 DF_raw = pd.read_excel(DATA_FILE_NAME, sheet_name='sheet1') # 엑셀파일의 sheet1시트의 텍스트 가져오기
3 len(DF_raw)
4
5 # 불러온 데이터의 값이 비어 있는지 확인
6 print('Null값이 있는지 확인합니다.', DF_raw.isnull().values.any()) # Null 값이 존재하는지 확인 (False=정상)
7 print('')
8 DF_raw = DF_raw.dropna(how = 'any') # Null 값이 존재하는 행 제거
9 DF_raw = DF_raw.drop_duplicates() # 중복 데이터 프레임 제거
10 DF_raw = DF_raw.reset_index(drop=True) # 데이터 프레임 재색인
11 print('중복 및, NULL값을 제거한 후, 다시 NULL값을 확인 합니다.', DF_raw.isnull().values.any()) # Null 값이 존재하는지 확인
12 print('')
13 print("처리할 데이터수 : ", len(DF_raw))
14
15 # raw데이터로부터 텍스트만 불러오기
16 DF_only_text = DF_raw['text'] #엑셀필드명
```

Null값이 있는지 확인합니다. True

중복 및, NULL값을 제거한 후, 다시 NULL값을 확인 합니다. False

처리할 데이터수 : 922

```
1 df_jl = pd.read_csv('jeongleungfood.csv')
```

```
1 df_jl.content
```

```
0      정릉시장 맛집모음 안녕하세요 허나짱입니다 ㅎ ! 정릉동으로 이사를 온지도 반년 정...
1      울 똥이, 언니랑 오빠 밥 먹는 동안 암전히 기다려줘서 고마워 #서울순대국 #정릉순...
2      오늘은 정릉시장 안에 있는 유일한 무한리필 고기집 고기굽터를 소개시켜드리려고 해요!...
3      양가네 손칼국수인데 안에 손님도 많더라구요~ 정릉시장 맛집 갈아보았어요. 무엇보다 ...
4      요즘 정릉시장 맛집으로 급부상하고 있다는 도미찰란도 박입니다. 자주 방문하는 곳이 ...

118     배달파트너, 요기요, 배달대행, 퀵서비스, 카카오크, 게시판 준수 부탁드립니다. -...
119     양파가 있어서 간장 소스도 따로 만들어먹을정도였어요 흥분하게 줬으면 좋았을텐데 ! ...
120     선릉역 10번출구에서 주육 직진하다가 구두수선코너에서 좌회전을하면 선정릉대표소가 대...
121     에서 골목식당촬영하고있다고 구경간저희친언니가찍어서보냈네요 앞엔백종원뒤엔김성주 ㅎㅎ 직...
122     동네가 멀지 않아서 성신여대엔 자주 오가는 편인데 생소한 정릉시장에 위치한 칼국수맛...
Name: content, Length: 123, dtype: object
```


2. 토큰나이징, 불용어 처리, 말뭉치 생성, 빈도 계수



토큰나이징, 불용어 처리, 말뭉치 생성, 빈도 계수

```
1 tokenizer = Okt() # 토큰나이저 지정
2 stopwords_vocab = ['정동', '시장', '맛집'] # 불용어 파일 불러오기
3 sep = "\n" # 불용어 처리 문자
4
5 def build_vocab(data_frame, stopwords_vocab, separate):
6
7     # 불용어 데이터를 가져와 리스트로 변환합니다.
8     # with open(stopwords_vocab, encoding = 'utf-8') as f:
9     #     temp1 = []
10    #     for i in f:
11    #         temp1.append(i)
12    #
13    # globals()['stopwords_vocab'] = []
14
15    # 불용어 데이터는 전역변수 stopwords_vocab 선언합니다.
16    # 구분자에 따라 stopwords_vocab에 추가하여 불용어 사전을 구축합니다.
17    # for j in range(len(temp1)):
18    #     temp2 = temp1[j].rstrip(separate)
19    #     globals()['stopwords_vocab'].append(temp2)
20
21    #okt token에서 명사와 출력합니다. 단어의 길이가 1 초과인 단어만 출력합니다.
22    globals()['list_sent2words'] = []
23    for i in range(len(data_frame)):
24        num_list = []
25        temp = tokenizer.nouns(data_frame[i])
26        for j in range(len(temp)):
27
28            if len(temp[j]) > 1:
29                num_list.append(temp[j])
30            globals()['list_sent2words'].append(num_list)
31
32    return [[word for word in doc if word not in globals()['stopwords_vocab']] for doc in globals()['list_sent2words']]
33
34 result_data = build_vocab(df_jl.content, stopwords_vocab, sep)
35
36 # 전체에 대한 워드 카운트 개수 확인
37
38 def word_corpus(result_data):
39     #전체 단어의 개수 파악
40     words = list(itertools.chain(*result_data))
41     print('전체 워드의 개수 : {}'.format(len(words)))
42
43     #단어의 빈도수를 확인 후 추가할 불용어 확인 작업
44     vocab = Counter(words)
45     vocab_size = len(words)
46     vocab = vocab.most_common(vocab_size) # 등장 빈도수가 높은 상위 n개의 단어만 저장 vocab
47     return vocab
48
49 vocab = word_corpus(result_data)
50
51
```

전체 워드의 개수 : 1549

	text	count
0	식당	40
1	위치	25
2	오늘	19
3	골목	18
4	추천	17
5	대국	17
6	아리랑	15
7	소개	14
8	메뉴	12
9	근처	12
10	기차	12
11	도이칠란드	11
12	방문	11
13	사람	11
14	정말	11
15	동네	11
16	아구찜	11
17	성북구	10
18	시간	10
19	바로	9

3. LSA, LDA 설명



- LSA : 잠재 의미 분석(Latent Semantic Analysis)

LSA는 정확히 토픽 모델링을 위해 최적화 된 알고리즘은 아니지만, 토픽 모델링이라는 분야에 아이디어를 제공한 알고리즘이라 볼 수 있다. BoW에 기반한 것들은 단어의 의미를 고려하지 못한다는 단점이 있다. 이를 위한 대안으로 잠재 의미 분석(LSA)이란 방법이 있다. 이 방법을 이해하기 위해서는 선형대수학의 특이값 분해(Singular Value Decomposition, SVD)를 이해할 필요가 있다.

- SVD : 특이값 분해(Singular Value Decomposition, SVD)

SVD란 A 가 $m \times n$ 행렬일 때, 다음과 같이 3개의 행렬의 곱으로 분해(decomposition)하는 것을 말한다

- LDA : 잠재 디리클레 할당(Latent Dirichlet Allocation)

LDA는 문서의 집합으로부터 어떤 토픽이 존재하는지를 알아내기 위한 알고리즘이다. 앞서 배운 빈도수 기반의 표현 방법인 BoW의 행렬 DTM 또는 TF-IDF 행렬을 입력으로 하는데, 이로부터 알 수 있는 사실은 단어의 순서는 신경쓰지 않겠다는 점이다.

4. LSA 모델

LSA 모델

```
1 from gensim import models
```

```
1 NUM_TOPIC_WORDS = 10
```

```
1 # ★ 모델링 후 각 토픽별로 중요한 단어들을 표시
2 def print_topic_words(model):
3
4     for topic_id in range(model.num_topics):
5         topic_word_probs = model.show_topic(topic_id, NUM_TOPIC_WORDS)
6         print("Topic ID: {}".format(topic_id))
7
8         for topic_word, prob in topic_word_probs:
9             print("{} {}".format(topic_word, prob))
10        print("\n")
```

```
1 model_LSA = models.LsiModel.LsiModel(corpus, num_topics=NUM_TOPICS, id2word=corpora.Dictionary(result_data))
```

```
1 print_topic_words(model_LSA) # 전체 토픽별 주요 어휘 출력
```

Topic ID: 0

식당	0.5636228127844031
대국	0.3805810669630907
골목	0.2811490824442053
기차	0.25310662087933716
아리랑	0.24663986012729774
위치	0.21331368507305978
오늘	0.1645193376404964
백종원	0.13638333193977895
근처	0.11403167508441524
장수식	0.11362823897076055

Topic ID: 1

대국	-0.6937313177843054
기차	-0.4190375423651655
식당	0.33192304972439357
골목	0.2343850882823268
아리랑	0.20832058428683056
백종원	0.11442152536760024
장수식	0.09248683558915463
외관	-0.0859377085405263
청국장	0.07859578654290425
오늘	-0.07054573754988012

Topic ID: 2

위치	-0.38165859356475873
아구찜	-0.30323536206608126
추천	-0.2305239729253408
초밥	-0.20838389085227438
식당	0.20248626212006438
오늘	-0.17418038121345847
볶음밥	-0.15587315726535955
파스타	-0.15578222499399028
정말	-0.15276434701960764
아리랑	0.14732963277894237

Topic ID: 3

초밥	-0.8213852180616641
강동	-0.2451196805630071
근처	-0.17661794667649958
추천	0.13359263250648717
아마	-0.1280530365752176
먹음	-0.12255984028150355
후훗	-0.12255984028150355
비주	-0.12255984028150355
세트	-0.1211042535981576
사진	-0.11696490234072186

Topic ID: 4

추천	-0.4323721994498716
위치	0.39496065777043793
파스타	-0.25030548037088723
오늘	0.24755561810840948
아구찜	-0.24220856126961346
정말	-0.1825317574452824
식당	-0.17630687503187864
볶음밥	-0.1686078549662047
소개	0.15963339469284568
사람	0.12480838704911568

Topic ID: 5

아구찜	0.4638762481044352
동네	-0.25250475354937785
방문	-0.2133836513809797
아리랑	0.2017085489491622
고기	-0.19119310354753516
볶음밥	0.1828853741236621
도이칠란드	-0.14287582094501133
근처	-0.1422394022710733
골목	0.13638571036531794
북한	-0.1363816140847618

5.1 LDA 토픽 기본 모델링

```
1 print('토픽 기본 모델링을 실시 합니다. 해당 모델은 "lda_model" 변수로 입력됩니다.')
2 print(' ')
3
4 NUM_TOPICS = int(input('토픽의 개수를 입력해 주세요. '))
5 TOPICS_W_NUM = int(input('출력할 토픽별 단어의 개수를 입력해 주세요 '))
6 save_lda_model = int(input('선택한 토픽 모델을 저장하시겠습니까? #n0 저장 #n1 미저장 '))
7
8 RANDOM_STATE = 100
9 UPDATE_EVERY = 1
10 CHUNKSIZE = 100
11 PASSES = 10
12 ALPHA = 'auto'
13 PER_WORD_TOPICS = True
14
15 # 해당 섹션은 토픽모델링(LDA)에 대해 모델을 정의하는 섹션입니다.
16 lda_model = gensim.models.ldamodel.LdaModel(corpus=corpus, id2word=id2word,
17                                             num_topics=NUM_TOPICS, random_state=RANDOM_STATE,
18                                             update_every=UPDATE_EVERY, chunksize=CHUNKSIZE,
19                                             passes=PASSES, alpha=ALPHA, per_word_topics=PER_WORD_TOPICS)
20
21 # 토픽 출력
22 pprint(lda_model.print_topics(num_words=TOPICS_W_NUM))
23 doc_lda = lda_model[corpus]
24
25 # 모델 저장
26 if save_lda_model == 0:
27     lda_model.save(LDA_MODEL_SAVE_NAME)
28 # 0번 토픽 - 중요단어들이 가중치 순으로 나옴(20개)
```

토픽 기본 모델링을 실시 합니다. 해당 모델은 "lda_model" 변수로 입력됩니다.

토픽의 개수를 입력해 주세요. 6

출력할 토픽별 단어의 개수를 입력해 주세요 10

선택한 토픽 모델을 저장하시겠습니까?

0 저장

1 미저장 0

```
{(0,
  '0.028*\'위치\' + 0.020*\'사진\' + 0.017*\'식당\' + 0.016*\'층수\' + 0.016*\'건물\' + 0.014*\'오늘\' +
  ' + 0.012*\'입구\' + 0.011*\'정도\' + 0.011*\'메뉴\' + 0.011*\'타고\''),
 (1,
  '0.030*\'식당\' + 0.023*\'생각\' + 0.020*\'국민대\' + 0.015*\'청년\' + 0.014*\'막걸리\' +
  '0.014*\'사람\' + 0.013*\'산보\' + 0.013*\'아주\' + 0.012*\'동네\' + 0.012*\'국문\''),
 (2,
  '0.024*\'느낌\' + 0.023*\'시간\' + 0.017*\'골목길\' + 0.017*\'영업\' + 0.017*\'입구\' +
  '0.017*\'저녁\' + 0.016*\'근처\' + 0.015*\'정릉천\' + 0.014*\'요즘\' + 0.013*\'먹거리\''),
 (3,
  '0.027*\'메뉴\' + 0.026*\'출구\' + 0.023*\'대국\' + 0.019*\'북한\' + 0.019*\'도보\' + 0.018*\'식당\' +
  ' + 0.017*\'민속\' + 0.016*\'우이신설선\' + 0.016*\'가게\' + 0.015*\'기차\''),
 (4,
  '0.035*\'동네\' + 0.021*\'배달\' + 0.021*\'호호\' + 0.021*\'카레\' + 0.018*\'볶음밥\' +
  '0.016*\'소개\' + 0.016*\'가지\' + 0.015*\'구경\' + 0.015*\'저녁\' + 0.014*\'위치\''),
 (5,
  '0.048*\'식당\' + 0.035*\'골목\' + 0.031*\'아리랑\' + 0.029*\'백종원\' + 0.019*\'사장\' +
  '0.018*\'모두\' + 0.017*\'짜장면\' + 0.017*\'보리밥\' + 0.014*\'파스타\' + 0.012*\'마뽀함박\''))}
```

5.2 LDA 토픽 평가



```
In [ ]: 1 #토픽평가
        2 """
        3 해당 셀은 설계한 모델을 계산하는 셀입니다.
        4 측정은 Perplexity와 Coherence Score입니다.
        5 """
        6
        7 # Perplexity
        8 print('\nPerplexity: ', lda_model.log_perplexity(corpus)) # a measure of how good the model is. lower the better.
        9
        10 # Coherence Score
        11 coherence_model_lda = CoherenceModel(model=lda_model, texts=result_data, dictionary=id2word, coherence='c_v')
        12 coherence_lda = coherence_model_lda.get_coherence()
        13 print('\nCoherence Score: ', coherence_lda)
        14 # Perplexity는 작을 수록 Coherence Score는 높을 수록 좋다. 모델 1개의 값
        15 # 토픽의 개수를 다르게 하여 판단 비교 해보세요.
        16 # 젠심 코히런스 코스로 검색해봐서 coherence='c_v' 값을 바꿔가면서 해보세요
```

Perplexity: -7.120796976684369

Coherence Score: 0.4431963187016972

5.3 LDA 토픽별 키워드 조회



Topic ID: 0

위치	0.02848828211426735
사진	0.019662857055664062
식당	0.016969211399555206
층수	0.01605936326086521
건물	0.016059160232543945
오늘	0.013784802518785
입구	0.011646302416920662
정도	0.010929428040981293
메뉴	0.010708834044635296
타고	0.010703622363507748

Topic ID: 1

식당	0.030197495594620705
생각	0.023427467793226242
국민대	0.01983819715678692
청년	0.015393372625112534
막걸리	0.014332178048789501
사람	0.013568541966378689
산보	0.012859427370131016
아주	0.01284735556691885
동네	0.011827626265585423
국문	0.011755856685340405

Topic ID: 2

느낌	0.023971889168024063
시간	0.023255683481693268
골목길	0.016552384942770004
영업	0.01654745638370514
입구	0.016545835882425308
저녁	0.016530338674783707
근처	0.01646403968334198
정릉천	0.01483615767210722
요즘	0.013647637329995632
먹거리	0.013128062710165977

Topic ID: 3

메뉴	0.026733482256531715
출구	0.026239069178700447
대국	0.023174511268734932
북한	0.018765628337860107
도보	0.018728261813521385
식당	0.01780254952609539
민속	0.017400844022631645
우이신설선	0.01583845540881157
가게	0.015611366368830204
기차	0.014924811199307442

Topic ID: 4

동네	0.035242483019828796
배알	0.021412312984466553
호호	0.020709309726953506
카레	0.020709309726953506
볶음밥	0.01819918118417263
소개	0.01639971137046814
가지	0.01606355793774128
구경	0.014996573328971863
저녁	0.014718295074999332
위치	0.01394092570990324

Topic ID: 5

식당	0.04775138944387436
골목	0.03466831520199776
아리랑	0.030860770493745804
백종원	0.028888966888189316
사장	0.01894877664744854
모두	0.017727695405483246
짜장면	0.01659274660050869
보리밥	0.01656627096235752
파스타	0.013829137198626995
마몽함박	0.012256860733032227

분석결과

- 1번 TOPIC : 위치, 사진, 식당, 층수, 건물 등이 관련이 높은 것을 보아 1번 TOPIC은 정릉시장맛집들에 대한 위치 정보 및 생김세에 관한 TOPIC일 가능성이 높다.
- 2번 TOPIC : 식당, 생각, 국민대, 청년 등이 관련이 높은 것을 보아 2번 TOPIC은 국민대생들이 많이 가거나 국민대 근처에 있는 정릉시장맛집들일 가능성이 높다.
- 3번 TOPIC : 느낌, 시간, 골목길, 영업, 입구, 저녁, 근처 등이 관련이 높은 것을 보아 3번 TOPIC은 정릉시장맛집들에 대한 영업 시간 및 위치 및 음식점 분위기 등에 관한 TOPIC일 가능성이 높다.
- 4번 TOPIC : 메뉴, 출구, 대국, 북한, 도보, 식당, 민속, 우이신설선 등이 관련이 높은 것을 보아 4번 TOPIC은 우이신설선인 북한산보국문역 근처 맛집이거나 역과의 거리가 나와있는 TOPIC일 가능성이 높다.
- 5번 TOPIC : 동네, 배달, 카레, 호호, 볶음밥 등이 관련이 높은 것을 보아 5번 TOPIC은 배달이 되는 정릉시장 맛집들에 대한 정보거나 카레, 볶음밥 등등을 파는 음식점이 관련이 있는 TOPIC일 가능성이 높다.
- 6번 TOPIC : 식당, 골목, 아리랑, 백종원 등이 관련이 높은 것을 보아 6번 TOPIC은 백종원 골목 식당이라는 프로그램과 관련된 정릉시장 맛집일 가능성이 높다.



THANK YOU