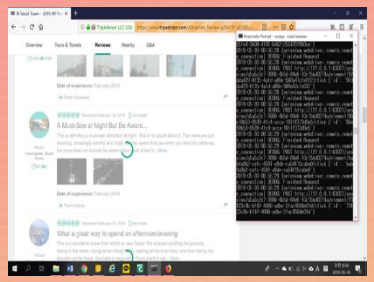
# 자연어 처리를 활용한
# 서울 종로구 관광지 리뷰 분석

SNU Growth Hackers
고이경 / 김정은 / 최수연 / 현아영

# Contents


01. Background

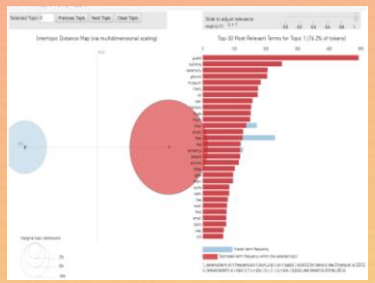


02. Data Crawling

| | review | rating | neg | pos | neu | compound | label |
|---|---|---|---|---|---|---|---|
| 0 | great look feel stepped back century novelty b... | 4.0 | 0.000 | 0.301 | 0.699 | 0.8481 | Positive |
| 1 | bad lot see fairly packed tourists even rainy ... | 2.0 | 0.298 | 0.205 | 0.497 | -0.1280 | Negative |
| 2 | houses look great nice walk around absolutely ... | 3.0 | 0.000 | 0.496 | 0.504 | 0.7845 | Positive |
| 3 | people housing area resident time business ren... | 3.0 | 0.000 | 0.145 | 0.855 | 0.5994 | Positive |
| 4 | bukchon hanok village situated middle modern p... | 3.0 | 0.000 | 0.272 | 0.728 | 0.7964 | Positive |
| 5 | really enjoyed walk around hilly neighborhood ... | 4.0 | 0.000 | 0.366 | 0.634 | 0.8858 | Positive |
| 6 | beautiful area hill small alleys many still in... | 5.0 | 0.000 | 0.326 | 0.674 | 0.8910 | Positive |
| 7 | like nice relaxing quite walk around outside l... | 5.0 | 0.000 | 0.548 | 0.452 | 0.8176 | Positive |
| 8 | nice romantic local korean village walk korean... | 5.0 | 0.000 | 0.444 | 0.556 | 0.9625 | Positive |
| 9 | lucky enough walk streets less crowded ones co... | 5.0 | 0.000 | 0.128 | 0.872 | 0.4215 | Positive |
| 10 | like transported back time definitely must obs... | 5.0 | 0.201 | 0.327 | 0.472 | 0.4939 | Positive |
| 11 | place good visit great view peace cos resident... | 5.0 | 0.000 | 0.571 | 0.429 | 0.9231 | Positive |
| 12 | place great see old homes however high tech do... | 5.0 | 0.000 | 0.194 | 0.806 | 0.6249 | Positive |
| 13 | minutes away gyeongbokgung palace changdeokung... | 3.0 | 0.192 | 0.123 | 0.685 | -0.2500 | Negative |
| 14 | glad accompanied guide explained background re... | 4.0 | 0.143 | 0.251 | 0.606 | 0.3612 | Positive |

03. Sentiment Analysis (with VADER)


04. Deep Learning (LSTM)



05. Wordcloud & Topic Modeling

1

# Background of Analysis

# 분석 배경

**공통 관심사**

"관광 데이터"

Q. 서울 관광지에 대한 외국인들의 생각? → 관광지 리뷰 데이터

**중간 발표에서의 피드백**

딥러닝 모델 구축을 통한 리뷰 데이터 라벨링 (긍정/부정)

**분석 순서**

긍정/부정으로 라벨링 된 관광지 후기 dataset 만들기
→ dataset으로 sentiment classification 모델 만들기
→ 분석하고자 하는 데이터(object data)에 모델 적용, 긍정/부정으로 분류
→ 분류된 후기를 바탕으로 긍정/부정 요인 추출

# 분석 배경

-한국관광공사 외래관광객실태조사(2018)

## 아. 관광지 매력도 만족도

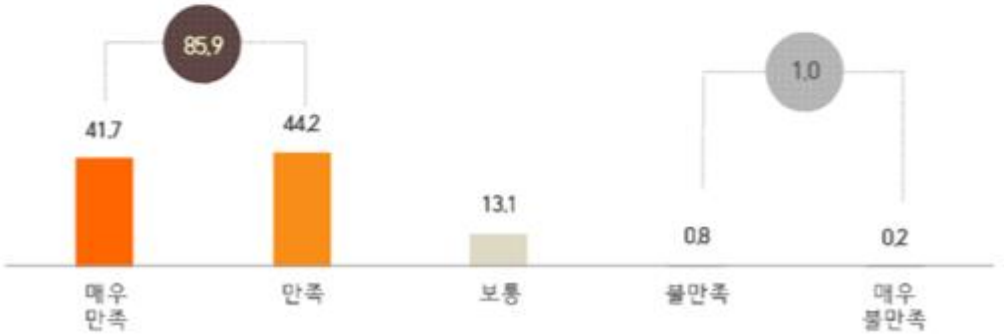
- 관광지 매력도에 대한 '만족' 비율은 85.9%로 나타남

### 1) 전체 분석

- 2018년 관광지 매력도에 대해 살펴보면, 방한 외래관광객 중 85.9%가 '만족'(매우 만족 : 41.7% + 만족 : 44.2%)한 것으로 나타남
- 한편, '보통' 비율은 13.1%, '불만족' 비율은 1.0%(매우 불만족 : 0.2% + 불만족 : 0.8%)로 조사됨

| 그림 3-53 | 관광지 매력도 만족도

(단위: %)



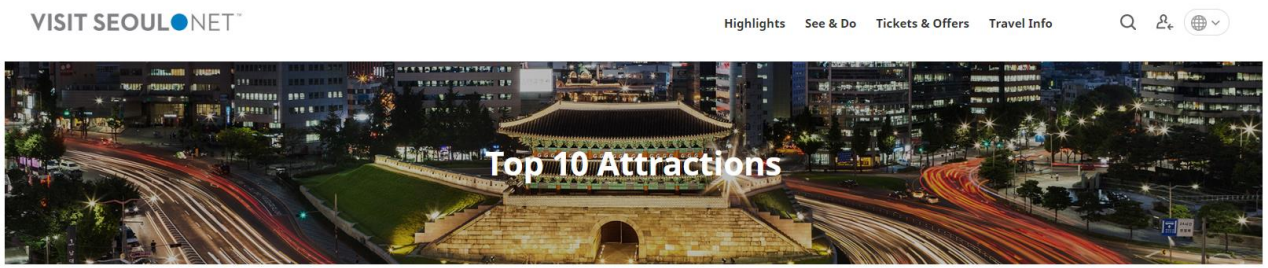

기존 조사는 모든 관광지의 만족도를 단편적으로 조사한다는 한계,

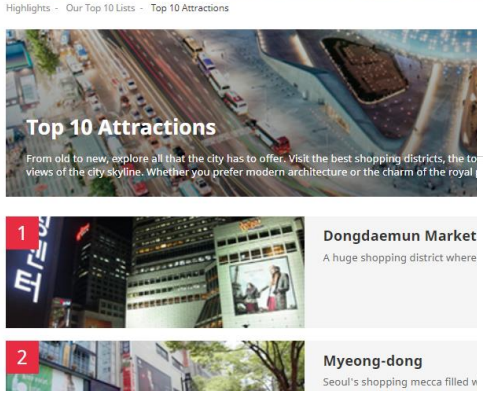
**개별 관광지**의 어떤 부분이 만족/불만족스러운지를 분석!

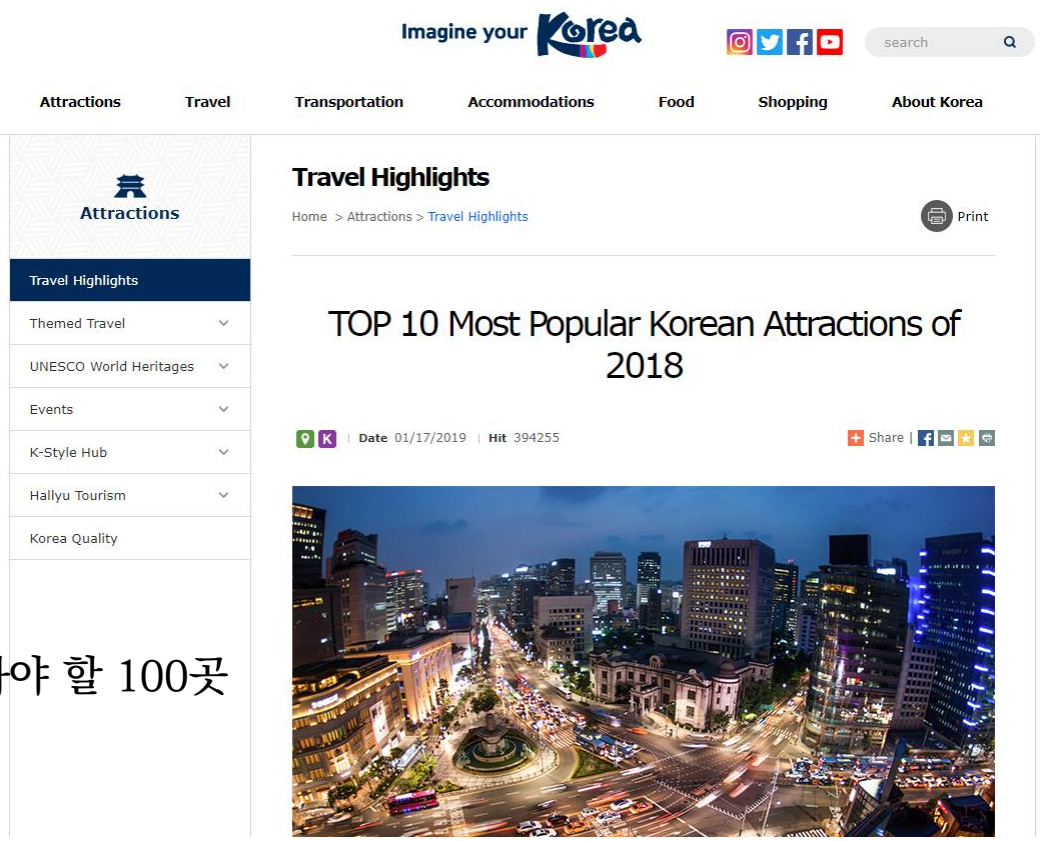
2

# Data Crawling

# 크롤링할 관광지 선정

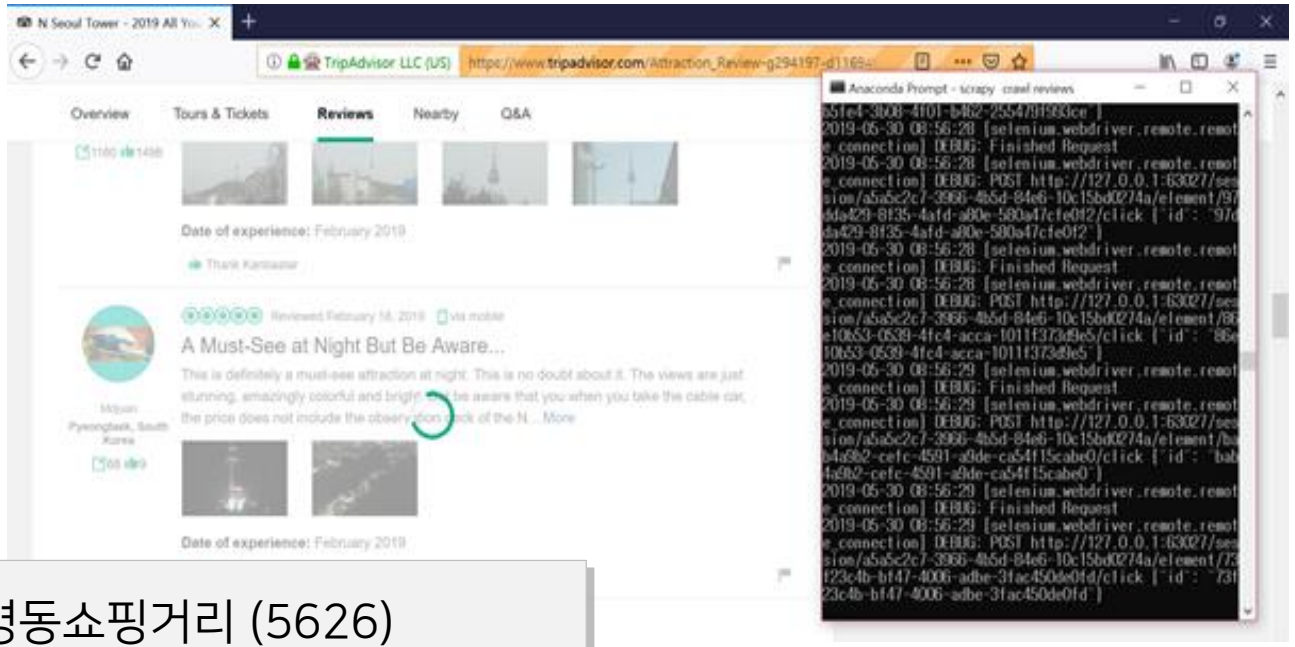



-한국관광공사 2019년에 한국에서 꼭 가봐야 할 100곳
-Visit seoulnet 서울의 top10 관광지

# 서울 주요 14개 관광지 리뷰 tripadvisor에서 크롤링

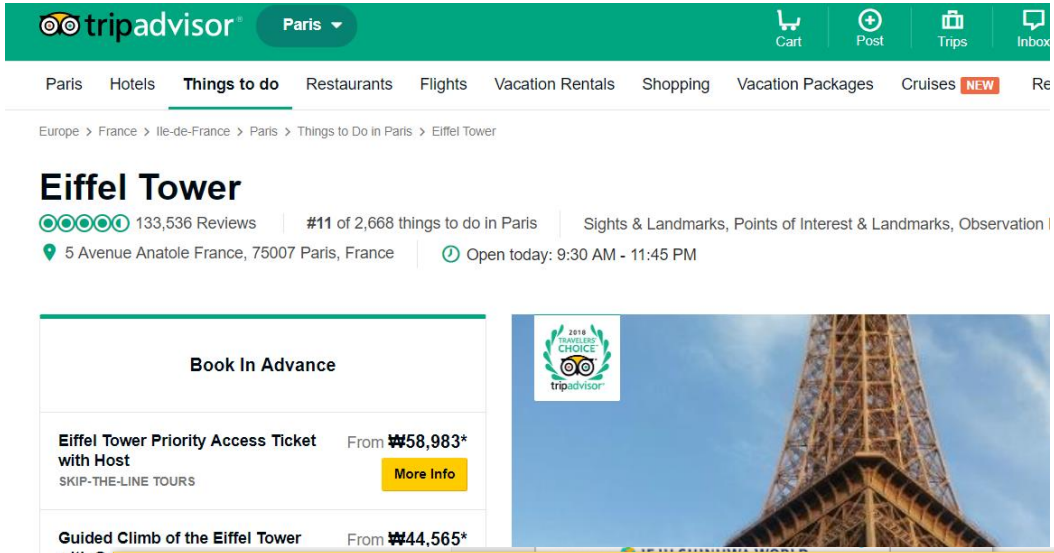


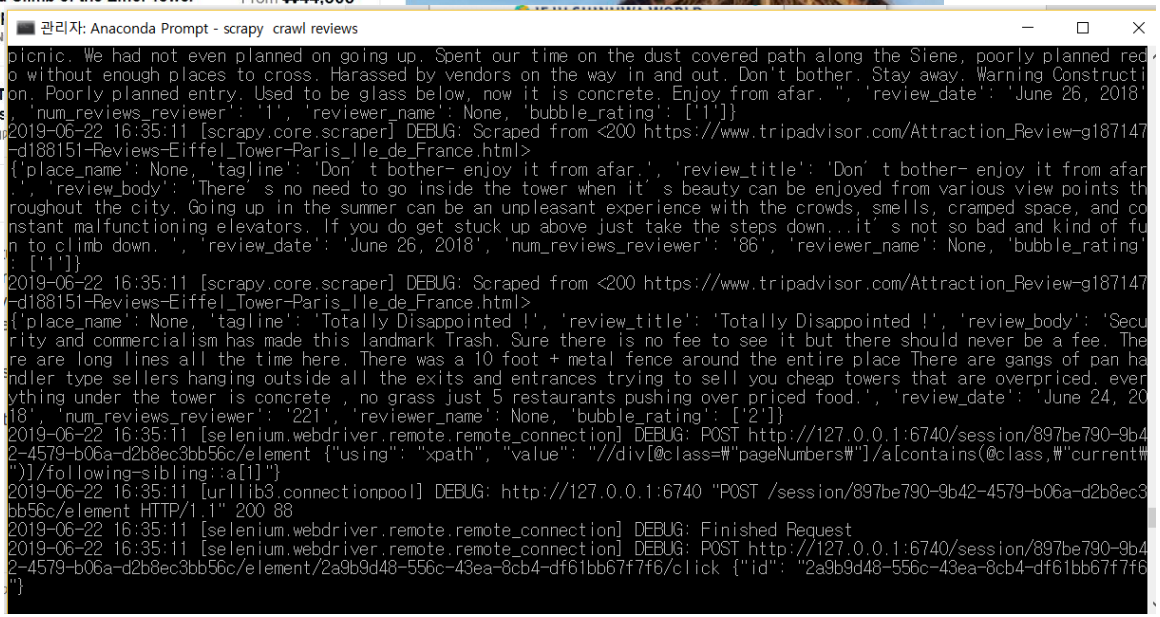

명동쇼핑거리 (5626)
동대문시장 (5077)
DDP (884)
남대문시장 (1264)
남산타워 (4995)
전쟁기념관 (2840)
홍대 쇼핑거리 (1039)
코엑스몰(307)
롯데월드 (1317)
롯데월드타워 (258)

경복궁 (5724)
인사동거리 (4901)
북촌한옥마을 (2488)
청계천 (1787)

**종로구 관광지 리뷰 데이터 (object data)**

# 해외 유명 관광지 7곳 부정적인 리뷰만 크롤링




# 데이터 전처리 1)불용문 제거, 소문자화

```
from nltk.corpus import wordnet

def get_wordnet_pos(pos_tag):
    if pos_tag.startswith('J'):
        return wordnet.ADJ
    elif pos_tag.startswith('V'):
        return wordnet.VERB
    elif pos_tag.startswith('N'):
        return wordnet.NOUN
    elif pos_tag.startswith('R'):
        return wordnet.ADV
    else:
        return wordnet.NOUN
```

```
impor
nltk.d

impor
from r
from r
from r
from r
impor
impor
```

```
def clean_text(text):
    # lower text
    text = text.lower()
    # 문자 제외 다 제거
    text= re.sub('[^a-zA-Z]', ' ', str(text))
    #토크나이즈
    text = [word.strip(string.punctuation) for word in text.split
    #토크나이즈 뒤에 stopwords 제거
    stop = stopwords.words('english')
    text = [x for x in text if x not in stop]
    # remove empty tokens
    text = [t for t in text if len(t) > 0]
    # pos tag text
    pos_tags = pos_tag(text)
    # remove words with only one letter
    text = [t for t in text if len(t) > 1]
    # join all
    text = " ".join(text)
    return(text)
```

```
# clean text data
coex = coexmall['review_body'].apply(lambda x: clean_text(x))
print(coex)
```

```
0     lots typical mall including movie theatre aqua...
1     normally frequent several favorite stores mall...
2     library huge nice almost books korean unless k...
3     walk least one saturday month shop brands hyun...
4     coex mall upmarket underground shopping mall s...
5     came visit library good mix restaurants fast f...
6     seoul famous markets relaxed shopping experien...
7     mall tours perfect place also library gangnam ...
8     located busiest territory gangnam district mal...
9     staying near visited mall frequently shopping ...
10    went mall wanted take photos library lots shop...
11    underground mall great place go winter months ...
12    given mall spread wide take much walking cover...
13          seoul gangnam gu area mall visit reasons
14    seoul definitely come coex spend day afternoon...
15    really largest underground mall asia despite l...
16    place carries cheap dollar store well foreign ...
17    prettiest library ever seen library located mi...
18    building coex area intercontinental hotel calt...
19    mall reached via subway stations samseong bong...
20    limited time visiting korea made sure visit co...
21    great place eat shop stayed hotel connected co...
22    big underground shopping mall lots things lot ...
23    coex huge mall lots shops restaurants entertai...
24    quite lot see full shops restaurants even beau...
25    seen movie beauty beast know scene beast takes...
26    place everything want spectacular tarfield lib...
```

# 데이터 전처리 2) stemming & lemmatizing

| | review | snowball_stem | lemmatize |
|---|---|---|---|
| 0 | lots typical mall including movie theatre aqua... | lot typic mall includ movi theatr aquarium lib... | lot typical mall including movie theatre aquar... |
| 1 | normally frequent several favorite stores mall... | normal frequent sever favorit store mall store... | normally frequent several favorite store mall ... |
| 2 | library huge nice almost books korean unless k... | librari huge nice almost book korean unless kn... | library huge nice almost book korean unless kn... |
| 3 | walk least one saturday month shop brands hyun... | walk least one saturday month shop brand hyund... | walk least one saturday month shop brand hyund... |
| 4 | coex mall upmarket underground shopping mall s... | coex mall upmarket underground shop mall sever... | coex mall upmarket underground shopping mall s... |
| 5 | came visit library good mix restaurants fast f... | came visit librari good mix restaur fast food ... | came visit library good mix restaurant fast fo... |
| 6 | seoul famous markets relaxed shopping experien... | seoul famous market relax shop experi coex muc... | seoul famous market relaxed shopping experienc... |
| 7 | mall tours perfect place also library gangnam ... | mall tour perfect place also librari gangnam s... | mall tour perfect place also library gangnam s... |
| 8 | located busiest territory gangnam district mal... | locat busiest territori gangnam district mall ... | located busiest territory gangnam district mal... |
| 9 | staying near visited mall frequently shopping ... | stay near visit mall frequent shop dine great ... | staying near visited mall frequently shopping ... |
| 10 | went mall wanted take photos library lots shop... | went mall want take photo librari lot shop pha... | went mall wanted take photo library lot shop p... |
| 11 | underground mall great place go winter months ... | underground mall great place go winter month p... | underground mall great place go winter month p... |
| 12 | given mall spread wide take much walking cover... | given mall spread wide take much walk cover gr... | given mall spread wide take much walking cover... |
| 13 | seoul gangnam gu area mall visit reasons | seoul gangnam gu area mall visit reason | seoul gangnam gu area mall visit reason |
| 14 | seoul definitely come coex spend day afternoon... | seoul definit come coex spend day afternoon gr... | seoul definitely come coex spend day afternoon... |
| 15 | really largest underground mall asia despite l... | realli largest underground mall asia despit la... | really largest underground mall asia despite l... |
| 16 | place carries cheap dollar store well foreign ... | place carri cheap dollar store well foreign ko... | place carry cheap dollar store well foreign ko... |
| 17 | prettiest library ever seen library located mi | prettiest librari ever seen librari locat midd | prettiest library ever seen library located mi |

**stemming** : flying -> fli, flies -> fli (어간 추출)
**lemmatizing** : flying -> fly, flies -> fly (품사에 따라 사전형 추출)

3

# Sentiment Analysis (with VADER)

# 감성 분석 with VADER

## VADER란?

Python의 감성 분석 패키지로,
"텍스트"의 감정을 나타내는 단어에 점수를 부여하여
Positive, neutral, negative, compound score로 점수화.

1. **positive sentiment:** `compound` score >= 0.05
2. **neutral sentiment:** ( `compound` score > -0.05) and ( `compound` score < 0.05)
3. **negative sentiment:** `compound` score <= -0.05

compound score의 값을 기준으로 (0.05, -0.05)
텍스트를 positive/neutral/negative로 구분

# 감성 분석 with VADER

In [12]:

```
def analyze_sentiment(df):
    sentiments = []
    label=[]
    analyser = SentimentIntensityAnalyzer()

    for i in
        line
        sent

        #긍
        if se

        elif
            la
        else
            la

        sent
        df[['

    return
```

compound score에 따른 라벨링

| | review | rating | neg | pos | neu | compound | label |
|---|---|---|---|---|---|---|---|
| 0 | great look feel stepped back century novelty b... | 4.0 | 0.000 | 0.301 | 0.699 | 0.8481 | Positive |
| 1 | bad lot see fairly packed tourists even rainy ... | 2.0 | 0.298 | 0.205 | 0.497 | -0.1280 | Negative |
| 2 | houses look great nice walk around absolutely ... | 3.0 | 0.000 | 0.496 | 0.504 | 0.7845 | Positive |
| 3 | people housing area resident time business ren... | 3.0 | 0.000 | 0.145 | 0.855 | 0.5994 | Positive |
| 4 | bukchon hanok village situated middle modern p... | 3.0 | 0.000 | 0.272 | 0.728 | 0.7964 | Positive |
| 5 | really enjoyed walk around hilly neighborhood ... | 4.0 | 0.000 | 0.366 | 0.634 | 0.8858 | Positive |
| 6 | beautiful area hill small alleys many still in... | 5.0 | 0.000 | 0.326 | 0.674 | 0.8910 | Positive |
| 7 | like nice relaxing quite walk around outside l... | 5.0 | 0.000 | 0.548 | 0.452 | 0.8176 | Positive |
| 8 | nice romantic local korean village walk korean... | 5.0 | 0.000 | 0.444 | 0.556 | 0.8625 | Positive |
| 9 | lucky enough walk streets less crowded ones co... | 5.0 | 0.000 | 0.128 | 0.872 | 0.4215 | Positive |
| 10 | like transported back time definitely must obs... | 5.0 | 0.201 | 0.327 | 0.472 | 0.4939 | Positive |
| 11 | place good visit great view peace cos resident... | 5.0 | 0.000 | 0.571 | 0.429 | 0.9231 | Positive |
| 12 | place great see old homes however high tech do... | 5.0 | 0.000 | 0.194 | 0.806 | 0.6249 | Positive |
| 13 | minutes away gyeongbokgung palace changdeokung... | 3.0 | 0.192 | 0.123 | 0.685 | -0.2500 | Negative |
| 14 | glad accompanied guide explained background re... | 4.0 | 0.143 | 0.251 | 0.606 | 0.3612 | Positive |

# 감성 분석(VADER 점수 + Rating 점수)

**Vader 점수에 따라 나온 라벨링과, rating(1~5점 리뷰 점수)에 따른 라벨링**
모두 유효한 변수라고 판단!

# 긍정적인 리뷰

```
result=pd.concat([coex_p,ddm_p,ddp_p,hong_p,loto_p,lowo_p,my_p,nam_p,ns_p])
result
#10,301개
```

| | review | rating | label | sent |
|---|---|---|---|---|
| 1 | normally frequent several favorite stores mall... | 5.0 | Positive | Positive |
| 3 | walk least one saturday month shop brands hyun... | 5.0 | Positive | Positive |
| 4 | coex mall upmarket underground shopping mall s... | 4.0 | Positive | Positive |
| 6 | seoul famous markets relaxed shopping experien... | 5.0 | Positive | Positive |
| 7 | mall tours perfect place also library gangnam ... | 4.0 | Positive | Positive |
| 8 | located busiest territory gangnam district mal... | 4.0 | Positive | Positive |
| 9 | staying near visited mall frequently shopping ... | 4.0 | Positive | Positive |
| 10 | went mall wanted take photos library lots shop... | 4.0 | Positive | Positive |
| 14 | seoul definitely come coex spend day afternoon... | 4.0 | Positive | Positive |
| 15 | really largest underground mall asia despite l... | 5.0 | Positive | Positive |
| 16 | place carries cheap dollar store well foreign ... | 5.0 | Positive | Positive |
| 17 | prettiest library ever seen library located mi... | 5.0 | Positive | Positive |
| 18 | building coex area intercontinental hotel calt... | 4.0 | Positive | Positive |
| 19 | mall reached via subway stations samseong bong... | 4.0 | Positive | Positive |
| 20 | limited time visiting korea made sure visit co... | 5.0 | Positive | Positive |
| 21 | great place eat shop stayed hotel connected co... | 5.0 | Positive | Positive |
| 22 | big underground shopping mall lots things lot ... | 4.0 | Positive | Positive |
| 23 | coex huge mall lots shops restaurants entertai... | 4.0 | Positive | Positive |
| 24 | quite lot see full shops restaurants even beau... | 4.0 | Positive | Positive |

모든 데이터에 대해
Vader → P 라벨링
rating → P 라벨링
된 데이터들 합침
→ 총 10,301개의 리뷰

# 부정적인 리뷰

```
result=pd.concat([coex_n,ddm_n,ddp_n,hong_n,loto_n,lowo_n,my_n,nam_n,ns_n,british,eiffel,ts,tr,lou,guell])
result
#9,486개
```

| | review | rating | label | sent |
|---|---|---|---|---|
| 144 | frustrating way built quite mess really shops ... | 2 | Negative | Negative |
| 159 | hell find anything rhyme reason laid way thoug... | 2 | Negative | Negative |
| 207 | bother visit mall pure waste time shops mall f... | 1 | Negative | Negative |
| 210 | sells branded stuff things really expensive ma... | 2 | Negative | Negative |
| 227 | right coex mall undergoing massive renovation ... | 2 | Negative | Negative |
| 233 | might well closed miles kilometers boarded sho... | 1 | Negative | Negative |
| 237 | standard mall lousy dull came conference | 1 | Negative | Negative |
| 239 | decidied take advantage rainy day visiting coe... | 1 | Negative | Negative |
| 242 | known renovation would avoided place walkathon... | 1 | Negative | Negative |
| 289 | vast mean physicist complex seems expand outwa... | 2 | Negative | Negative |
| 302 | five day vacation seoul went instead anything ... | 2 | Negative | Negative |
| 310 | five day vacation seoul went instead anything ... | 2 | Negative | Negative |
| 25 | horrible shopping experience like visited plac... | 1 | Negative | Negative |
| 125 | really disappointed market food left baking su... | 1 | Negative | Negative |
| 146 | absolutely nothing unique market sells fake br... | 2 | Negative | Negative |
| 162 | dongdaemun market like many similar markets as... | 2 | Negative | Negative |
| 176 | afraid nothing good say area markets seemed pr... | 2 | Negative | Negative |
| 54 | remember monday closed one told even tripadvis... | 1 | Negative | Negative |
| 235 | attention grabbing building little utility lit... | 2 | Negative | Negative |
| 288 | nothing special place pay extra special exhibi... | 2 | Negative | Negative |

모든 데이터에 대해
Vader → N 라벨링
rating → N 라벨링
된 데이터들 합침
→ 총 9,486개의 리뷰

# 긍정, 부정 리뷰 각 5,000개씩 랜덤 추출

모델에 긍정, 부정 리뷰를 같은 비율로 넣어서 학습시키기 위해
긍정, 부정 리뷰에서 각각 5,000개씩 랜덤 추출

```
import random
random.seed(999)
s=random.sample(li_review,5000)
my_dict={'review':s,'label':'Positive'}
random_positive=pd.DataFrame(my_dict)
random_positive.head()
```

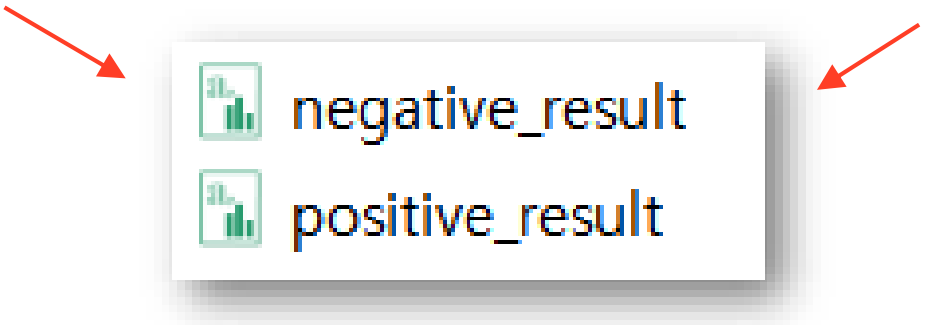
| | review | label |
|---|---|---|
| 0 | literally right next train station hongdae off... | Positive |
| 1 | get top climb metres | Positive |
| 2 | best place get amazing view city well worth wa... | Positive |
| 3 | opens late offers amazing view city mapped end... | Positive |
| 4 | energy love going market place truly massive e... | Positive |

```
import random
random.seed(999)
s=random.sample(li_review,5000)
my_dict={'review':s,'label':'Negative'}
random_negative=pd.DataFrame(my_dict)
random_negative.head()
```

| | review | label |
|---|---|---|
| 0 | book eiffel tower anything special boyfriend g... | Negative |
| 1 | beautiful architecture way many people hot plu... | Negative |
| 2 | main official residence thai royal family larg... | Negative |
| 3 | scope size art louvre amazing however tourists... | Negative |
| 4 | waited queue hour get tickets another mins go ... | Negative |

```
len(random_negative)
```

```
5000
```

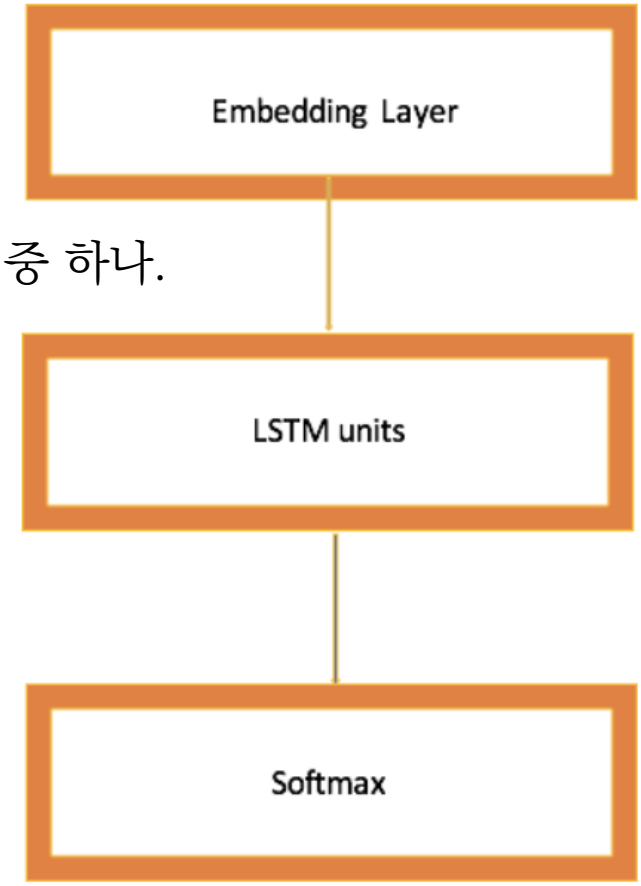


4

# Deep Learning (LSTM)

# LSTM이란?

자연어처리에 사용되는 딥러닝 패키지로,
인공신경망의 여러 종류(RNN, LSTM, Bi-LSTM) 중 하나.
특히 긴 문장에서
RNN 학습 능력이 떨어지는 것을 보완.




LSTM network

# LSTM vs Bi-LSTM

## LSTM 정확도

```
pos_cnt, neg_cnt, pos_correct, neg_correct = 0, 0, 0, 0
for x in range(len(X1_test)):

    result = model.predict(X1_test[x].reshape(1,X1_test.shape[1]),batch_size=1,verbose = 2)[0]

    if np.argmax(result) == np.argmax(Y1_test[x]):
        if np.argmax(Y1_test[x]) == 0:
            neg_correct += 1
        else:
            pos_correct += 1

    if np.argmax(Y1_test[x]) == 0:
        neg_cnt += 1
    else:
        pos_cnt += 1



print("pos_acc", pos_correct/pos_cnt*100, "%")
print("neg_acc", neg_correct/neg_cnt*100, "%")
```

```
pos_acc 96.4622641509434 %
neg_acc 95.84690553745928 %
```

pos_acc 96.4622641509434 %
neg_acc 95.84690553745928 %

## Bi-LSTM 정확도

```
pos_cnt, neg_cnt, pos_correct, neg_correct = 0, 0, 0, 0
for x in range(len(X1_test)):

    result = model.predict(X1_test[x].reshape(1,X1_test.shape[1]),batch_size=1,verbose = 2)[0]

    if np.argmax(result) == np.argmax(Y1_test[x]):
        if np.argmax(Y1_test[x]) == 0:
            neg_correct += 1
        else:
            pos_correct += 1

    if np.argmax(Y1_test[x]) == 0:
        neg_cnt += 1
    else:
        pos_cnt += 1

print("pos_acc", pos_correct/pos_cnt*100, "%")
print("neg_acc", neg_correct/neg_cnt*100, "%")
```

```
pos_acc 97.64150943396226 %
neg_acc 90.87947882736157 %
```

pos_acc 97.64150943396226 %
neg_acc 90.87947882736157 %

# Word embedding

```
from keras.preprocessing.text import Tokenizer
t = Tokenizer()
t.fit_on_texts(df['review'])
vocab_size = len(t.word_index) + 1

X_encoded = t.texts_to_sequences(df['review'])
max_length=max(len(l) for l in X_encoded)

from keras.preprocessing.sequence import pad_sequences
X_train=pad_sequences(X_encoded, maxlen=max_length, padding='post')
print(X_train)
```

```
Using TensorFlow backend.
```

```
[[ 557  208  247 ...    0    0    0]
 [   4   53  751 ...    0    0    0]
 [  72    1    4 ...    0    0    0]
 ...
 [  54   54   54 ...    0    0    0]
 [  82 1665  232 ...    0    0    0]
 [  13    1  135 ...    0    0    0]]
```

```
import numpy as np
embedding_dict = dict()
f = open('C:\\Users\\USER\\Documents\\GH\\내부플젝\\glove.6B\\glove.6B.100d.txt', encoding="utf8")
# 예를 들어 윈도우 바탕화면에서 실습한 저자의 경우
# f = open(r'C:\Users\USER\Desktop\glove.6B.100d.txt', encoding="utf8") 였습니다.
for line in f:
    word_vector = line.split()
    word = word_vector[0]
    word_vector_arr = np.asarray(word_vector[1:], dtype='float32') # 100개의 값을 가지는 array로 변환
    embedding_dict[word] = word_vector_arr
f.close()
print('%s개의 Embedding vector가 있습니다.' % len(embedding_dict))
```

# LSTM Modeling

```
model = Sequential()
model.add(embedding_layer)
model.add(LSTM(128, dropout=0.5, return_sequences=False
               , input_shape=(max_length,embed_dim)))
model.add(Dense(2,activation='softmax')) #예측하고자 하는 수에 따라 숫자
model.compile(loss = 'categorical_crossentropy', optimizer='adam',metrics = ['accuracy'])
print(model.summary())
```

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 251, 100)          1467700
_________________________________________________________________
lstm_2 (LSTM)                (None, 128)               117248
_________________________________________________________________
dense_2 (Dense)              (None, 2)                 258
=================================================================
Total params: 1,585,206
Trainable params: 117,506
Non-trainable params: 1,467,700
_________________________________________________________________
None
```

# Testing

```
pos_cnt, neg_cnt, pos_correct, neg_correct = 0, 0, 0, 0
for x in range(len(X1_test)):

    result = model.predict(X1_test[x].reshape(1,X1_test.shape[1]),batch_size=1,verbose = 2)[0]

    if np.argmax(result) == np.argmax(Y1_test[x]):
        if np.argmax(Y1_test[x]) == 0:
            neg_correct += 1
        else:
            pos_correct += 1

    if np.argmax(Y1_test[x]) == 0:
        neg_cnt += 1
    else:
        pos_cnt += 1



print("pos_acc", pos_correct/pos_cnt*100, "%")
print("neg_acc", neg_correct/neg_cnt*100, "%")
```

```
pos_acc 96.4622641509434 %
neg_acc 95.84690553745928 %
```

pos_acc 96.4622641509434 %
neg_acc 95.84690553745928 %

# Result

| | review | lstm_label |
|---|---|---|
| 0 | great look feel stepped back century novelty b... | 1 |
| 1 | bad lot see fairly packed tourist even rainy d... | 0 |
| 2 | house look great nice walk around absolutely p... | 1 |
| 3 | people housing area resident time business ren... | 1 |
| 4 | bukchon hanok village situated middle modern p... | |
| 5 | really enjoyed walk around hilly neighborhood ... | |
| 6 | beautiful area hill small alley many still inh... | |
| 7 | like nice relaxing quite walk around outside l... | |
| 8 | nice romantic local korean village walk korean... | |
| 9 | lucky enough walk street le crowded one come a... | 0 |
| 10 | like transported back time definitely must obs... | 0 |
| 11 | place good visit great view peace co resident ... | 1 |
| 12 | place great see old home however high tech doo... | 1 |
| 13 | minute away gyeongbokgung palace changdeokung ... | 0 |
| 14 | glad accompanied guide explained background re... | 0 |

relemmatized_cheong

relemmatized_chon

relemmatized_gung

relemmatized_insadong

5

# Wordcloud & Topic Modeling

# Wordcloud란?

텍스트를 시각화하는 툴로, 빈도수가 많은 단어를 크게 보여준다.

**종로구 관광지 output data (object data)**

경복궁 (P:1,452/N:1,098)
인사동거리 (P:1,117/N:53)
북촌한옥마을 (P:2,124/N:425)
청계천 (P:1,349/N:471)

부정적인 리뷰만 추출,
해당 관광지의
관광객들의 불만사항을
알아보자!

긍정적인 리뷰만 추출,
해당 관광지의
매력을 알아보자!

# 북촌한옥마을 wordcloud


긍정 리뷰
wordcloud


부정 리뷰
wordcloud

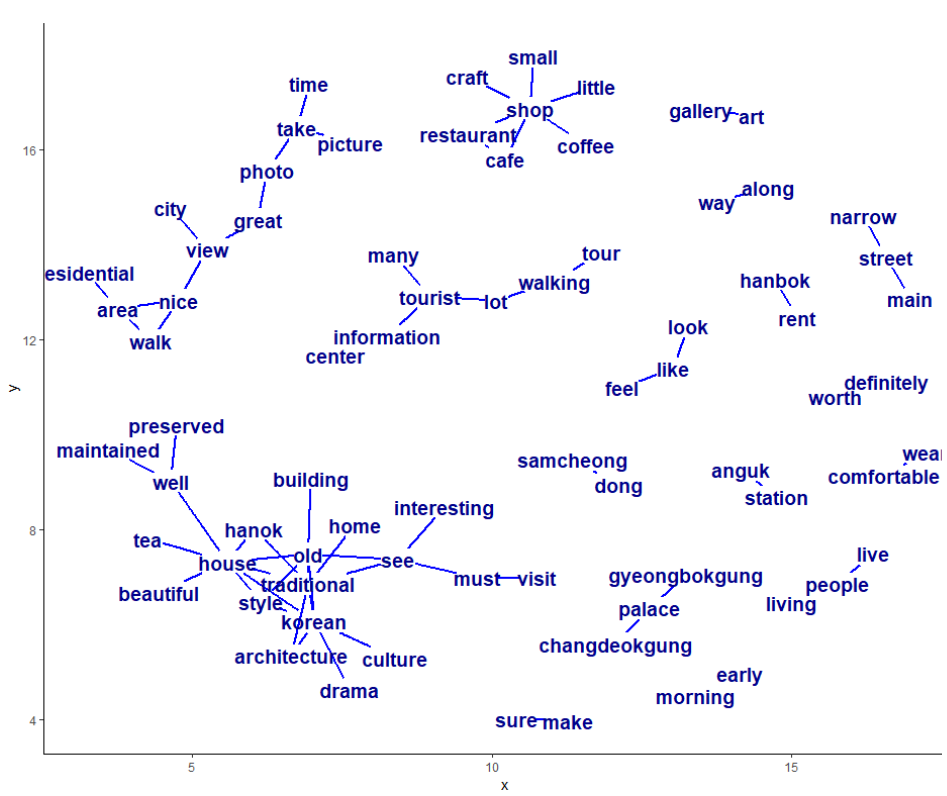
# 북촌한옥마을 wordcloud


부정단어 제외
긍정 리뷰
wordcloud

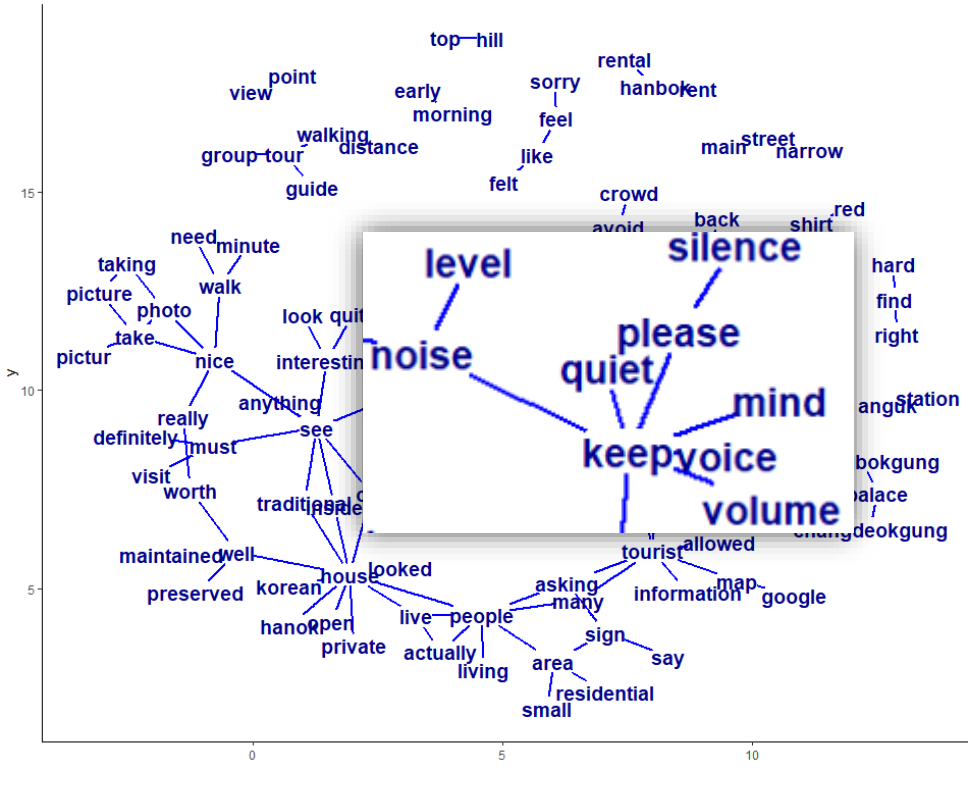

긍정단어 제외
부정 리뷰
wordcloud

# 북촌한옥마을 Bi-gram

문장에서 사용된 전/후 단어를 2개씩 조합하여 분석, 단어 간의 관계 파악




긍정 리뷰
Bi-gram




부정 리뷰
Bi-gram

# Object data 리뷰에 대한 토픽 모델링

## Topic modeling이란?

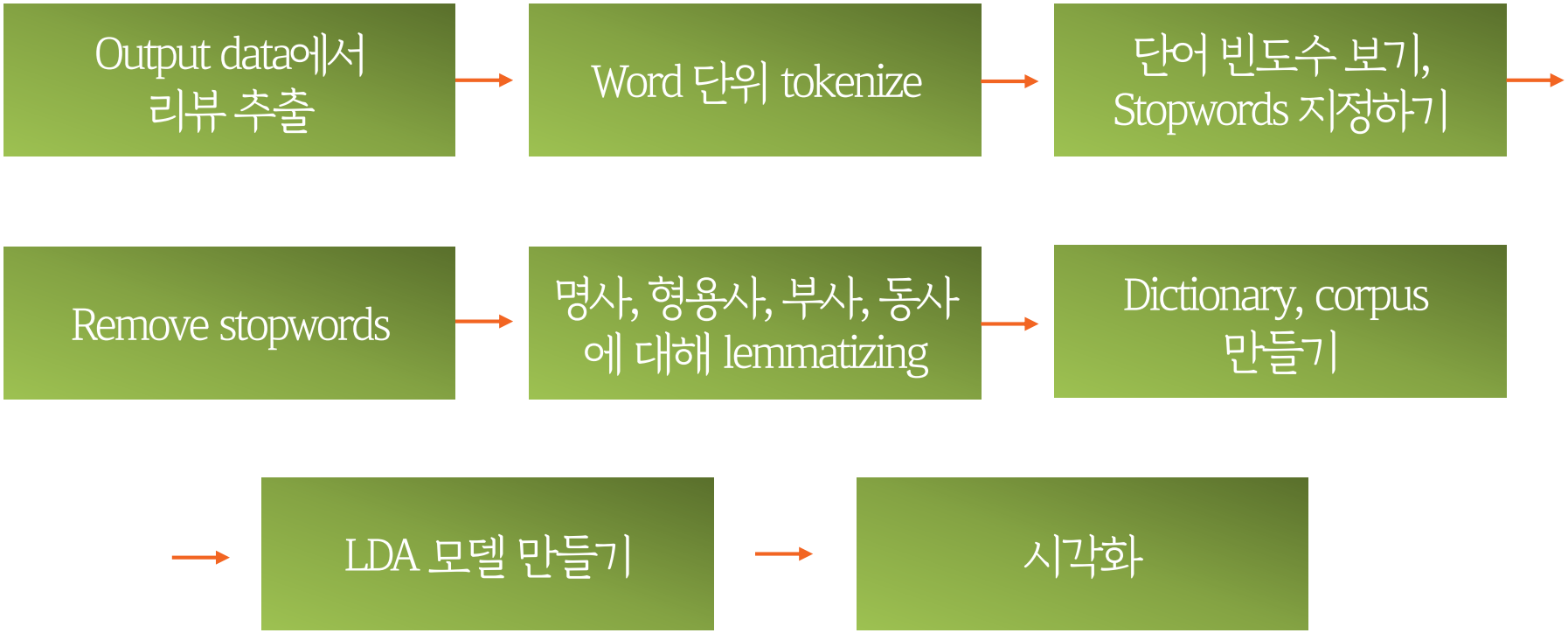
많은 양의 텍스트 정보에서 숨겨진 주제를 추출하는 기법.
LDA가 토픽 모델링의 가장 유명한 알고리즘이며,
Python의 Gensim 패키지를 활용하여 실행 가능하다.

종로구 관광지 output data
(object data)

→

관광지 리뷰 대상으로,
해당 관광지의
숨겨진 주제를 알아보자!

# 토픽 모델링 과정


Output data에서 리뷰 추출
Word 단위 tokenize
단어 빈도수 보기, Stopwords 지정하기
Remove stopwords
명사, 형용사, 부사, 동사에 대해 lemmatizing
Dictionary, corpus 만들기
LDA 모델 만들기
시각화

# 종로구 토픽 모델링

```
#tokenize sentence to words
def sent_to_words(sentences):
    for sentence in sentences:
        yield(gensim.utils.simple_preprocess(str(sentence), deacc=True))  # deacc=True removes punctuations

data_words = list(sent_to_words(n_gung_list))

print(data_words[:1])
```

```
[['timing', 'critical', 'want', 'see', 'changing', 'guard', 'tour', 'guide', 'vip', 'travel', 'susan', 'got', 'us', 'crowded', 'morning', 'get', 'good', 'standing', 'spot', 'ceremony', 'tour',
hru', 'concierge', 'car', 'really', 'worth', 'every', 'single', 'showed', 'us', 'invented', 'heating', 'system', 'buildings', 'heated', 'floors', 'pretty', 'smart', 'miss', 'pretty', 'garden
s', 'get', 'ceremony', 'viewing', 'buildings', 'hope', 'makes', 'visit', 'easier']]
```

→ Word 단위 tokenize

```
# WordCount _ word의 개수 세기
word_count = collections.Counter(nnp_list_1d)
result = word_count.most_common(1000)
print(len(word_count))
print(result)
```

```
4056
[('palace', 1917), ('visit', 474), ('see', 469), ('changing', 392), ('place', 391), ('tour', 348), ('around', 335), ('time', 327), ('free', 305), ('history', 299)
t', 240), ('buildings', 229), ('take', 223), ('ceremony', 220), ('grounds', 217), ('many', 213), ('hanbok', 212), ('seoul', 208), ('palaces', 201), ('day', 2
5), ('great', 176), ('walk', 176), ('traditional', 174), ('museum', 169), ('must', 167), ('also', 166), ('well', 165), ('entrance', 163), ('visited', 161), ('mu
7), ('like', 156), ('hours', 152), ('huge', 151), ('english', 145), ('really', 138), ('interesting', 135), ('walking', 129), ('inside', 122), ('good', 121), ('lot',
erience', 108), ('area', 106), ('big', 104), ('city', 103), ('photos', 102), ('pictures', 99), ('gate', 98), ('recommend', 97), ('rent', 96), ('us', 95), ('back'
88), ('visiting', 85), ('fee', 85), ('crowded', 84), ('taking', 84), ('wearing', 84), ('guided', 83), ('make', 83), ('quite', 82), ('entry', 81), ('dynasty', 81),
d', 78), ('still', 78), ('dress', 78), ('sure', 77), ('historical', 77), ('joseon', 76), ('want', 73), ('pm', 72), ('outside', 71), ('first', 71), ('need', 68), ('look',
ar', 66), ('going', 66), ('took', 65), ('tours', 63), ('definitely', 62), ('feel', 62), ('national', 61), ('amazing', 61), ('old', 60), ('spend', 58), ('able', 57), ('
56), ('visitors', 55), ('tourist', 55), ('building', 55), ('though', 54), ('closed', 54), ('think', 54), ('complex', 53), ('part', 52), ('dressed', 52), ('several',
```

→ 각 단어의 개수

# 종로구 토픽 모델링

```
#토픽모델링에 필요한 dictionary와 corpus 만들기

# Create Dictionary
id2word = corpora.Dictionary(data_lemmatized)

# Create Corpus
texts = data_lemmatized

# Term Document Frequency
corpus = [id2word.doc2bow(text) for text in texts]

# View
print(corpus[:1])
```

```
[[(0, 1), (1, 2), (2, 1), (3, 2), (4, 1), (5, 1), (6, 1), (7, 1), (8
```

→ Dictionary, corpus 만들기

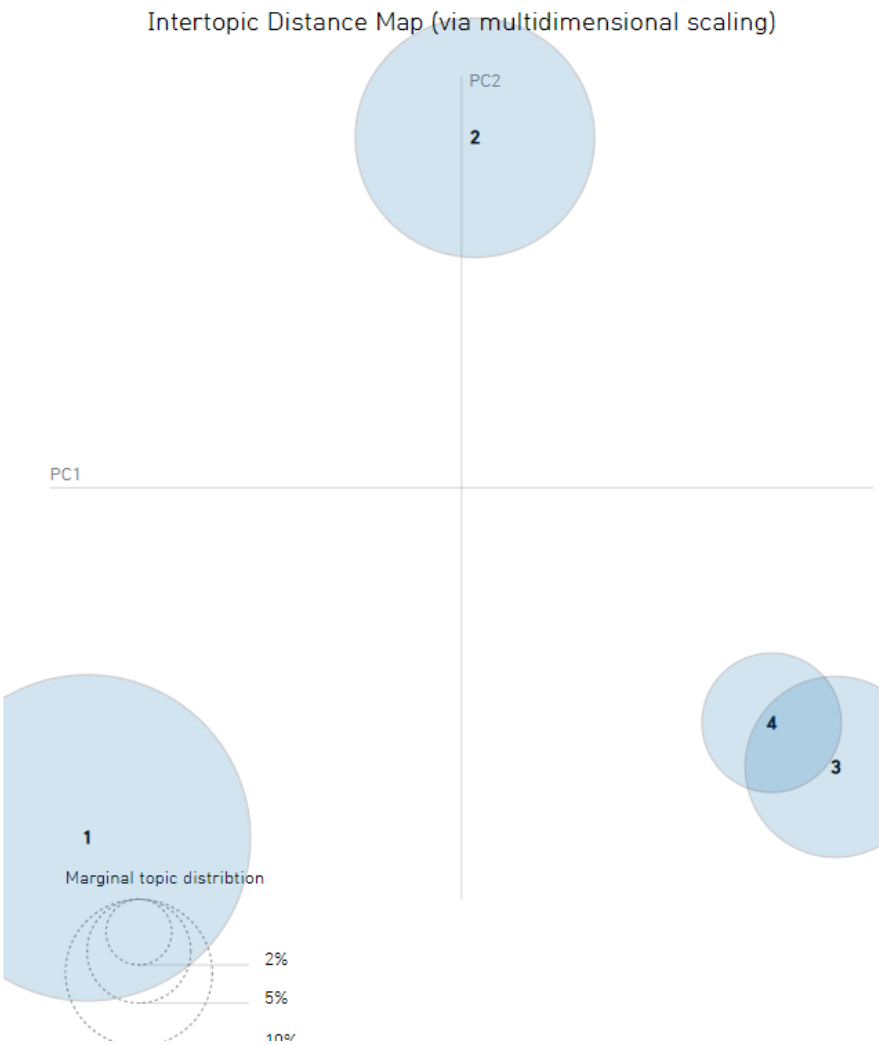
```
# Build LDA model
NUM_TOPICS=4
lda_model = gensim.models.ldamodel.LdaModel(corpus=corpus,
                                            id2word=id2word,
                                            num_topics=NUM_TOPICS,
                                            random_state=100,
                                            update_every=1,
                                            chunksize=100,
                                            passes=10,
                                            alpha='auto',
                                            per_word_topics=True)
```

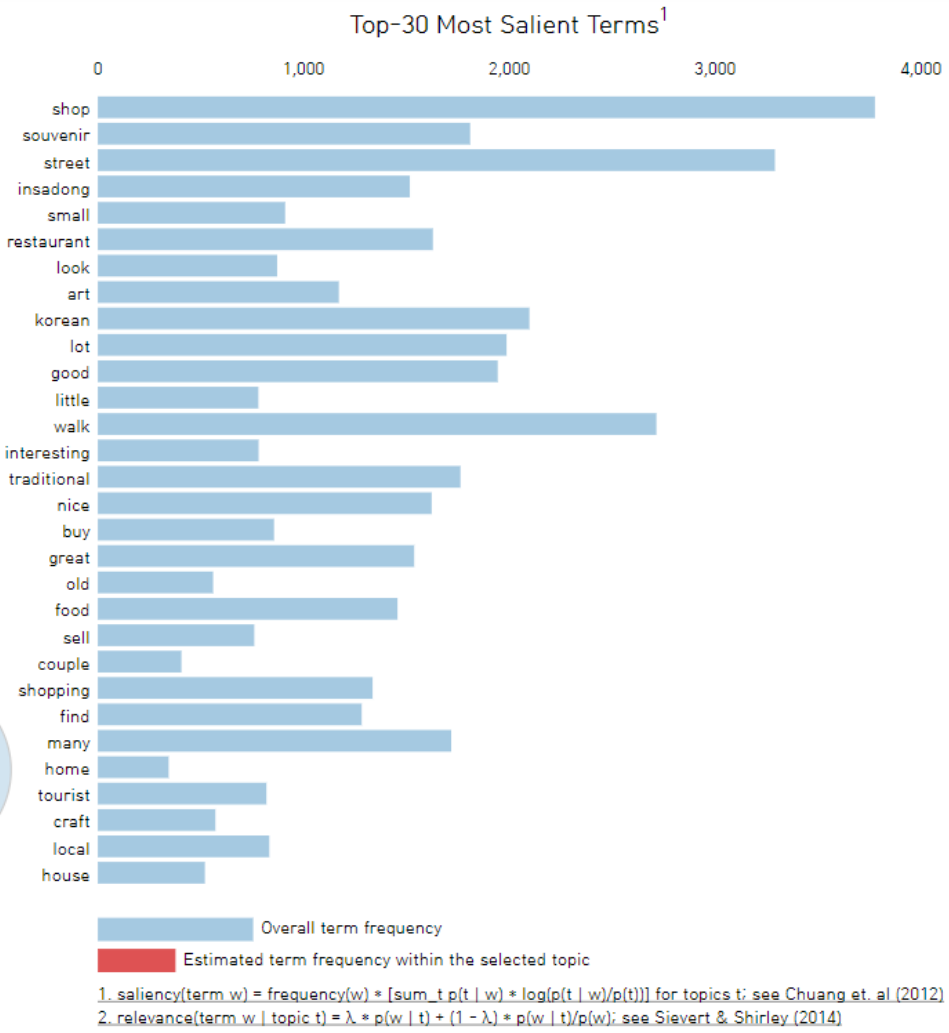
```
# Print the Keyword in the topics
#The weights reflect how important a keyword is to that topic.
pprint(lda_model.print_topics())
doc_lda = lda_model[corpus]
```

```
[(0,
  '0.038*"small" + 0.036*"look" + 0.031*"little" + 0.029*"interesting" + '
  '0.024*"old" + 0.015*"stuff" + 0.014*"bit" + 0.014*"still" + '
  '0.013*"touristy" + 0.012*"could"'),
 (1,
  '0.091*"shop" + 0.044*"souvenir" + 0.037*"insadong" + 0.034*"restaurant" + '
  '0.028*"art" + 0.022*"find" + 0.021*"buy" + 0.018*"sell" + 0.017*"tourist" + '
  '0.016*"local"'),
 (2,
  '0.029*"couple" + 0.024*"home" + 0.019*"cheonggyecheon" + 0.017*"high" + '
  '0.015*"center" + 0.014*"nothing" + 0.012*"dongdaemun" + 0.011*"bridge" + '
  '0.011*"kid" + 0.010*"life"'),
 (3,
  '0.043*"street" + 0.034*"walk" + 0.027*"korean" + 0.026*"lot" + 0.025*"good" '
  '+ 0.023*"traditional" + 0.022*"many" + 0.021*"nice" + 0.020*"great" + '
  '0.019*"food"')]
```

→ LDA 모델 만들기

# Result


Intertopic Distance Map (via multidimensional scaling)
PC2
2
PC1
4
3
1
Marginal topic distribtion
2%
5%
10%


Top-30 Most Salient Terms[1]
0
1,000
2,000
3,000
4,000
shop
souvenir
street
insadong
small
restaurant
look
art
korean
lot
good
little
walk
interesting
traditional
nice
buy
great
old
food
sell
couple
shopping
find
many
home
tourist
craft
local
house
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 − λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

# Result

| | Topic # 01 | Topic # 02 | Topic # 03 | Topic # 04 |
|---|---|---|---|---|
| 0 | small | shop | couple | street |
| 1 | look | souvenir | home | walk |
| 2 | little | insadong | cheonggyecheon | korean |
| 3 | interesting | restaurant | high | lot |
| 4 | old | art | center | good |
| 5 | stuff | find | nothing | traditional |
| 6 | bit | buy | dongdaemun | many |
| 7 | still | sell | bridge | nice |
| 8 | touristy | tourist | kid | great |
| 9 | could | local | life | food |
| 10 | sit | craft | green | shopping |
| 11 | clean | house | live | tea |
| 12 | kind | cafe | special | enjoy |
| 13 | handicraft | alley | open | take |
| 14 | modern | main | fish | people |
| 15 | building | eat | music | store |
| 16 | road | spend | change | love |
| 17 | work | gallery | point | night |
| 18 | district | way | cross | go |
| 19 | line | quite | probably | item |

## 서울 종로구 관광지 소개 페이지

Instagram

경복궁 #한복 #즐거움가득 #살짝지루해
인사동거리 #쇼핑몰 #걷기좋은곳 #북적북적
북촌한옥마을 #전통한국멋 #기념품많아요 #다닐땐조용히
청계천 #걷기좋은곳 #데이트로딱 #벌레가좀…

# Reference


〈2018 외래관광객 실태조사 최종보고서〉, 한국관광공사)
https://kto.visitkorea.or.kr/kor/notice/data/statis/tstatus/forstatus/board/view.kto?id=431236&isNotice=false&instanceId=295&rnum=4)

크롤링) https://github.com/Sentylic/Tripadvisor-Scraper

전처리) https://wikidocs.net/21707

감성분석)https://github.com/MOONJOOYOUNG/DataScience/blob/master/Sentiment%20Analysis/Sentiment%20Analysis.py

LSTM) https://machinelearningmastery.com/diagnose-overfitting-underfitting-lstm-models/
https://stackoverflow.com/questions/41816439/loss-function-in-lstm-neural-network
https://ratsgo.github.io/natural%20language%20processing/2017/03/09/rnnlstm/
https://www.kaggle.com/c/santander-product-recommendation/discussion/25802
https://towardsdatascience.com/tagged/lstm
https://towardsdatascience.com/understanding-lstm-and-its-quick-implementation-in-keras-for-sentiment-analysis-af410fd85b47
https://hackernoon.com/understanding-architecture-of-lstm-cell-from-scratch-with-code-8da40f0b71f4

워드클라우드, 바이그램) https://medium.com/myrealtrip-product/%EB%A7%88%EC%9D%B4%EB%A6%AC%EC%96%BC%ED%8A%B8%EB%A6%BD-%EC%97%AC%ED%96%89-%ED%9B%84%EA%B8%B0-%EB%8D%B0%EC%9D%B4%ED%84%B0-%EB%B6%84%EC%84%9D-be3f6c557ca2

토픽모델링) https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/
https://blog.naver.com/upennsolution/221437143732

# Thanks for listening!

자유로운 질문 환영합니다 ☺