parrot

Data Science

Toxic comment classification : 안소윤, 이수정, 정상희

Contents

R&R

Toxic dataset EDA

계획

- **R&R**

안소윤 : 발표자료 만들기

이수정 : 로그 정리

정상희 : 발표 진행

*전체적인 코드는 각자 돌려본 후 공유하는 방식으로 진행

Toxic dataset EDA

• Toxic dataset EDA

데이터 불러오기

```
1 import pandas as pd
2
3 train_data = pd.read_csv('/content/drive/MyDrive/Parrot_teamproject/train.csv')
4 test_data = pd.read_csv('/content/drive/MyDrive/Parrot_teamproject/test.csv')
5 test_labels = pd.read_csv('/content/drive/MyDrive/Parrot_teamproject/test_labels.csv')
```

• Toxic dataset EDA

```
#데이터 불러오기
train_data = pd.read_csv('/content/drive/MyDrive/Parrot_teamproject/train.csv')
train_data.head()
```

	id	comment_text	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	0000997932d777bf	Explanation\nWhy the edits made under my usern...	0	0	0	0	0	0
1	000103f0d9cfb60f	D'aww! He matches this background colour I'm s...	0	0	0	0	0	0
2	000113f07ec002fd	Hey man, I'm really not trying to edit war. It...	0	0	0	0	0	0
3	0001b41b1c6bb37e	"\nMore\nI can't make any real suggestions on ...	0	0	0	0	0	0
4	0001d958c54c6e35	You, sir, are my hero. Any chance you remember...	0	0	0	0	0	0

```
test= pd.read_csv('/content/drive/MyDrive/Parrot_teamproject/test.csv')
test.head()
```

	id	comment_text
0	00001cee341fdb12	Yo bitch Ja Rule is more succesful then you'll...
1	0000247867823ef7	== From RfC == \n\n The title is fine as it is...
2	00013b17ad220c46	" \n\n == Sources == \n\n * Zawe Ashton on Lap...
3	00017563c3f7919a	:If you have a look back at the source, the in...
4	00017695ad8997eb	I don't anonymously edit articles at all.

```
test_labels = pd.read_csv('/content/drive/MyDrive/Parrot_teamproject/test_labels.csv')
test_labels.head()    #value of -1 indicates it was not used for scoring
```

	id	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	00001cee341fdb12	-1	-1	-1	-1	-1	-1
1	0000247867823ef7	-1	-1	-1	-1	-1	-1
2	00013b17ad220c46	-1	-1	-1	-1	-1	-1
3	00017563c3f7919a	-1	-1	-1	-1	-1	-1
4	00017695ad8997eb	-1	-1	-1	-1	-1	-1

• Toxic dataset EDA

train에서 id열 제외

```
1 #id는 학습시킬 필요 없으므로 제외
2 train = train_data.drop(columns=['id'], axis=1)
3 train.head()
```

	comment_text	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	Explanation\nWhy the edits made under my usern...	0	0	0	0	0	0
1	D'aww! He matches this background colour I'm s...	0	0	0	0	0	0
2	Hey man, I'm really not trying to edit war. It...	0	0	0	0	0	0
3	"\nMore\nI can't make any real suggestions on ...	0	0	0	0	0	0
4	You, sir, are my hero. Any chance you remember...	0	0	0	0	0	0

• Toxic dataset EDA

결측치 확인

```
 1 train.info()   #결측치 없음
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 159571 entries, 0 to 159570
Data columns (total 7 columns):
 #   Column          Non-Null Count   Dtype
---  ------          --------------   -----
 0   comment_text    159571 non-null  object
 1   toxic           159571 non-null  int64
 2   severe_toxic    159571 non-null  int64
 3   obscene         159571 non-null  int64
 4   threat          159571 non-null  int64
 5   insult          159571 non-null  int64
 6   identity_hate   159571 non-null  int64
dtypes: int64(6), object(1)
memory usage: 8.5+ MB
```

• Toxic dataset EDA

Shape 및 train/test data 비율 확인

```
1 #shape 확인
2 train.shape , test.shape
```

```
((159571, 7), (153164, 2))
```

```
1 #train/test data 비율
2 sum = train.shape[0]+test.shape[0]
3 round(train.shape[0]*100/sum), round(test.shape[0]*100/sum)
```

```
(51, 49)
```

• Toxic dataset EDA

각 label의 sample 뽑아보기

```
labels = ['toxic', 'severe_toxic', 'obscene', 'threat', 'insult', 'identity_hate']

for label in labels:
  print("{} :".format(label.upper()))
  print(train.loc[train[label]==1, 'comment_text'].sample().values, "\n")
```

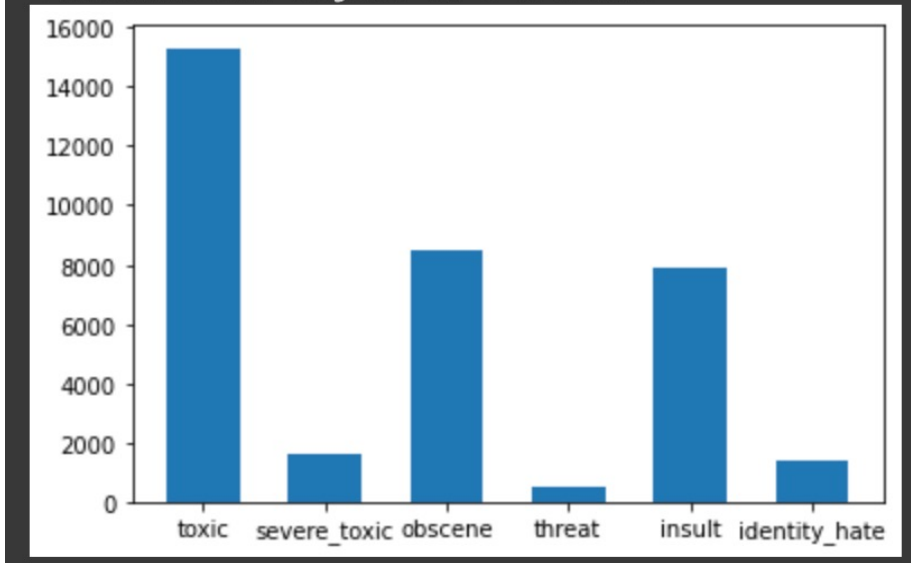
```
TOXIC :
['This guy is such a loser']

SEVERE_TOXIC :
['nigger dick shit \n\nYOU ARE A BIG NIGGER DICK SHIT']

OBSCENE :
["piece of shit. \n\nfuck your warning and fuck your mum. and gg I didn't sign this so u cant ban me as u don't know who wrote this sloppy ass

THREAT :
["let me tell you little man, a personal attack will be when I find you and beat the hell out of you.  Be very glad I don't know where you live

INSULT :
['Go fuck yourself Tbnotch I am on a public computer. Suck on it bitch face.']

IDENTITY_HATE :
["which one you faild the exams or the names? I've heard that bulgarian women don't wash...errr that's stinky smelling bad.."]
```

• Toxic dataset EDA

label별 분포

```
#카테고리별 분포
comments = ["toxic", "severe_toxic", "obscene", "threat", "insult", "identity_hate"]
comments_count = [train.toxic.sum(), train.severe_toxic.sum(),
         train.obscene.sum(), train.threat.sum(),
         train.insult.sum(), train.identity_hate.sum() ]

plt.bar(comments, comments_count, width=0.6)
```

```
<BarContainer object of 6 artists>
```



• Toxic dataset EDA

Mean값을 통해 각 label에 속할 확률 확인

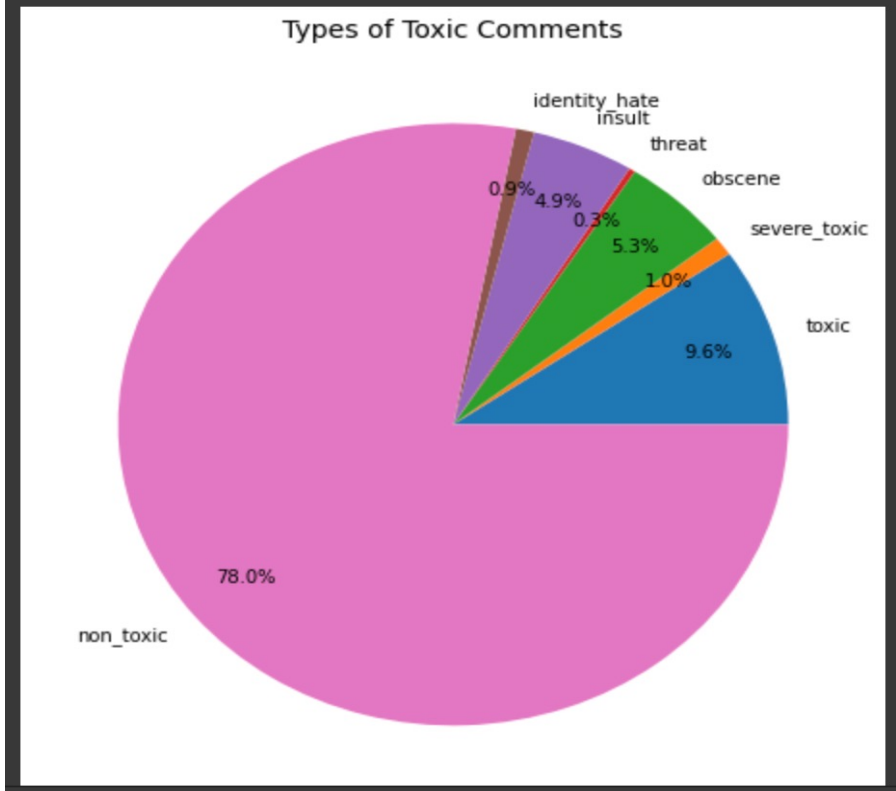
```
4 train_data.describe()
```

	toxic	severe_toxic	obscene	threat	insult	identity_hate
count	159571.000000	159571.000000	159571.000000	159571.000000	159571.000000	159571.000000
mean	0.095844	0.009996	0.052948	0.002996	0.049364	0.008805
std	0.294379	0.099477	0.223931	0.054650	0.216627	0.093420
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
75%	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

• Toxic dataset EDA

전체 train data의 label별 분포 그래프

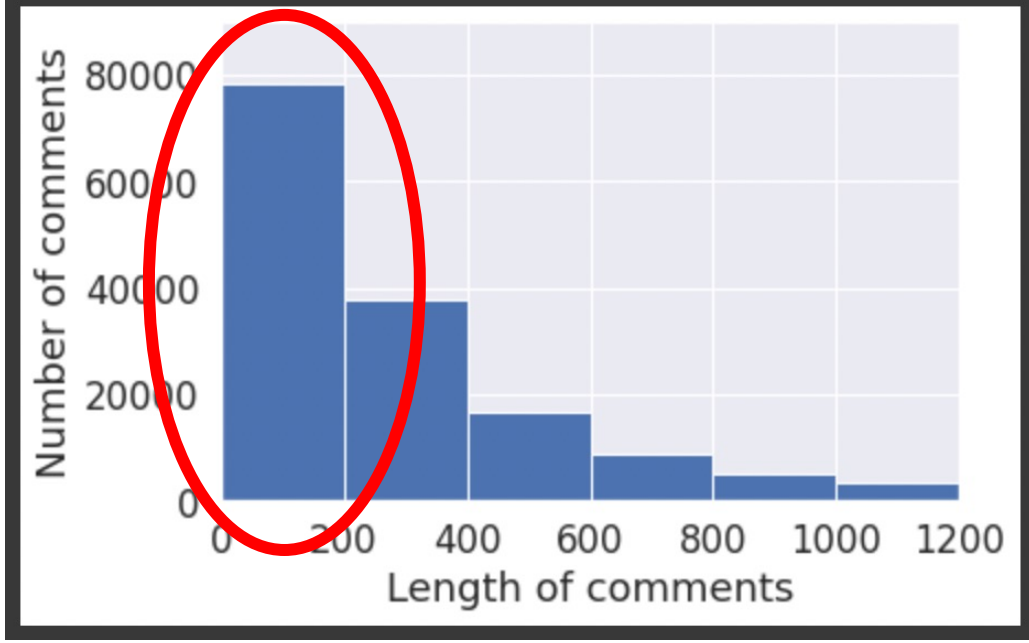
```python
import matplotlib.pyplot as plt

pie, ax = plt.subplots(figsize=[10,8])

plt.pie(x = distribution.values(), autopct="%.1f%%", labels = distribution.keys(), pctdistance = 0.8)
plt.title("Types of Toxic Comments", fontsize=14)
plt.show()
```



• Toxic dataset EDA

Comment별 문자열의 길이

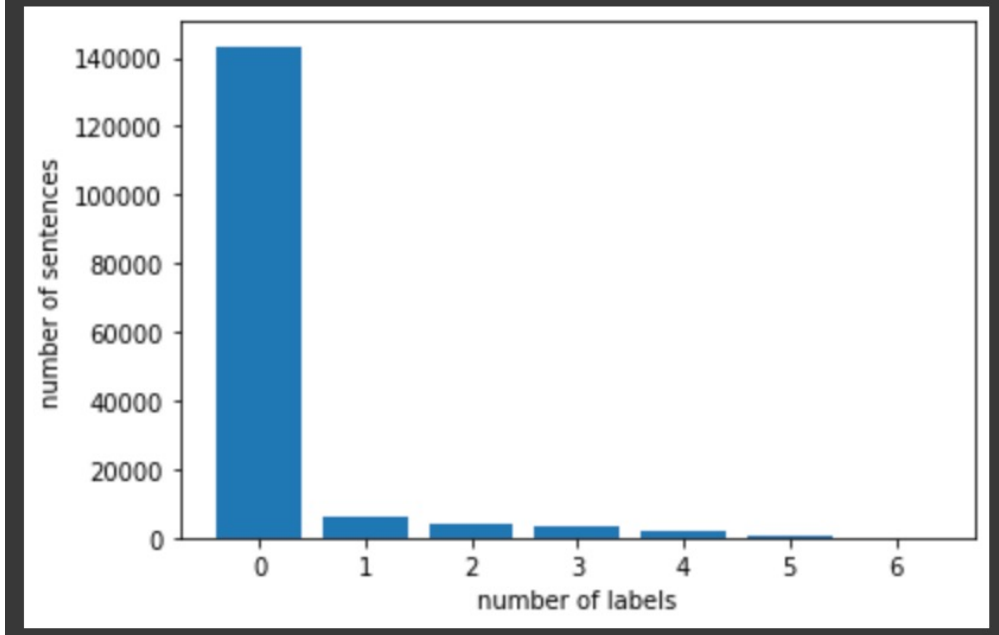
```
1 x = [len(comment[i]) for i in range(comment.shape[0])]
2
3 plt.hist(x, bins=[1, 200, 400, 600, 800, 1000, 1200])
4 plt.xlabel('Length of comments')
5 plt.ylabel('Number of comments')
6 plt.axis([0, 1200, 0, 90000])
7 plt.grid(True)
8 plt.show()
```



• Toxic dataset EDA

각 comment의 겹치는 label 개수 확인

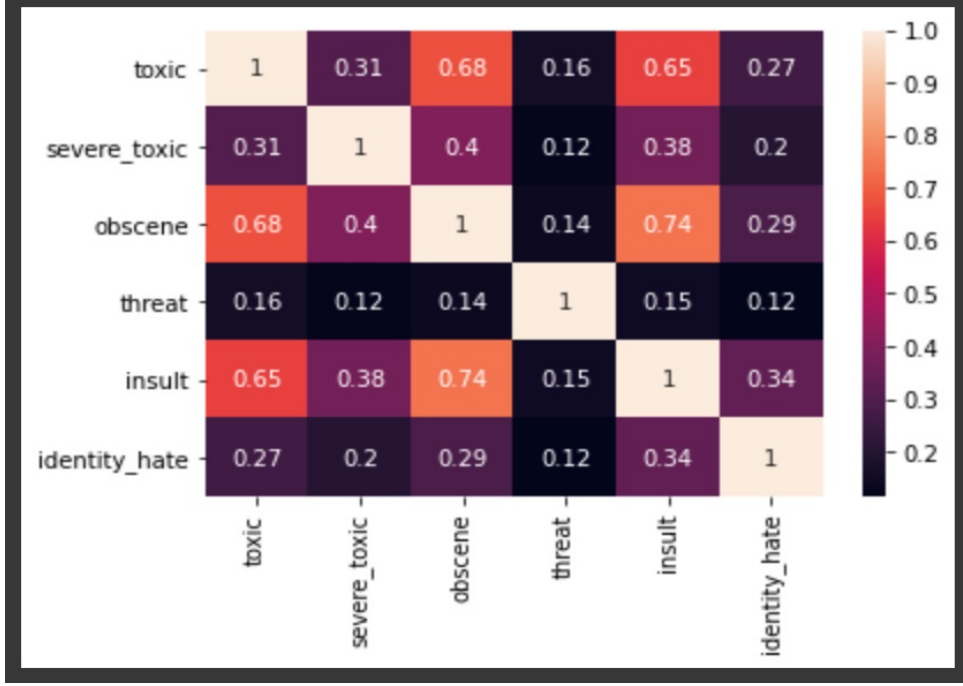
```
1 import numpy as np
2 #겹치는 라벨 개수 확인
3
4 plt.bar(np.arange(0, 7), train_data.iloc[:, 2:].sum(axis=1).value_counts().values)
5 plt.xlabel('number of labels')
6 plt.ylabel('number of sentences')
7 plt.show()
```



• Toxic dataset EDA

label별 corr

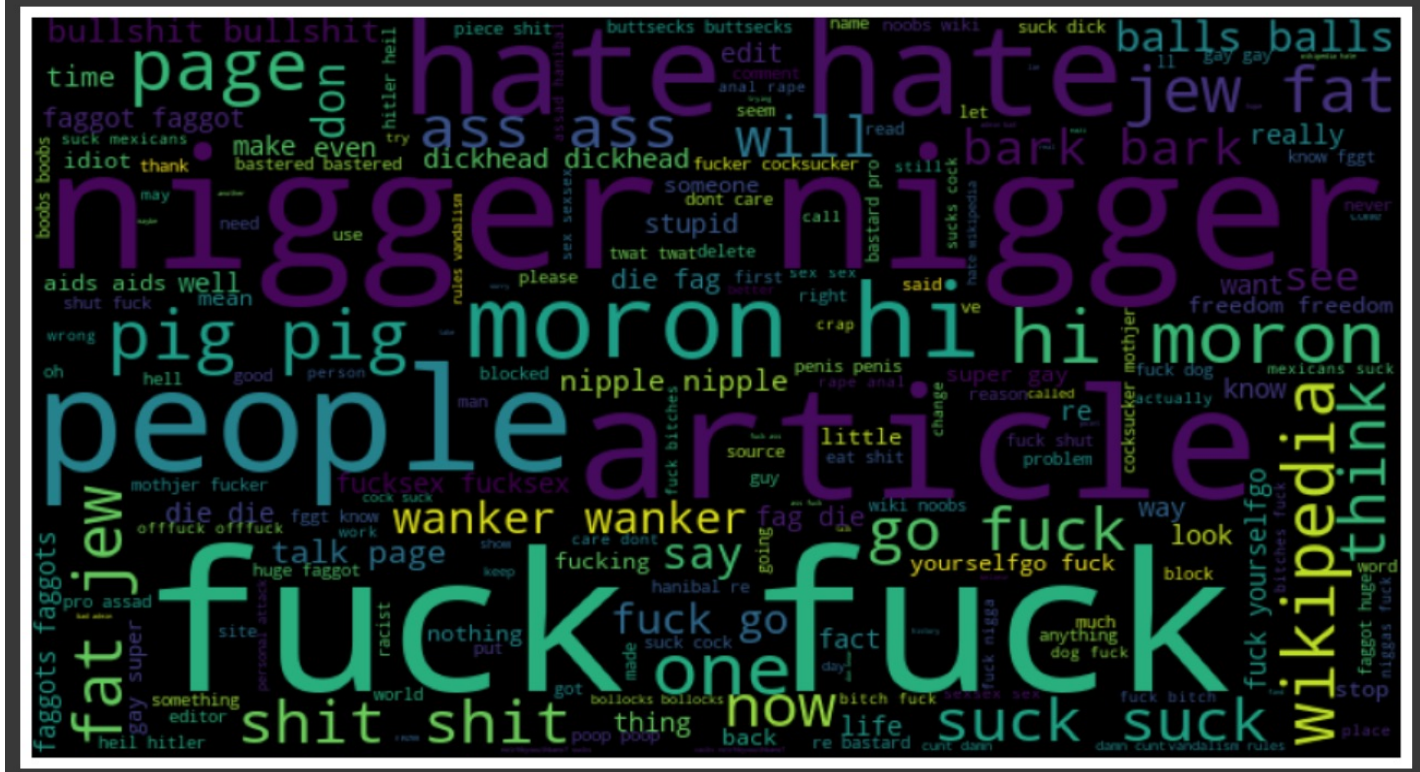
```python
#corr 분석
import seaborn as sns

target_data = train_data.drop(['id', 'comment_text'], axis=1)
corr_matrix = target_data.corr()
sns.heatmap(corr_matrix, annot=True)
plt.show()
```



• Toxic dataset EDA

Word cloud로 시각화

```
#많이 등장하는 단어(욕설) 확인
from wordcloud import WordCloud

cloud = WordCloud(width=700, height=400).generate(' '.join(target_data['comment_text'].astype(str)))
plt.figure(figsize=(15,13))
plt.imshow(cloud)
plt.axis('off')
plt.show()
```



- **계획**

- 일주일에 2번 줌 회의를 통해 코드 공유 및 방향성 결정

- 인터넷에 오픈된 코드/논문을 통해 모델 디벨롭

- 깃헙을 통해 개념정리 꾸준히 하기

감사합니다🙂