DATA CLEANSING TO CENSOR AND MASK PROFANITY ON TWITTER BY USING API

Ahmad Fadlan Amin

BINAR Academy

BINAR

BACKGROUND

As the emerging social platforms that provides everyone with freedom of speech, many people takes this for granted and abuse it to spread hate and derogatory words, including Twitter. This social platform infamous for having users that often spreads toxicity, and use profanities in most of tweets that are sent and making these tweets not safe for work (nsfw).

As a movement to build a more positive community, while maintaining people aspirations in sharing their thoughts and ideas, tweets that are contains negative connotations such as hate speech and abusive words needs to be cleansed and mask in order to minimize discrimination against marginalized community, prevent hate-spread and motivate people to use proper Bahasa Indonesia in the process.

RESEARCH METHOD

PYTHON	Using Pandas and Regex to data cleansing and inputting the source code into dhe Flask
FLASK AND SWAGGER UI	To create the API for data cleansing, which feature : • User input in the form of text and file • Output of cleansed text and file
SQLITE	To store the database and import raw data file
MATPLOTLIB	To provide visualization of analyzed data
GITHUB AND GIT	To document the making process of the API and data cleansing

CODE FUNCTION (DATA CLEANSING)

#1 Importing ReGex, Pandas and SQLite into Python and create connection to the abusive and alay database

```
import re
import pandas as pd
import sqlite3

conn = sqlite3.connect('raw_data.db')
```

#2

- Import of Abusive Words and change into Lists and adjust the special characters for regexx pattern
- Import of alay words and change ino dictionary to lookup the formal words
- Alay words is used as key to lookup the formal words as value

```
data_abusive = pd.read_sql('Select word from abusive', conn)
df_abusive_list = data_abusive.values.tolist()
df_dict_abusive = str(df_abusive_list).replace('[','').replace('[','').replace(']','').replace(',','|').replace("'","")
```

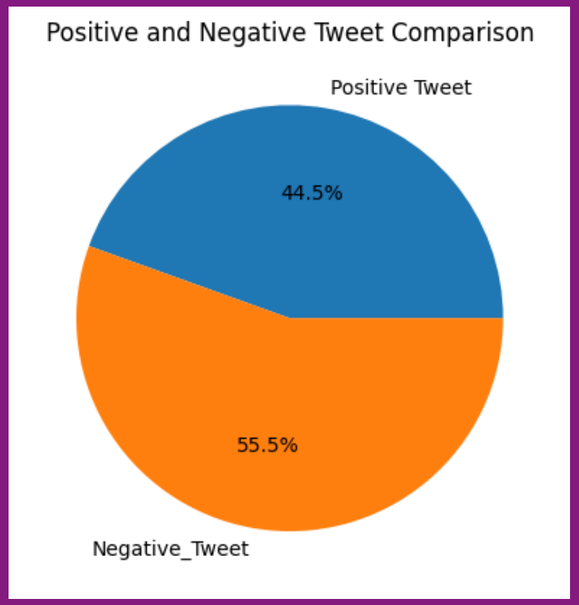
```
data_alay_word = pd.read_sql('Select * from alay', conn)
df_alay = dict(zip(data_alay_word['alay_word'],data_alay_word['formal_word']))
```

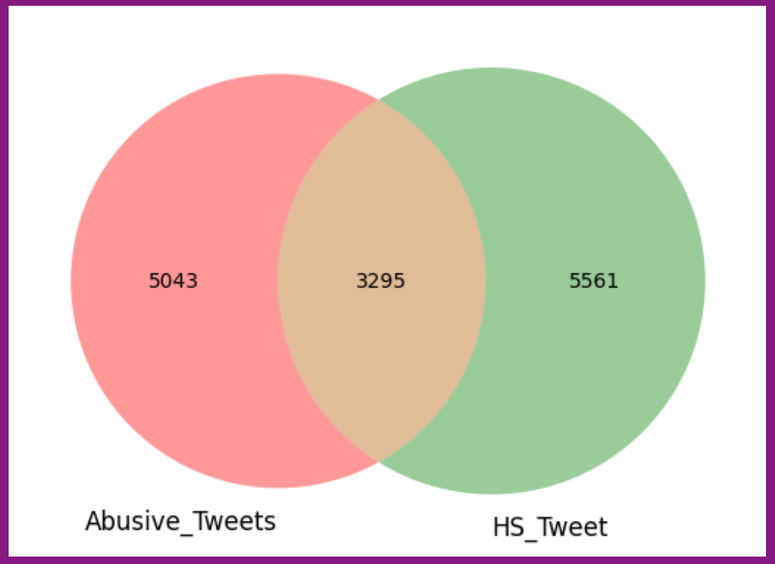
CODE FUNCTION (DATA CLEANSING)

```python
def text_cleansing(text):
    # Non-alphabet and numbers cleansing
    clean_text = re.sub(r'[^a-zA-Z0-9\s]', '', text)

    # Lower text
    clean_text = clean_text.lower()

    # Masking abusive words with xxx
    clean_text = re.sub(f'\{df_dict_abusive}\S+', ' xxx', clean_text)

    # Replacing alay words with its formal words
    clean_text = " ".join(df_alay.get(word, word) for word in clean_text.split())

    return clean_text
```

#3 Clean text :

- Remove characters that non-alphabet and numbers,
- Lower-case all characters
- Mask abusive words into ' xxx'
- And replace 'alay words' into its formal words





DATA ANALYSIS AND VISUALIZATION

Using matplotlib and univariate analysis on sample data provided, to looks at the distribution of a tweets connotations (positive and negative) insights below :

- Out of 13169 sample tweets, 55% (around 7.8k tweets) of these tweets contains negative connotations (abusive or hate speech)
- From 7.8k tweets with negative connotations :
  - 5043 contains abusive words
  - 5561 contains hate speech
  - 3295 contains both hatespeech and abusive words

DATA CLEANSING SAMPLE

#Example of masking and censor data from text input

Response body


```
{
  "raw_text": "%!^$!@^%$  aNjInG loE",
  "clean_text": "xxx kamu"
}
```


Download

#Example of masking and censor data from upload file input


```
{
  "0": {
    "raw_text": "- disaat semua cowok berusaha melacak perhatian gue. loe lantas remehkan perhatian yg gue kasih khusus ke elo. basic elo cowok bego ! !
!'",
    "clean_text": "di saat semua cowok berusaha melacak perhatian gue kamu lantas remehkan perhatian yang gue kasih khusus ke kamu basic kamu cowok xxx"
  },
  "1": {
    "raw_text": "RT USER: USER siapa yang telat ngasih tau elu?edan sarap gue bergaul dengan cigax jifla calis sama siapa noh licew juga'",
    "clean_text": "rt pengguna pengguna siapa yang telat memberi tau eluedan xxx gue bergaul dengan cigax jifla calis sama siapa itu licew juga"
  },
  "2": {
    "raw_text": "41. Kadang aku berfikir, kenapa aku tetap percaya pada Tuhan padahal aku selalu jatuh berkali-kali. Kadang aku merasa Tuhan itu ning-
galkan aku sendirian. Ketika orangtuaku berencana berpisah, ketika kakakku lebih memilih jadi Kristen. Ketika aku anak ter",
    "clean_text": "41 kadang aku berpikir kenapa aku tetap percaya pada tuhan padahal aku selalu jatuh berkalikali kadang aku merasa tuhan itu mening-
galkan aku sendirian ketika orang tuaku berencana berpisah ketika kakakku lebih memilih jadi kristen ketika aku anak ter"
  },
  "3": {
    "raw_text": "USER USER AKU ITU AKU\\n\\nKU TAU MATAMU SIPIT TAPI DILIAT DARI MANA ITU AKU'",
    "clean_text": "pengguna pengguna aku itu akunnku tau matamu xxx tapi dilihat dari mana itu aku"
  },
  "4": {
    "raw_text": "USER USER Kaum cebong kapir udah keliatan dongoknya dari awal tambah dongok lagi hahahah'",
```

SUMMARY

The sample data shows that more than half of the tweets that sent contains negative connotations. This shows the importance of data cleansing to mask and censor abusive and hate speech words, and replace 'alay word' to its formal word.

Based on the results, we can summarize that API that has been built to makes the tweets with negative connotations safe for work are successful. This API therefore is effective to minimize discrimination against marginalized community, prevent hate-spread and motivate people to use proper Bahasa Indonesia in the process.

Thank you!