TLAB DATA SCIENTIST TECHNICAL TEST
BY : RIYAN ANDRIYANTO

1. Preprocessing Data

Di awal dari penyusunan sentiment analisis yang dirancang, dilakukan beberapa cleansing pada dataset yang tersedia antara lain :

a. Menghapus data duplikat apabila ditemukan duplikasi data pada dataset, untuk menjalankan proses ini dilakukan dengan drop_duplicate().

b. Menghapus data dengan value NaN pada dataset karena data tersebut tidak dapat di proses dan mengganggu proses berikutnya ketika pembuatan model. Proses ini dijalankan dengan perintah dropna().

c. Melakukan cleansing terhadap text yang dilebih lebihkan seperti contoh PSBB yang di ketik sebagai PSBBBB apabila ditemukan text tersebut.

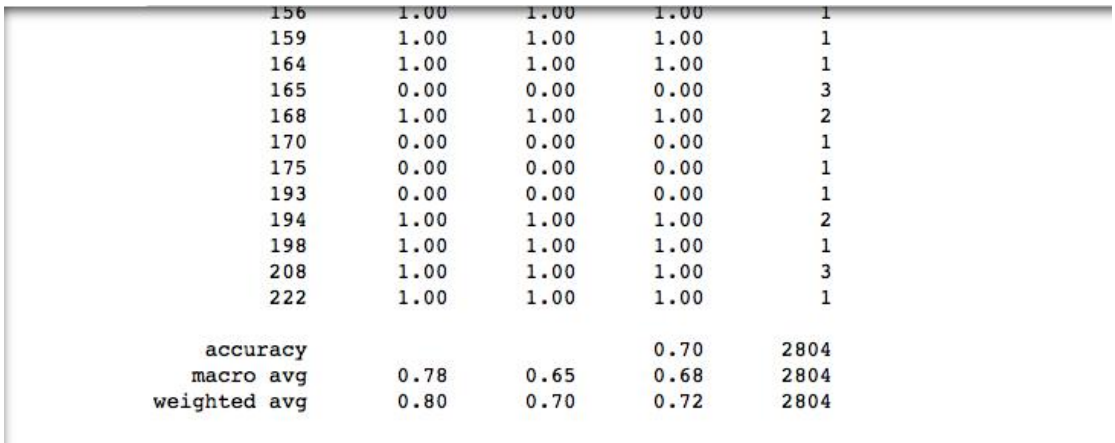
d. Membuat word dictionary dan word tokenization.

2. Library Bahasa Indonesia Yang Digunakan

Library Bahasa Indonesia yang digunakan untuk merancang sentyment analysis pada test ini adalah Sastrawi, dipilihnya library Sastrawi dikarenakan library ini cukup sederhana untuk digunakan pada proses stemming kata berimbuhan menjadi kata dasar dari kata tersebut.

3. Pemilihan Algoritma

Algoritma yang dipilih dalam melakukan sentyment analysis adalah Support Vector Classifier, karena algoritma tersebut dapat menemukan hyperplane terbaik dengan memaksimalkan jarak anter kelas data yang akan di klasifikasi. Hasil dari algoritma tersebut menunjukan akurasi 70%.



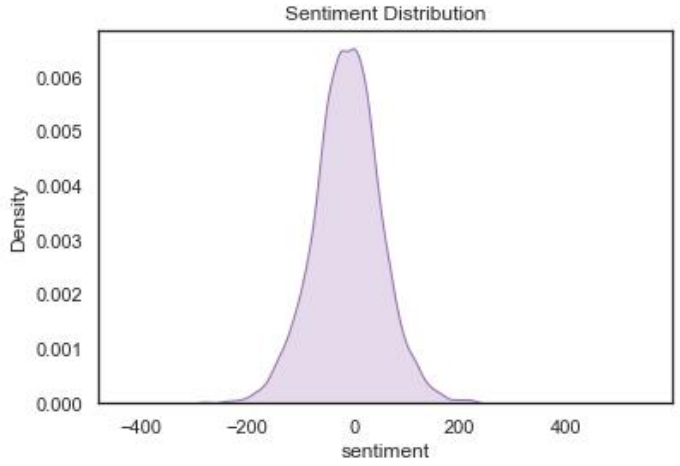
156	1.00	1.00	1.00	1
159	1.00	1.00	1.00	1
164	1.00	1.00	1.00	1
165	0.00	0.00	0.00	3
168	1.00	1.00	1.00	2
170	0.00	0.00	0.00	1
175	0.00	0.00	0.00	1
193	0.00	0.00	0.00	1
194	1.00	1.00	1.00	2
198	1.00	1.00	1.00	1
208	1.00	1.00	1.00	3
222	1.00	1.00	1.00	1
accuracy			0.70	2804
macro avg	0.78	0.65	0.68	2804
weighted avg	0.80	0.70	0.72	2804

4. Visualisasi Data

In [24]:
```
wordcloud = WordCloud(width = 800, height = 800, background_color = 'black', max_words = 1000
                      , min_font_size = 20).generate(str(word_to_plot_1))
fig = plt.figure(figsize = (8,8), facecolor = None)
plt.imshow(wordcloud)
plt.axis('off')
plt.show()
```

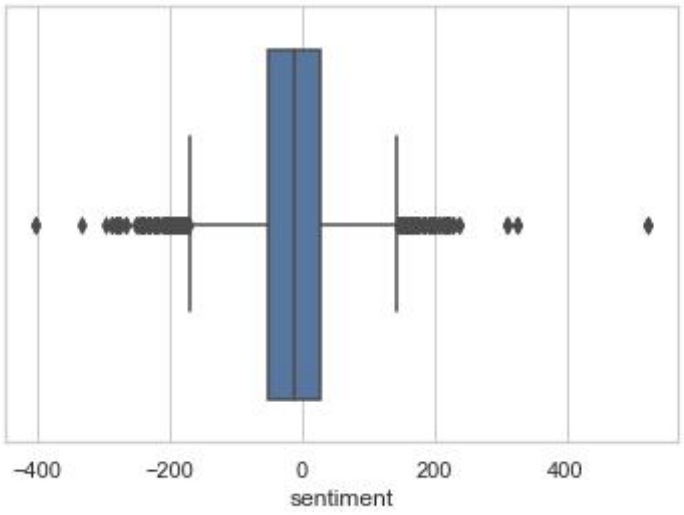

```
sns.set(style="white", palette="muted", color_codes=True)
sns.kdeplot(df_sen['sentiment'],color='m',shade=True)
plt.title('Sentiment Distribution')
plt.xlabel('sentiment')
```

```
Text(0.5, 0, 'sentiment')
```

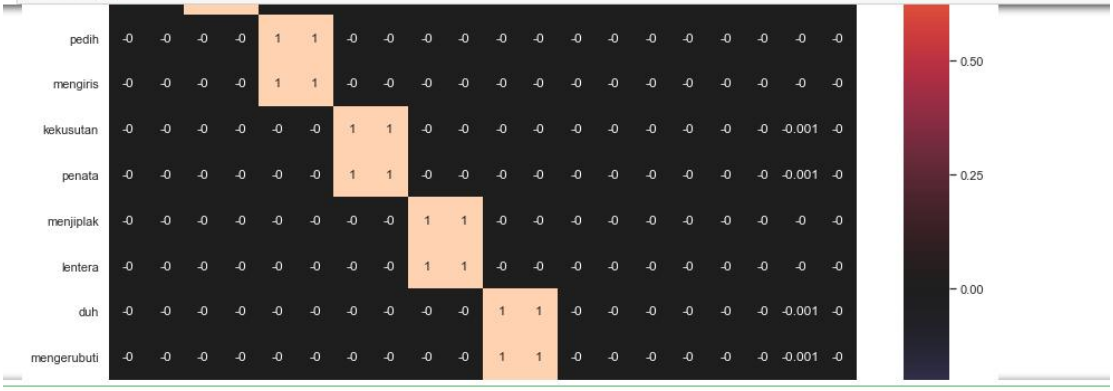


```
sns.set(style="whitegrid")
sns.boxplot(x=df_sen['sentiment'])
```

```
<AxesSubplot:xlabel='sentiment'>
```




```
plt.figure(figsize=(15,15))
h = sns.heatmap(corr, annot=True,vmin=-1, vmax=1, center= 0)
plt.show()
```



```
pal =sns.light_palette("navy", reverse=True,n_colors=15)
g = sns.barplot(y = top15_word.index , x = top15_word,palette=pal)
g.grid(False)
plt.xlabel('Occurences')
plt.ylabel('Words')
plt.title("Top 15 Most Often Occured Words",fontweight='bold')
for i in range(15):
    g.text(top15_word[i],i+0.22, top15_word[i],color='black')
plt.show()
```