

Penggalan Sosial Media

Tugas 3

Nama : Rahmatullah

Nim : 10221027

1. Tambahkan Confusion Matrix Pada Tugas 2 no 5

- Buat confusion matrix untuk mengevaluasi performa model.

```
from sklearn.metrics import classification_report, confusion_matrix

y_pred = classifier.predict(X)
print(confusion_matrix(labels, y_pred))
print(classification_report(labels, y_pred))
```

[10] ✓ 0.0s

```
... [[12  0]
      [ 0  8]]
```

		precision	recall	f1-score	support
	0	1.00	1.00	1.00	12
	1	1.00	1.00	1.00	8
	accuracy			1.00	20
	macro avg	1.00	1.00	1.00	20
	weighted avg	1.00	1.00	1.00	20

Pada hasil confusion matrix [12 0] menunjukkan model berhasil memprediksi 100% kalimat dengan sentiment positive dengan benar. Dan [0 8] menunjukkan model memprediksi semua sentiment negative dengan benar.

2. Tambahkan Visualisasi Data Pada Tugas 2 no 5

- Visualisasikan distribusi frekuensi kata menggunakan library seperti Matplotlib atau Seaborn

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

vocab = vectorizer.get_feature_names_out()
freqs = X.toarray().sum(axis=0)
word_freq_df = pd.DataFrame({'word': vocab, 'frequency': freqs})
word_freq_df = word_freq_df.sort_values('frequency', ascending=False)
top_n = 10
top_words = word_freq_df.head(top_n)
plt.figure(figsize=(12, 8))
sns.set_style("whitegrid")
sns.set_palette("viridis")
ax = sns.barplot(x='frequency', y='word', data=top_words, hue='word', legend=False)
plt.title("Top {} Most Frequent Words".format(top_n), fontsize=18, pad=20)
plt.xlabel("Frequency", fontsize=14)
plt.ylabel("Words", fontsize=14)

for i, v in enumerate(top_words['frequency']):
    ax.text(v + 0.1, i, str(int(v)), va='center')

plt.tight_layout()
plt.show()
```

