Penggalian Sosial Media

Tugas 2

Nama : Rahmatullah

Nim : 10221027

1. Buatkan Environtment Conda dengan nama envi_nama-mahasiswa
   - Tunjukkan dengan screnshot conda activate envi_nama-mahasiswa

```
#
# To activate this environment, use
#
#     $ conda activate envi_rahmatullah
#
# To deactivate an active environment, use
#
#     $ conda deactivate


(base) C:\laragon\www\psm>conda activate envi_rahmatullah

(envi_rahmatullah) C:\laragon\www\psm>
```

2. Modifikasi Dataset
   - Tambahkan beberapa kalimat baru ke dalam dataset (texts) dan label (labels).
   - Pastikan untuk menambahkan contoh teks dengan sentimen positif dan negatif.
   - Latih ulang model dan uji apakah prediksi untuk teks baru berubah.

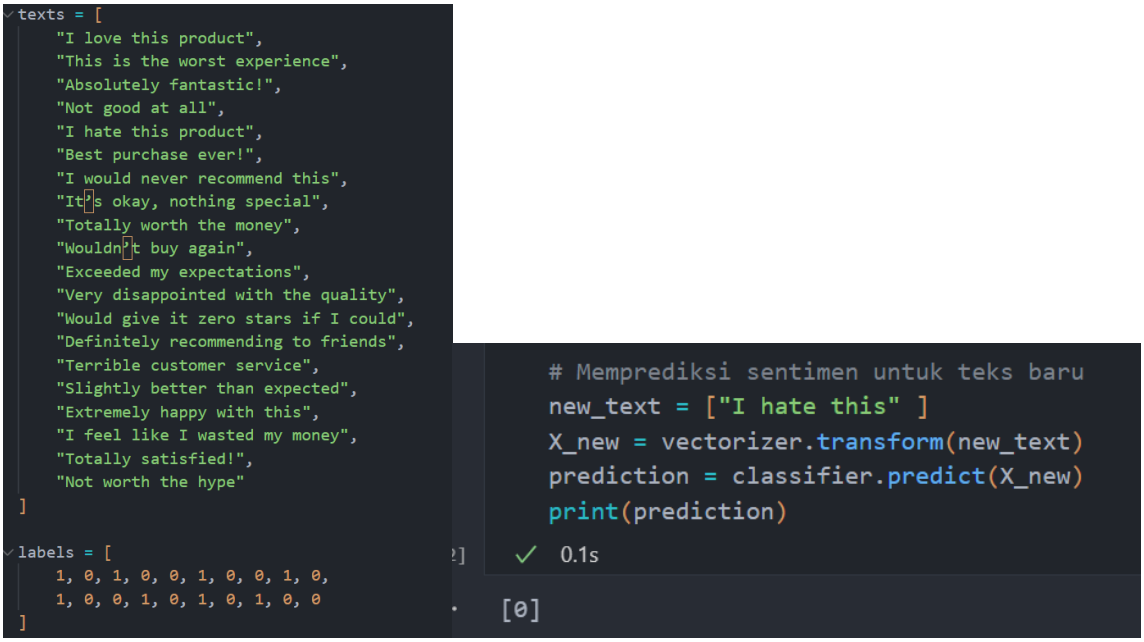
   Sebelum ditambah :

```python
from sklearn.feature_extraction.text import CountVect
from sklearn.naive_bayes import MultinomialNB

# Data contoh
texts = ["I love this product" , "This is the worst e
         "Not good at all" , "i hate product" ]
labels = [1, 0, 1, 0, 0]

# Mengubah data teks menjadi format angka
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(texts)

# Melatih model Naive Bayes
classifier = MultinomialNB()
classifier.fit(X, labels)

# Memprediksi sentimen untuk teks baru
new_text = ["I hate this" ]
X_new = vectorizer.transform(new_text)
prediction = classifier.predict(X_new)
print(prediction)
```

✓ 2.8s

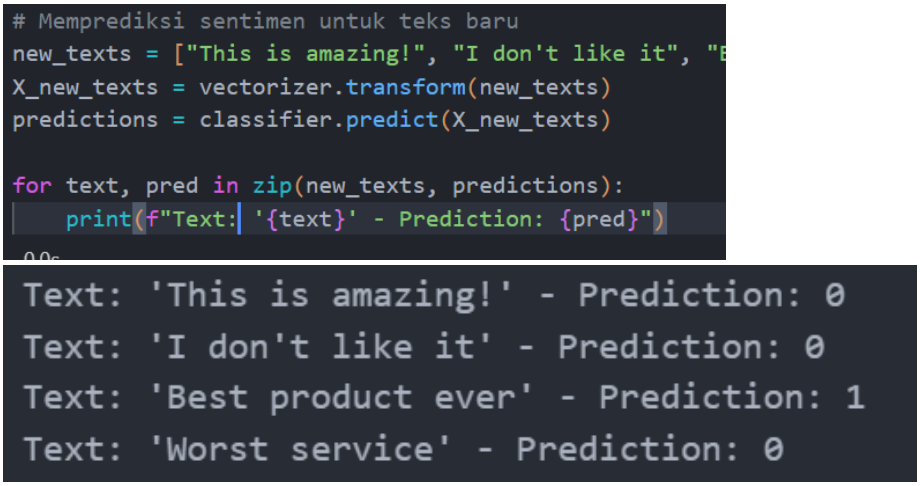
```
[0]
```

Setelah ditambah:



setelah dataset ditambah, dapat dilihat bahwa hasilnya tetap sama. Model masih bisa memprediksi kalimat “I hate this” merupakan sentiment negative.

3. Prediksi untuk Beberapa Teks Baru
   - Uji model dengan beberapa teks baru
   - Transformasikan teks baru menggunakan vectorizer.transform() dan prediksi sentimen untuk setiap teks.



Dari hasil yang didapat, model memprediksi 3 text baru dengan dengan banar dari 4 text baru. Model gagal memprediksi text “This is amazing!” yang seharusnya positif.

4. Analisis Vocabulary
   - Tampilkan vocabulary yang dihasilkan oleh CountVectorizer.
   - print("Vocabulary:", vectorizer.get_feature_names_out())

- Jelaskan bagaimana vocabulary ini digunakan dalam pembentukan matriks fitur.

```
    print("Vocabulary:", vectorizer.get_feature_names_out())
✓ 0.0s

Vocabulary: ['absolutely' 'again' 'all' 'at' 'best' 'better' 'buy' 'could' 'customer'
 'definitely' 'disappointed' 'ever' 'exceeded' 'expectations' 'expected'
 'experience' 'extremely' 'fantastic' 'feel' 'friends' 'give' 'good'
 'happy' 'hate' 'hype' 'if' 'is' 'it' 'like' 'love' 'money' 'my' 'never'
 'not' 'nothing' 'okay' 'product' 'purchase' 'quality' 'recommend'
 'recommending' 'satisfied' 'service' 'slightly' 'special' 'stars'
 'terrible' 'than' 'the' 'this' 'to' 'totally' 'very' 'wasted' 'with'
 'worst' 'worth' 'would' 'wouldn' 'zero']
```

Vocabulary dalam vectorizer seperti CountVectorizer, adalah daftar kata unik atau token yang ditemukan dalam seluruh korpus (kumpulan teks) yang digunakan untuk membangun matriks fitur. Setiap kata dalam vocabulary ini diberi indeks unik, dan vectorizer menggunakan vocabulary ini untuk mengubah teks mentah menjadi format numerik. Pertama dengan membangun vocabulary dari semua kata yang ada di dalam teks, kemudian setiap teks diubah menjadi vektor fitur berdasarkan kemunculan kata-kata dalam vocabulary. Setiap teks diubah menjadi vektor di mana setiap elemen mewakili jumlah kemunculan kata tertentu yang ada di vocabulary. Hasil akhirnya adalah matriks di mana setiap baris mewakili satu teks dan setiap kolom mewakili kata dalam vocabulary, dengan nilai dalam matriks menunjukkan frekuensi atau bobot kata tersebut dalam teks yang bersangkutan.

5. Modifikasi Dataset
   - Tambahkan beberapa kalimat baru berbahasa indonesia ke dalam dataset (texts) dan label (labels).
   - Pastikan untuk menambahkan contoh teks dengan sentimen positif dan negatif.
   - Gunakan library sastrawi
   - Latih ulang model dan uji apakah prediksi untuk teks baru berubah.

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

# Data contoh dalam bahasa Indonesia
texts = [
    "Saya suka produk ini",
    "Ini adalah pengalaman terburuk",
    "Sangat fantastis!",
    "Tidak bagus sama sekali",
    "Saya benci produk ini",
    "Pembelian terbaik!",
    "Saya tidak akan merekomendasikan ini",
    "Cukup oke, tidak istimewa",
    "Sangat layak dengan uangnya",
    "Tidak akan membeli lagi",
    "Melebihi ekspektasi saya",
    "Sangat kecewa dengan kualitasnya",
    "Akan memberikan nol bintang jika bisa",
    "Pasti merekomendasikan kepada teman",
    "Layanan pelanggan sangat buruk",
    "Sedikit lebih baik dari yang saya harapkan",
    "Sangat senang dengan ini",
    "Saya merasa seperti membuang uang saya",
    "Sangat puas!",
    "Tidak sesuai dengan hype"
]
```

```
labels = [
    1, 0, 1, 0, 0, 1, 0, 0, 1, 0,
    1, 0, 0, 1, 0, 1, 0, 1, 0, 0
]

# Preprocessing teks menggunakan Sastrawi (stemming)
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# Fungsi untuk melakukan stemming pada setiap teks
def preprocess(texts):
    return [stemmer.stem(text) for text in texts]

# Mengubah teks menjadi bentuk yang lebih dasar (setelah stemming)
processed_texts = preprocess(texts)

# Mengubah data teks menjadi format angka
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(processed_texts)

# Melatih model Naive Bayes
classifier = MultinomialNB()
classifier.fit(X, labels)

# Teks baru dalam bahasa Indonesia
new_texts = ["Ini luar biasa!", "Saya tidak suka ini", "Produk ter
```

```
    # Preprocessing pada teks baru
    processed_new_texts = preprocess(new_texts)

    # Memprediksi sentimen untuk teks baru
    X_new_texts = vectorizer.transform(processed_new_texts)
    predictions = classifier.predict(X_new_texts)

    # Menampilkan hasil prediksi
    for text, pred in zip(new_texts, predictions):
        print(f"Text: '{text}' - Prediction: {pred}")
```

✓ 0.8s

```
Text: 'Ini luar biasa!' - Prediction: 0
Text: 'Saya tidak suka ini' - Prediction: 0
Text: 'Produk terbaik' - Prediction: 1
Text: 'Layanan terburuk' - Prediction: 0
```

Model memprediksi 3 kalimat secara benar dari 4 kalimat yang diuji, Dimana prediksi kalimat yang salah adalah “Ini luar biasa” yang seharusnya sentiment positive, model memprediksi kalimat tersebut sebagai sentiment negative.