SENTIMENT ANALYSIS

NAMA : NAJLA DHIA RUSYDI

NIM : 164221043

MATA KULIAH : NLP

Lakukan analysis sentiment untuk dataset yang telah diberikan untuk yang berbahasa inggris.

1. Import data

```
df = pd.read_excel('ExerciseSAClass.xlsx')
df.head()
```

[2] ✓ 7.1s Python

	Post ID	Post description	Date	Language	Translated Post Description	Sentiment	Hate
0	CgXDOaQDvGm	"I have decided that the global #monkeypox out...	07/23/2022	English	"I have decided that the global #monkeypox out...	neutral	Not Hate
1	CgXpRmMIdzG	In light of the evolving monkeypox outbreak wi...	07/23/2022	English	In light of the evolving monkeypox outbreak wi...	neutral	Not Hate
2	CgXaFGDsevq	If you've been hearing about monkeypox and wan...	07/23/2022	English	If you've been hearing about monkeypox and wan...	neutral	Not Hate
3	CgXGNrmLwoL	Monkeypox is a rare disease caused by infectio...	07/23/2022	English	Monkeypox is a rare disease caused by infectio...	neutral	Not Hate
4	CgXTqcjOQD-	For today's @newyorkermag dispatch. \n'The Ago...	07/23/2022	English	For today's @newyorkermag dispatch. \n'The Ago...	negative	Not Hate

2. Filter data untuk language yang berbahasa inggris

```
print("Original dataset size:", len(df))
df_English = df[df['Language'] == 'English']
df_English = df_English[['Post description', 'Language', 'Sentiment']]
print("Dataset size after filtering for English:", len(df_English))
df_English.head()
```

[3] ✓ 0.0s

```
Original dataset size: 35719
Dataset size after filtering for English: 22358
```

	Post description	Language	Sentiment
0	"I have decided that the global #monkeypox out...	English	neutral
1	In light of the evolving monkeypox outbreak wi...	English	neutral
2	If you've been hearing about monkeypox and wan...	English	neutral
3	Monkeypox is a rare disease caused by infectio...	English	neutral
4	For today's @newyorkermag dispatch. \n'The Ago...	English	negative

Karna data awal memiliki beberapa Bahasa, maka data perlu di filter lebih spesifik dengan memfilter data untuk variable language yang berlabel English.

3. Preprocessing

Preprocessing merupakan tahap penting dalam analisis sentimen yang bertujuan untuk membersihkan data teks sebelum dimasukkan ke dalam model. Langkah ini memastikan bahwa teks yang dianalisis lebih konsisten dan terstruktur, sehingga model dapat lebih mudah memahami pola dalam data. Dengan membersihkan elemen-elemen seperti tanda baca, stopwords, dan teks yang tidak relevan, model dapat lebih fokus pada kata-kata penting yang menentukan sentimen, baik itu positif, negatif, maupun netral.

```
English_stopwords = set(stopwords.words('english'))

def preprocess_text(text):
    if pd.isna(text):
        return ""
    text = text.lower()
    text = re.sub(r'[^a-zA-Z\s]', '', text)
    tokens = word_tokenize(text)
    tokens = [token for token in tokens if token not in English_stopwords]
    return ' '.join(tokens)

df_English.loc[:, 'processed_text'] = df_English['Post description'].apply(preprocess_text)

df_English[['Post description', 'processed_text']].head()
```

[4] ✓ 8.6s

Preprocessing di atas melakukan beberapa langkah untuk membersihkan teks. Pertama, teks diubah menjadi huruf kecil agar konsisten. Kemudian, semua karakter selain huruf dan spasi dihapus menggunakan regular expression. Selanjutnya, teks dipecah menjadi token atau kata-kata individu, dan kata-kata umum yang tidak memberikan informasi penting (stopwords) dihilangkan. Hasil akhirnya adalah teks yang sudah bersih dan siap digunakan dalam analisis, dengan hanya menyisakan kata-kata yang relevan.

NAMA : NAJLA DHIA RUSYDI

NIM : 164221043

MATA KULIAH : NLP

Preprocessing ini diterapkan pada kolom 'Post description' dan disimpan dalam kolom baru 'processed_text'.

```
    sentiment_map = {'neutral': 0, 'negative': 1, 'positif': 2}
    df_English['sentiment_numeric'] = df_English['Sentiment'].map(sentiment_map)

    df_English = df_English.dropna(subset=['processed_text', 'sentiment_numeric'])
    print("Dataset size after removing NaN values:", len(df_English))

[6]  ✓ 0.0s

...  Dataset size after removing NaN values: 22358
```

Kemudian memetakan label sentimen dari teks ('neutral', 'negative', 'positif') menjadi bentuk numerik (0, 1, 2) agar lebih mudah diproses oleh model machine learning, dan kemudian menghapus baris yang memiliki nilai kosong (NaN) di kolom 'processed_text' dan 'sentiment_numeric', sehingga hanya data yang lengkap yang tersisa untuk analisis. Setelahnya, ukuran dataset yang bersih dicetak untuk mengetahui jumlah data yang dapat digunakan.

4. Split data

```
    X_train, X_test, y_train, y_test = train_test_split(df_English['processed_text'], df_English['sentiment_numeric'], test_size=0.2, random_state=42)
[7]  ✓ 0.0s                                                                                                                                    Python
```

Untuk melakukan modeling tentu data perlu di split antara data test dan data train. Nilai X berisi text ‘processed_text’ atau data yang telah dibersihkan, kemudian untuk variable y, merupakan kolom ‘sentiment_numeric’ yang hasil mapping sebelumnya.

5. TF-IDF

```
    tfidf_vectorizer = TfidfVectorizer(max_features=1000)
    X_train_tfidf = tfidf_vectorizer.fit_transform(X_train)
    X_test_tfidf = tfidf_vectorizer.transform(X_test)
[8]  ✓ 1.5s
```

Setelah melakukan split data, selanjutnya dilakukan proses transformasi teks menggunakan TF-IDF (Term Frequency-Inverse Document Frequency). Proses ini mengubah data teks menjadi representasi numerik berdasarkan frekuensi kemunculan kata yang relevan. Vektor TF-IDF diterapkan pada data ‘X_train’ untuk membangun model kata-kata dengan maksimal 1000 fitur, kemudian diterapkan pada data uji ‘X_test’ agar model dapat mempelajari dan menguji hubungan antara kata-kata dalam teks dan label sentimen. Proses ini penting untuk mempersiapkan teks menjadi input yang dapat diolah oleh model machine learning.

6. Modelling

```
rf_classifier = RandomForestClassifier()
rf_classifier.fit(X_train_tfidf, y_train)

y_pred_rf = rf_classifier.predict(X_test_tfidf)
```

Kemudian dilakukan pemodelan dilakukan menggunakan Random Forest, yang terdiri dari dua tahap: tahap pertama adalah melatih model (training) menggunakan data latih, di mana model mempelajari pola-pola dari teks yang ada. Setelah model berhasil

NAMA : NAJLA DHIA RUSYDI

NIM : 164221043

MATA KULIAH : NLP

mempelajari pola tersebut, dilakukan tahap kedua yaitu prediksi atau peramalan nilai y pada data uji untuk menilai performa model dalam memprediksi sentimen berdasarkan teks.

7. Evaluasi model

```
Random Forest:
Accuracy: 0.8671735241502684
Precision: 0.86
Recall: 0.87
F1-score: 0.85
```

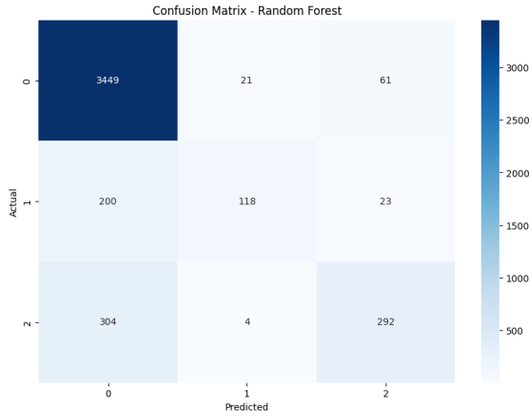
```
Classification Report:
              precision    recall  f1-score   support

           0       0.88      0.98      0.92      3531
           1       0.83      0.36      0.50       341
           2       0.80      0.50      0.61       600

    accuracy                           0.87      4472
   macro avg       0.83      0.61      0.68      4472
weighted avg       0.86      0.87      0.85      4472
```

Model Random Forest yang digunakan untuk memprediksi sentimen teks menunjukkan hasil yang cukup baik dengan akurasi sebesar 86.7%. Dari hasil evaluasi, nilai precision sebesar 0.86 mengindikasikan bahwa model cukup akurat dalam memberikan prediksi yang benar dari keseluruhan prediksi positif. Recall sebesar 0.87 menunjukkan bahwa model mampu mengenali sebagian besar instance yang benar-benar positif. Nilai F1-score yang mencapai 0.85 mengindikasikan keseimbangan yang baik antara precision dan recall.

Jika dilihat lebih rinci pada masing-masing kelas, kelas netral (0) memiliki performa paling baik dengan precision 0.88, recall 0.98, dan F1-score 0.92. Namun, untuk kelas negatif (1) performanya lebih rendah, dengan precision 0.83 dan recall 0.36, menunjukkan bahwa model kurang mampu mengenali sentimen negatif dengan baik. Kelas positif (2) juga menunjukkan performa sedang, dengan precision 0.80 dan recall 0.50, yang berarti prediksi untuk kelas positif lebih sulit bagi model. Secara keseluruhan, meskipun model memiliki performa baik untuk kelas netral, masih ada ruang untuk peningkatan pada prediksi kelas negatif dan positif.



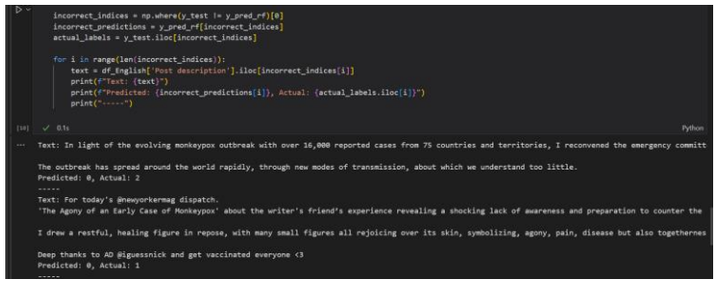
Di atas merupakan visualisasi dari confusion matrix, di mana kita dapat melihat sebaran data yang diprediksi dengan benar maupun salah oleh model. Confusion matrix memberikan gambaran rinci tentang berapa banyak instance yang diprediksi benar untuk setiap kelas serta kesalahan prediksi antara kelas-kelas tersebut. Dengan bantuan confusion matrix, kita dapat mengidentifikasi pola kesalahan yang terjadi, seperti instance yang seharusnya termasuk dalam kelas tertentu tetapi diprediksi sebagai kelas lain. Hal ini membantu untuk memahami kelemahan model dan menentukan area yang memerlukan perbaikan.

SENTIMENT ANALYSIS

NAMA : NAJLA DHIA RUSYDI

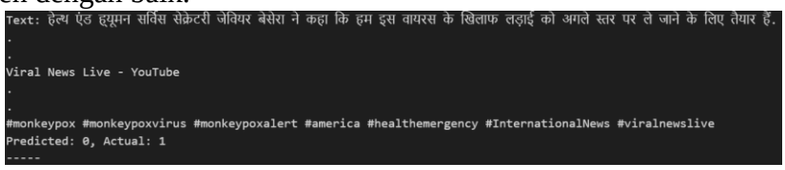
NIM : 164221043

MATA KULIAH : NLP

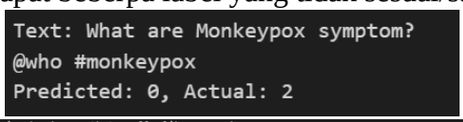


Untuk memahami lebih detail mengapa model memprediksi nilai y dengan salah, kita dapat melihat teks asli yang digunakan sebagai input pada model serta hasil prediksi yang tidak sesuai dengan label aslinya. Dengan menelusuri contoh-contoh ini, kita bisa menganalisis pola dari kesalahan prediksi, seperti apakah model kesulitan membedakan antara kelas-kelas tertentu atau apakah ada fitur dalam teks yang membuat prediksi menjadi sulit. Langkah ini sangat penting dalam proses debugging dan meningkatkan performa model, karena kita bisa mengetahui kelemahan spesifik model dalam menangani kasus-kasus tertentu. Berikut beberapa analisis saya :

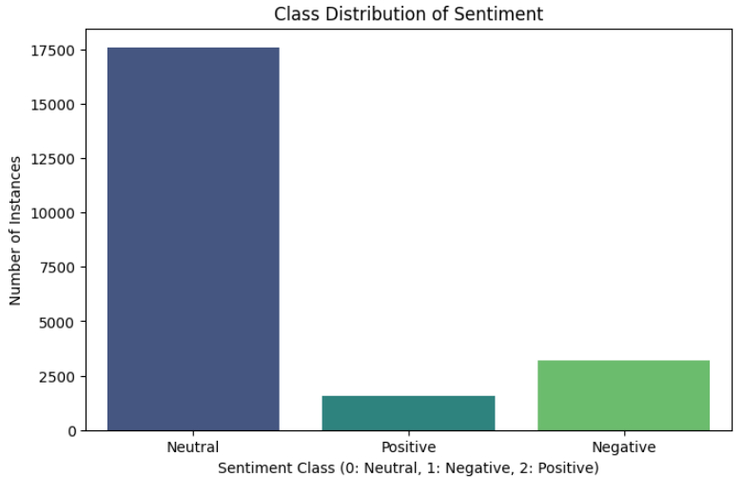
- Data dengan bahas berbeda masuk, sehingga text ini tidak dilakukan analisis sentimen dengan baik.



- Menurut saya terdapat beberapa label yang tidak sesuai/salah



- Imbalance data, banyak sekali text yang di prediksi sebagai 0/netral, karna model banyak belajar untuk label 0 ini.



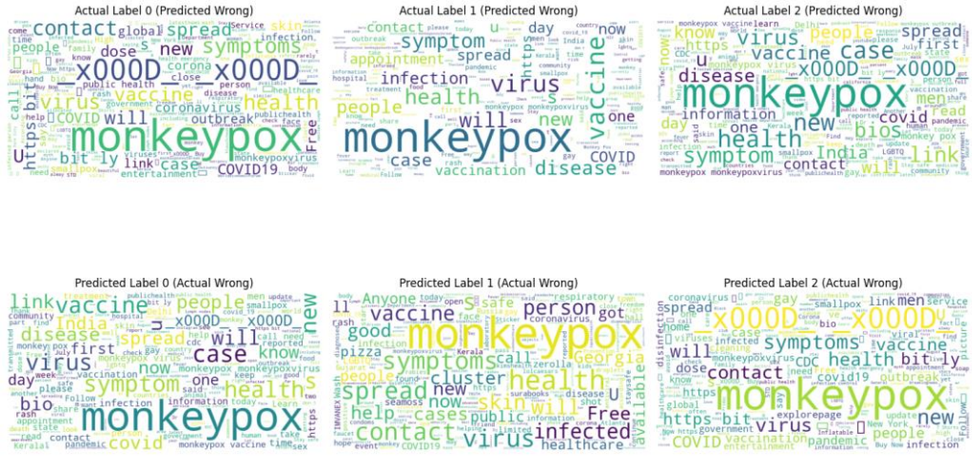
SENTIMENT ANALYSIS

NAMA : NAJLA DHIA RUSYDI

NIM : 164221043

MATA KULIAH : NLP

- Word cloud



Melalui visualisasi Word Cloud, saya dapat menggali bagaimana model salah dalam mengklasifikasikan label. Analisis ini difokuskan pada perbandingan antara label yang diprediksi salah oleh model dan label aktual yang seharusnya. Dengan demikian, kami dapat mengidentifikasi pola dan potensi masalah dalam proses klasifikasi yang dapat diperbaiki untuk meningkatkan akurasi model.

1. Ketidakseimbangan Konteks:
   - Kata Dominan: Actual Label 0 (diprediksi salah) didominasi oleh kata-kata seperti "vaccine", "dose", "contact", dan "X000D" (bisa jadi username). Sementara Predicted Label 0 (sebenarnya salah) lebih fokus pada "case", "spread", "virus", dan "health".
   - Analisis: Model mungkin kesulitan membedakan antara informasi faktual netral (seperti tentang vaksin dan dosis) dengan informasi tentang penyebaran penyakit yang mungkin dipersepsikan lebih negatif.
2. Ambiguitas Sentimen dalam Konteks Kesehatan:
   - Kata Dominan: Kata-kata seperti "spread", "virus", dan "case" yang muncul di label prediksi 0 memiliki konotasi negatif dalam konteks umum, tetapi dalam diskusi ilmiah atau laporan kesehatan, kata-kata ini mungkin digunakan secara netral.
   - Analisis: Model mungkin terlalu mengandalkan konotasi umum kata-kata ini tanpa mempertimbangkan konteks khusus diskusi kesehatan publik.
3. Overemphasis pada Kata Kunci Tertentu:
   - Kata "Monkeypox" muncul dominan di kedua label, menunjukkan bahwa model mungkin terlalu bergantung pada kehadiran kata ini untuk klasifikasi.
   - Analisis: Model perlu lebih baik dalam memahami konteks sekitar kata kunci utama, bukan hanya kehadirannya.
4. Kurangnya Nuansa dalam Klasifikasi:
   - Kata Dominan: Kata-kata seperti "COVID", "coronavirus", dan "outbreak" muncul di kedua label, menunjukkan kesulitan model dalam membedakan nuansa diskusi tentang berbagai penyakit menular.

NAMA : NAJLA DHIA RUSYDI

NIM : 164221043

MATA KULIAH : NLP

- Analisis: Model mungkin perlu pelatihan lebih lanjut untuk membedakan konteks spesifik dari berbagai wabah penyakit.

5. Potensi Bias dalam Data Pelatihan:
   - Kata Dominan: Munculnya kata-kata seperti "India", "Georgia", dan "Kerala" menunjukkan adanya fokus geografis tertentu.
   - Analisis: Model mungkin mengalami overfitting pada contoh-contoh spesifik dalam data pelatihan, yang dapat membuatnya kurang mampu melakukan generalisasi.
6. Kesulitan dalam Menginterpretasikan Frasa Multi-Kata:
   - Frasa seperti "public health" atau "healthcare" memiliki makna yang lebih kompleks dan spesifik dibandingkan dengan kata-kata individual yang menyusunnya.
   - Analisis: Model mungkin mengalami kesulitan dalam memahami arti dari frasa-frasa ini secara utuh. Hal ini menunjukkan perlunya perbaikan dalam pengolahan NLP untuk mempertimbangkan konteks dan makna frasa secara keseluruhan, bukan hanya mengandalkan analisis kata per kata. Pemahaman yang lebih baik terhadap frasa multi-kata dapat membantu model dalam menangkap nuansa dan kompleksitas makna yang tidak dapat diungkapkan hanya dengan menganalisis kata-kata secara terpisah.
7. Keterbatasan dalam Memahami Sentimen Implisit:
   - Kata Dominan: Kata-kata seperti "help", "learn", dan "information" yang muncul dalam prediksi label 0 bisa mengindikasikan sentimen positif atau netral yang tidak ditangkap dengan baik.
   - Analisis: Model mungkin perlu pelatihan lebih lanjut untuk menangkap sentimen implisit dalam konteks kesehatan.

KESIMPULAN

Model Random Forest yang digunakan untuk memprediksi sentimen teks menunjukkan hasil yang cukup baik dengan akurasi sebesar 86.7%. Evaluasi lebih lanjut menunjukkan bahwa model memiliki nilai precision sebesar 0.86, yang mengindikasikan bahwa proporsi prediksi positif yang benar cukup tinggi. Recall model mencapai 0.87, menandakan bahwa sebagian besar instance yang benar-benar positif berhasil dikenali. Nilai F1-score yang diperoleh sebesar 0.85 menunjukkan keseimbangan yang baik antara precision dan recall, mencerminkan kemampuan model untuk menangkap sentimen secara efektif.

Ketika melihat performa masing-masing kelas, kelas netral (0) menonjol dengan precision 0.88, recall 0.98, dan F1-score 0.92, menandakan model sangat efektif dalam mengidentifikasi sentimen netral. Namun, kelas negatif (1) menunjukkan performa yang lebih rendah, dengan precision 0.83 dan recall 0.36, mengindikasikan kesulitan model dalam mengenali sentimen negatif. Kelas positif (2) juga menunjukkan hasil sedang, dengan precision 0.80 dan recall 0.50. Ini menunjukkan bahwa meskipun model baik dalam mendeteksi kelas netral, terdapat tantangan signifikan dalam memprediksi kelas negatif dan positif, yang menandakan adanya ruang untuk peningkatan.

SENTIMENT ANALYSIS

NAMA : NAJLA DHIA RUSYDI

NIM : 164221043

MATA KULIAH : NLP

Analisis melalui Word Cloud memberikan wawasan mendalam mengenai kesalahan prediksi model. Kata-kata yang dominan dalam kelas yang diprediksi salah menunjukkan adanya kesulitan dalam membedakan konteks dan nuansa dari istilah yang digunakan. Masalah seperti ketidakseimbangan data, ambiguitas dalam konteks kesehatan, serta ketergantungan model pada kata kunci tertentu menjadi faktor penyebab utama dalam kesalahan prediksi.

Melalui analisis confusion matrix, kita dapat memahami distribusi prediksi model secara lebih rinci. Confusion matrix memberikan gambaran yang jelas tentang jumlah instance yang diprediksi benar maupun salah untuk setiap kelas. Dengan memanfaatkan informasi ini, kita dapat mengidentifikasi pola kesalahan yang terjadi, seperti kesulitan model dalam membedakan antara kelas positif dan negatif.

Untuk meningkatkan kinerja model, beberapa langkah yang direkomendasikan termasuk perbaikan dalam preprocessing data, penggunaan teknik NLP yang lebih canggih, serta pelatihan model dengan dataset yang lebih seimbang dan beragam. Dengan langkah-langkah ini, diharapkan model dapat memberikan hasil prediksi yang lebih akurat dan bermanfaat dalam analisis sentimen teks di masa mendatang