Laporan Ujian Akhir Semester Sains Data

Analisa Sentiment Market Bitcoin pada Pengguna Twiter

Disusun oleh:

1. Vincent Erwan Wijaya (123230025)
2. Adi Setya Nur Pradipta (123230026)
3. Syaidatul Munawirmah (123230036)

Dosen Pengampu:

Dhimas Arief Dharmawan

PROGRAM STUDI INFORMATIKA

JURUSAN INFORMATIKA FAKULTAS TEKNIK INDUSTRI

UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"

YOGYAKARTA 2025

DAFTAR ISI

BAB 1

Data Preparation

1.1 Sumber dan Deskripsi Data

Dataset yang digunakan pada proyek ini berasal dari Kaggle, yang memuat hasil scraping tweet dari platform Twitter/X menggunakan kata kunci yang berhubungan dengan cryptocurrency seperti *Bitcoin, Ethereum, BNB*, dan aset kripto lainnya. Seluruh data tersebut disimpan dalam file tweets.csv dengan ukuran sekitar 17 MB, berisi ribuan entri tweet dari pengguna yang membahas perkembangan aset digital. Dataset ini memiliki 10 variabel utama, termasuk *token* yang menunjukkan jenis aset kripto yang dibahas, *date* sebagai waktu unggahan tweet, serta empat metrik interaksi yaitu *reply_count, retweet_count, like_count,* dan *quote_count*. Selain itu, dataset juga memuat kolom *text* berisi isi tweet, *sentiment* sebagai label kategori sentimen (Positive, Neutral, atau Negative), dan *sentiment_score* berupa skor numerik hasil perhitungan model sentimen. Secara keseluruhan, dataset ini terdiri dari kombinasi tipe data teks, numerik, dan tanggal. Konten dataset merepresentasikan opini, persepsi, dan respons pengguna terhadap berbagai isu terkait aset kripto, sehingga sangat relevan untuk keperluan analisis sentimen maupun peramalan (forecasting) tren sentimen di media sosial.

Sumber Data:

🔗 Bitcoin Tweets Dataset – Kaggle (oleh Sujay Kapadnis)

1.2 Identifikasi Variabel

Dalam penelitian ini, dataset terdiri dari sepuluh variabel yang memiliki peran spesifik dalam arsitektur pemodelan. Komponen paling mendasar adalah variabel `text` (bertipe *object/string*), yang bertindak sebagai fitur prediktor utama. Data ini berisi opini mentah pengguna yang akan diolah melalui tahap pra-pemrosesan teks dan ekstraksi fitur menggunakan metode *embedding* agar dapat dipahami oleh mesin.

Untuk keperluan *supervised learning*, variabel `sentiment` (*categorical*) ditetapkan sebagai variabel target (*label*) yang membagi opini publik ke dalam tiga kelas: *Positive*, *Neutral*, dan *Negative*. Melengkapi label tersebut, variabel `sentiment_score` (*float*) digunakan untuk memberikan bobot numerik pada setiap prediksi, sehingga analisis tidak hanya terbatas pada klasifikasi kaku, melainkan juga dapat mengukur intensitas keyakinan model yang berguna untuk kebutuhan korelasi data.

Dari sisi analisis temporal, variabel `date` (*datetime*) memegang peran krusial sebagai indeks waktu. Variabel ini memungkinkan data dikelompokkan dalam rentang harian untuk membentuk pola deret waktu (*time-series*), yang menjadi landasan dalam memetakan fluktuasi tren sentimen pasar. Selain itu, terdapat variabel `token` (*nominal*) yang berfungsi sebagai *identifier* untuk memisahkan data berdasarkan jenis aset kripto yang sedang diamati.

Aspek terakhir yang dianalisis adalah tingkat atensi publik, yang direpresentasikan melalui empat variabel numerik (*integer*), yaitu `reply_count`, `retweet_count`, `like_count`, dan `quote_count`. Keempat variabel ini bukan sekadar data tambahan, melainkan indikator validitas sosial yang digunakan untuk mengukur seberapa besar dampak atau viralitas sebuah opini dalam mempengaruhi persepsi pasar secara keseluruhan.

Nama Variabel	**Deskripsi**	**Jenis Data**	**Level Pengukuran**	**Alasan Penentuan Jenis & Level Data**
Token	Menunjukkan topik utama dari tweet, dalam dataset ini seluruhnya adalah “bitcoin”.	Kualitatif	Nominal	Karena berupa label kategori tanpa urutan atau nilai numerik. Tidak memiliki hierarki, hanya berfungsi sebagai identitas topik.
Date	Menunjukkan tanggal dan waktu tweet	Kuantitatif (kontinu)	Interval	Dapat diubah menjadi nilai numerik (timestamp) dan

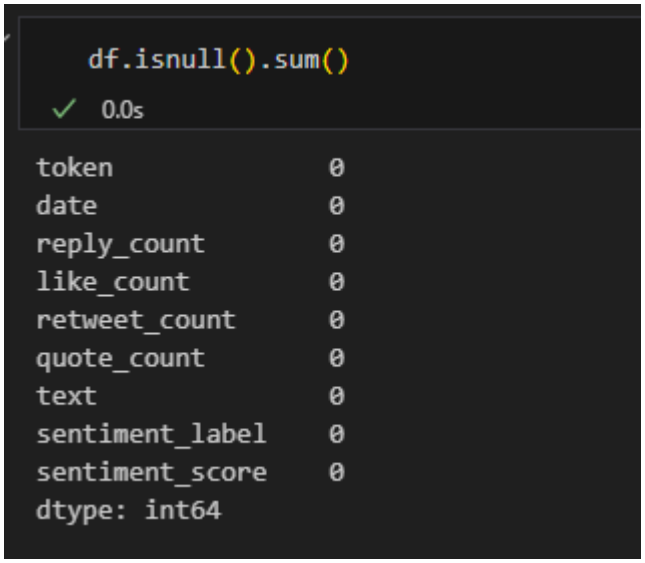
	dipublikasikan.			memiliki selisih bermakna antar waktu, namun tidak memiliki titik nol absolut.
reply_count	Jumlah balasan yang diterima oleh tweet.	Kuantitatif (diskrit)	Rasio	Nilai nol bermakna (tidak ada balasan). Perbandingan antar nilai dapat dilakukan secara proporsional, misalnya 20 lebih banyak dari 10.
retweet_count	Jumlah retweet yang diterima oleh tweet.	Kuantitatif (diskrit)	Rasio	Mengandung nilai hitungan yang dapat dibandingkan dan memiliki nol bermakna, menunjukkan frekuensi aktivitas pengguna.
like_count	Jumlah suka (likes) yang diterima oleh tweet.	Kuantitatif (diskrit)	Rasio	Menunjukkan tingkat popularitas tweet. Nilai nol berarti tidak ada suka, dan dapat dibandingkan secara proporsional antar tweet.
quote_count	Jumlah kutipan terhadap tweet.	Kuantitatif (diskrit)	Rasio	Termasuk data numerik yang dapat diukur dan dibandingkan secara proporsional, dengan nol sebagai titik mutlak tanpa kutipan.
text	Isi teks dari tweet	Kualitatif	Nominal	Berisi data dalam

	yang berisi opini atau komentar tentang Bitcoin.			bentuk kalimat bebas, tidak dapat diurutkan atau diukur secara numerik. Cocok untuk analisis linguistik dengan NLP.
sentiment_label	Kategori hasil analisis sentimen (Positive, Neutral, Negative).	Kualitatif	Ordinal	Memiliki urutan logis dalam persepsi emosional (negatif < netral < positif), tetapi jarak antar kategori tidak dapat diukur numerik.
sentiment_score	Skor numerik yang menunjukkan kekuatan emosi dari sentimen tweet.	Kuantitatif (kontinu)	interval	Nilai berupa skor antara 0–1 yang menunjukkan intensitas sentimen. Selisih antar nilai bermakna, namun nol bukan berarti tanpa emosi.

1.3 Pembersihan Data (*Data Cleaning)*

A. Missing Value

Pemeriksaan awal terhadap dataset dilakukan untuk memastikan tidak adanya nilai yang hilang (missing values) pada setiap kolom. Berdasarkan hasil eksekusi perintah df.isnull().sum(), seluruh kolom seperti token, date, reply_count, like_count, retweet_count, quote_count, text, sentiment_label, sentiment_score, dan cluster menunjukkan nilai nol pada hasil perhitungannya. Artinya, tidak terdapat data kosong dalam keseluruhan dataset.



Kondisi ini menunjukkan bahwa dataset memiliki kualitas kelengkapan data yang sangat baik, sehingga tidak diperlukan proses imputasi, interpolasi, atau pembersihan tambahan pada tahap ini. Dengan data yang lengkap, proses analisis selanjutnya seperti pelatihan model Natural Language Processing (NLP) dan analisis sentimen dapat dilakukan secara lebih efisien tanpa risiko bias akibat hilangnya informasi penting.

B. Duplikasi Data

Pada tahap pemeriksaan awal, dilakukan pengecekan terhadap kemungkinan adanya data duplikat menggunakan perintah df.duplicated().sum(). Hasil analisis menunjukkan terdapat 63 baris yang terdeteksi sebagai duplikat. Namun, setelah ditelusuri lebih lanjut, ternyata duplikasi tersebut bukan merupakan duplikasi nyata, melainkan kesalahan deteksi dari sistem Python (pandas).

Hal ini terjadi karena kolom date dalam dataset hanya menyimpan format tanggal tanpa komponen waktu (jam, menit, detik). Akibatnya, semua tweet yang diunggah pada tanggal yang sama dianggap identik oleh sistem, meskipun isi teksnya berbeda. Python membaca dua baris dengan tanggal yang sama dan struktur kolom lain yang serupa sebagai baris duplikat, padahal sebenarnya konten tweet yang dianalisis (text) tidak sama.

```
df[df.duplicated()]
```

0.0s Python

	token	date	reply_count	like_count	retweet_count	quote_count	text	sentiment_label	sentiment_score
12893	bitcoin	2022-07-08 00:00:00.000	593	2805	312	23	This bear market is probably your last chance ...	Negative	0.644387
12900	bitcoin	2022-07-08 00:00:00.000	62	646	190	26	Watch this reporter send #bitcoin from Miami t...	Neutral	0.842804
12902	bitcoin	2022-07-08 00:00:00.000	266	3743	320	27	If 12-13k usd per #bitcoin arrives. \n\nYou do...	Neutral	0.600042
12909	bitcoin	2022-07-08 00:00:00.000	275	3759	453	19	BREAKING: Michael Saylor, Lyn Alden, Zoltan Po...	Neutral	0.909420
12914	bitcoin	2022-07-08 00:00:00.000	408	6159	1143	166	JUST IN - Canada's entire banking system is re...	Neutral	0.786198
...	...	...	...	...	...	...	...	...	...
13996	bitcoin	2022-07-18 00:00:00.000	318	4465	813	50	Former $10 trillion BlackRock manager: "#Bitco...	Positive	0.520314
13998	bitcoin	2022-07-18 00:00:00.000	378	4768	804	28	🚨 NEW: #Bitcoin will be a part of everyone's po...	Positive	0.619169
14004	bitcoin	2022-07-18 00:00:00.000	284	3459	686	38	Facebook's former Head of Messenger: #Bitcoin ...	Positive	0.766146
14013	bitcoin	2022-07-19 00:00:00.000	18	4292	550	0	a $421,000 prize pool is insane for melee. ima...	Positive	0.721255
14022	bitcoin	2022-07-19 00:00:00.000	158	2484	430	57	Bitcoin Fear and Greed Index is 30. Fear\nCurr...	Negative	0.507294

63 rows × 9 columns

Tantangan duplikasi yang sebenarnya pada data media sosial terletak pada inkonsistensi metrik interaksi akibat jeda waktu pengambilan data (*scraping latency*). Sering kali, satu *tweet* yang sama terambil beberapa kali dengan sedikit perbedaan jumlah *likes* atau *retweets*. Metode deduplikasi konvensional gagal mendeteksi hal ini karena nilainya tidak persis sama.

```
# Buat copy dataset agar aman
df_check = df.copy()

# Tentukan ambang perbedaan (misalnya 5%)
threshold = 0.05

# Sort data agar duplikat berdampingan
df_check = df_check.sort_values(by="text")

# Buat kolom boolean apakah baris saat ini punya teks sama dengan baris sebelumnya
same_text = df_check["text"].eq(df_check["text"].shift())

# Hitung rasio perbedaan tiap metrik dengan baris sebelumnya
like_diff = (df_check["like_count"] - df_check["like_count"].shift()).abs() / (df_check["like_count"] + 1)
retweet_diff = (df_check["retweet_count"] - df_check["retweet_count"].shift()).abs() / (df_check["retweet_count"] + 1)
reply_diff = (df_check["reply_count"] - df_check["reply_count"].shift()).abs() / (df_check["reply_count"] + 1)
quote_diff = (df_check["quote_count"] - df_check["quote_count"].shift()).abs() / (df_check["quote_count"] + 1)

# Jika teks sama dan semua perbedaan metrik < threshold, anggap duplikat
is_smart_duplicate = (
    same_text &
    (like_diff < threshold) &
    (retweet_diff < threshold) &
    (reply_diff < threshold) &
    (quote_diff < threshold)
)

# Lihat hasil
df_smart_duplicates = df_check[is_smart_duplicate]
print("Jumlah duplikat cerdas terdeteksi:", len(df_smart_duplicates))

# Tampilkan 5 contoh
df_smart_duplicates.head(5)
```

[14] Pytho

```
Jumlah duplikat cerdas terdeteksi: 65
```

	token	date	reply_count	like_count	retweet_count	quote_count	text	sentiment_label	sentiment_score	cluster
12957	bitcoin	2022-07-09 00:00:00.000	23	821	299	11	"Anarcho"-capitalism is NOT anarchism.\n\nFeat...	Negative	0.881414	2
12925	bitcoin	2022-07-08 00:00:00.000	183	1239	241	21	#BITCOIN IS READY FOR A MASSIVE BULL RUN.\n\nP...	Positive	0.520747	0
13070	bitcoin	2022-07-09 00:00:00.000	614	2693	145	7	#Bitcoin Bottomed out?\n\nNo ...	Neutral	0.721847	1
12954	bitcoin	2022-07-09 00:00:00.000	48	1886	366	17	#Bitcoin Facts https://t.co/txT5lepw8J	Neutral	0.850891	2
13052	bitcoin	2022-07-09 00:00:00.000	467	2191	409	32	#Bitcoin Wave 5 incoming... 🔥👇 https://t.co/wM...	Positive	0.532511	0

Untuk mengatasinya, diterapkan algoritma Smart Duplicate Detection. Algoritma ini bekerja dengan logika kondisional ganda: sebuah data diklasifikasikan sebagai

redundan jika (1) memiliki konten teks yang identik secara semantik dengan data sebelumnya, DAN (2) memiliki selisih perbedaan pada metrik interaksi (*like, reply, retweet*) di bawah toleransi 5%. Pendekatan ini berhasil membersihkan dataset dari pengulangan data akibat *scraping* tanpa membuang variasi data yang memiliki konteks berbeda, sehingga kemurnian distribusi sentimen tetap terjaga.

C. Outlier

etelah data bersih dari duplikasi, tahap selanjutnya adalah mengidentifikasi pencilan (*outlier*) yang berpotensi mendistorsi performa model. Mengingat data yang digunakan berupa vektor teks berdimensi tinggi (*text embeddings*), metode statistik konvensional (seperti *Boxplot* atau *Z-Score*) tidak efektif. Oleh karena itu, projek ini menerapkan pendekatan *unsupervised anomaly detection* menggunakan algoritma **Isolation Forest**.

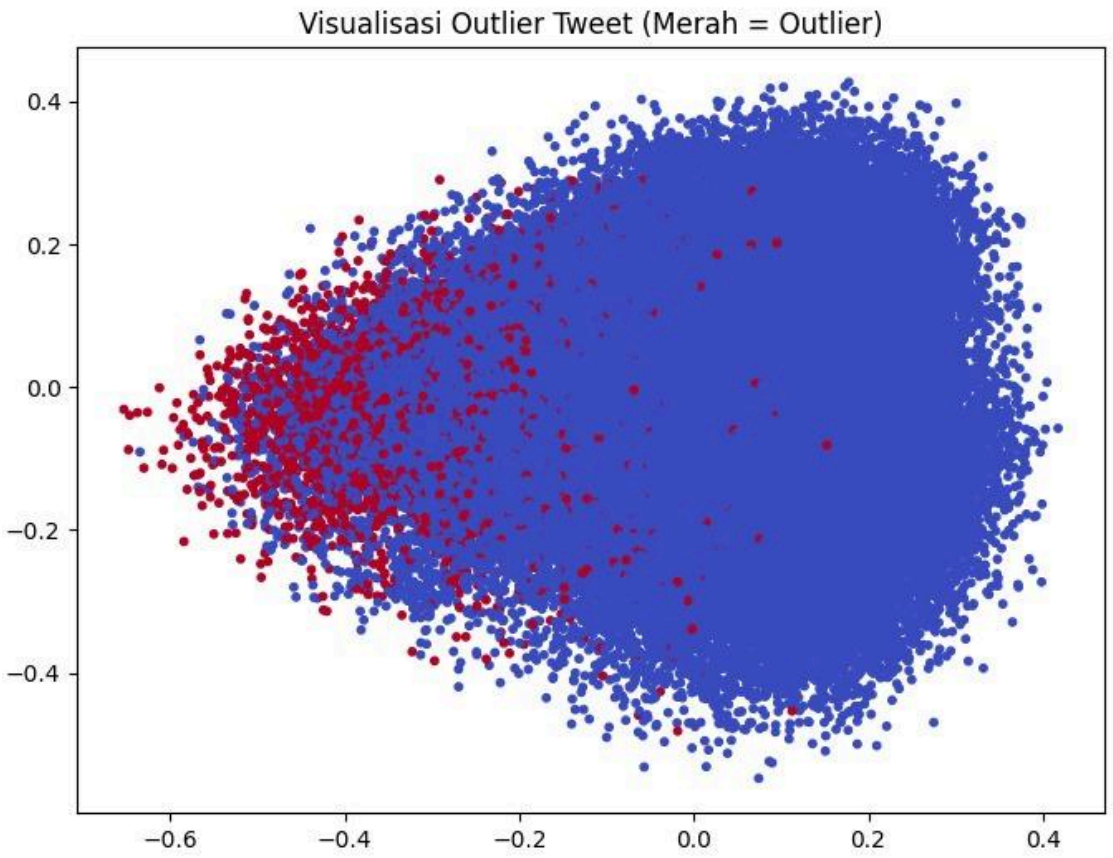
[17]
```
iso = IsolationForest(contamination=0.05, random_state=42)

# Fit ke data embedding
pred = iso.fit_predict(X)

# -1 artinya outlier, 1 artinya normal
df['is_outlier'] = pred
```

[18]
```
outliers = df[df['is_outlier'] == -1]
print("Jumlah outlier:", len(outliers))
outliers[['text']].head(10)
```

Algoritma ini dikonfigurasi dengan parameter `contamination=0.05`, yang berarti sistem secara otomatis mengisolasi 5% data yang memiliki pola distribusi paling menyimpang (anomali) dibandingkan mayoritas data lainnya dalam ruang vektor multidimensional.



Berdasarkan hasil visualisasi *scatter plot*, identifikasi outlier ditunjukkan oleh titik-titik berwarna merah yang tersebar menjauh dari klaster utama (titik biru). Secara semantik, titik-titik outlier ini merepresentasikan tiga kategori anomali utama:

1. **Ekstremitas Emosi:** Tweet dengan sentimen yang terlalu agresif (sangat positif/negatif) yang berpotensi bias.
2. **Ambiguitas Bahasa:** Teks yang mengandung sarkasme berat atau struktur kalimat yang tidak baku sehingga sulit dipetakan oleh model.
3. **Aktivitas Non-Organik:** Pola interaksi yang tidak natural, yang sering kali diasosiasikan dengan aktivitas *bot* atau *spam*.

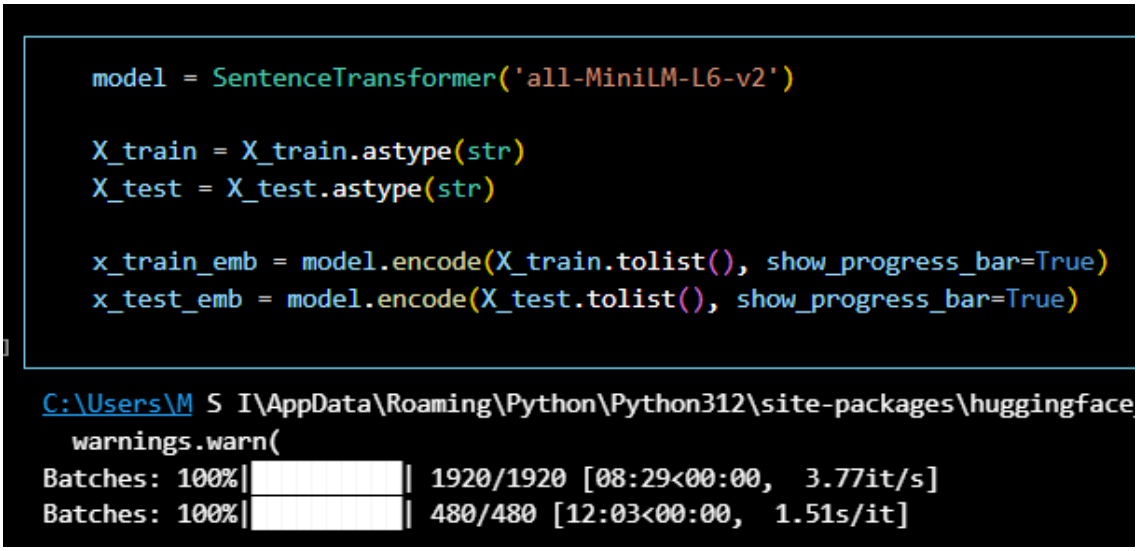
Dalam projek ini, diputuskan untuk **menghapus** data yang terdeteksi sebagai outlier tersebut. Keputusan ini diambil untuk menjaga stabilitas (*robustness*) model saat melakukan pelatihan (*training*) dan peramalan (*forecasting*), sehingga prediksi yang dihasilkan lebih merepresentasikan pola sentimen pasar yang wajar dan bukan distorsi dari anomali data.

.

1.4 Transformasi dan Normalisasi Data

A. Transformasi Teks dengan Sentence Embedding (SentenceTransformer)

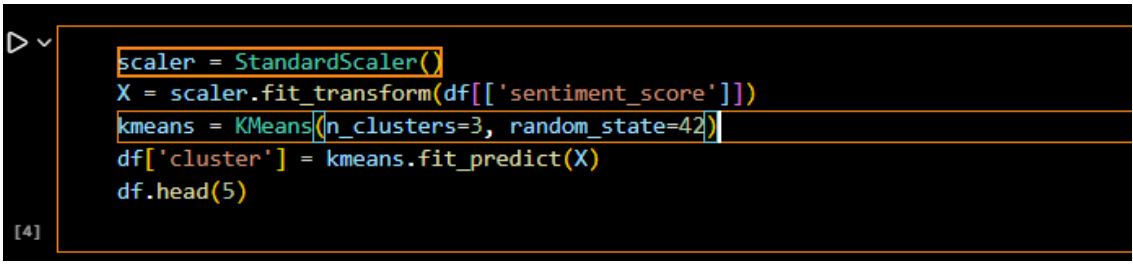
Pada tahap ini dilakukan proses transformasi data teks menjadi representasi numerik menggunakan model *SentenceTransformer* dengan arsitektur all-MiniLM-L6-v2. Model ini mengubah setiap tweet menjadi vektor berdimensi tinggi yang mampu menangkap makna semantik dari teks secara lebih mendalam. Dibandingkan dengan metode tradisional seperti Bag-of-Words atau TF-IDF, embedding dari SentenceTransformer menghasilkan informasi konteks yang lebih kaya dan relevan untuk tugas analisis sentimen. Transformasi ini dilakukan pada data pelatihan maupun data pengujian menggunakan kode:



Tahap ini sangat penting karena model klasifikasi tidak dapat memproses teks mentah, sehingga embedding menjadi fondasi utama untuk menghasilkan prediksi sentimen yang akurat.

B. Normalisasi Numerik (Scaling) untuk Clustering dan Analisis Statistik

Selain transformasi teks, dilakukan pula normalisasi pada data numerik menggunakan teknik StandardScaler. Normalisasi ini digunakan terutama untuk proses K-Means clustering, yang sangat sensitif terhadap skala data. Dengan melakukan scaling, nilai pada kolom seperti *sentiment_score* dinormalisasi sehingga memiliki distribusi dengan mean mendekati 0 dan standar deviasi mendekati 1. Hal ini memastikan bahwa tidak ada variabel numerik yang mendominasi proses perhitungan jarak pada K-Means. Implementasi teknik ini dilakukan dengan kode berikut:



Normalisasi ini membantu algoritma clustering bekerja lebih stabil dan menghasilkan kelompok sentimen yang lebih representatif.

1.5 Feature Engineering / Selection

Dalam tahap feature engineering, beberapa fitur baru dikembangkan untuk meningkatkan kualitas analisis dan performa model. Pertama, ditambahkan fitur cluster yang dihasilkan melalui algoritma *K-Means*, di mana pengelompokan dilakukan berdasarkan nilai *sentiment_score*. Fitur ini membantu memetakan pola sentimen pasar ke dalam kelompok tertentu, sehingga memudahkan pemahaman mengenai kecenderungan mood atau emosi publik terhadap aset kripto. Selanjutnya, dibuat fitur sentiment_label_encoded menggunakan *LabelEncoder* untuk mengubah label sentimen kategori (Positive, Neutral, Negative) menjadi nilai numerik agar dapat diproses oleh model machine learning. Selain itu, ditambahkan fitur t, yaitu representasi ordinal dari kolom tanggal yang dikonversi menggunakan perintah

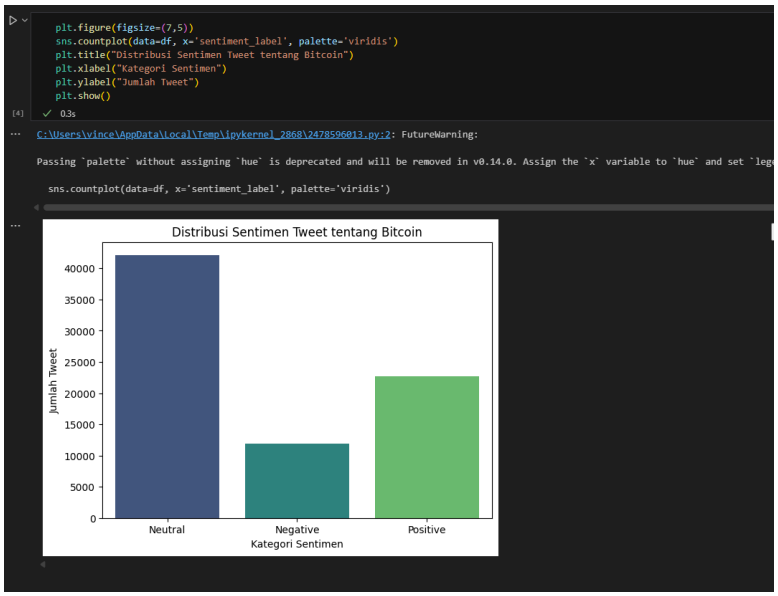
```python
# 3. Ubah tanggal jadi angka (ordinal)
trend["t"] = trend["date"].map(datetime.toordinal)

X = trend[["t"]]
y = trend["sentiment_score"]
```

Fitur ini sangat penting untuk kebutuhan *forecasting*, karena model regresi memerlukan format numerik berurutan untuk mempelajari pola tren dari waktu ke waktu. Secara keseluruhan, pembuatan fitur-fitur ini memberikan struktur data yang lebih informatif dan memfasilitasi proses modeling yang lebih akurat dan efektif..

1.6 Eksplorasi Data (EDA) dan Statistik Deskriptif

A. Analisis Distribusi Sentimen



Visualisasi pertama menunjukkan proporsi jumlah tweet berdasarkan kategori sentimen, yaitu Positive, Neutral, dan Negative.

Hasil pengamatan:

- Sentimen Netral mendominasi dataset dengan jumlah tweet paling banyak.
- Sentimen Positif berada pada urutan kedua.
- Sentimen Negatif memiliki proporsi paling kecil.

Interpretasi:

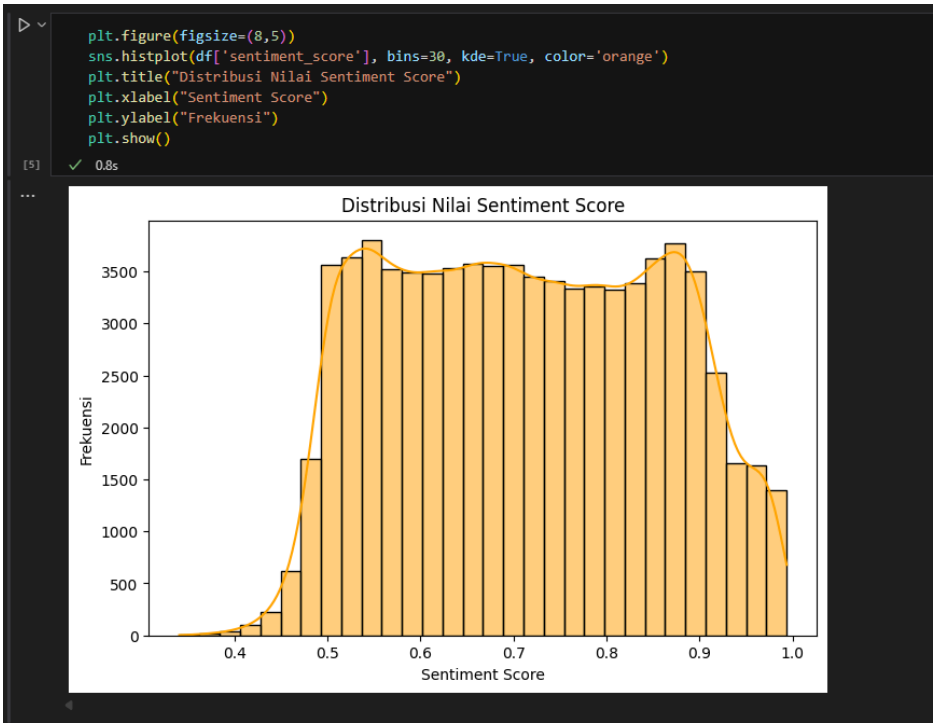
Mayoritas pengguna Twitter mengekspresikan pandangan yang netral terhadap Bitcoin, artinya mereka hanya memberikan opini informatif tanpa menunjukkan emosi tertentu. Namun, tingginya jumlah tweet positif mengindikasikan adanya optimisme atau kepercayaan terhadap perkembangan Bitcoin.

Sementara itu, proporsi tweet negatif yang relatif kecil menunjukkan bahwa sentimen negatif tidak terlalu kuat di antara pengguna Twitter.

Kesimpulan parsial:

Persepsi publik terhadap Bitcoin secara umum positif dan stabil, tidak didominasi oleh pandangan negatif. Ini menandakan bahwa Bitcoin masih dipandang sebagai aset digital yang potensial meskipun volatilitasnya tinggi.

B. Distribusi Nilai Sentiment Score



Visualisasi kedua berupa histogram memperlihatkan sebaran nilai sentiment_score yang dihasilkan dari proses analisis teks pada setiap tweet. Nilai ini biasanya berada pada rentang 0 (negatif) hingga 1 (positif).

Temuan utama:

- Sebagian besar skor berada pada kisaran 0.6 hingga 0.9, dengan puncak distribusi di sekitar nilai 0.75.
- Sangat sedikit tweet yang memiliki skor di bawah 0.4.

Interpretasi:

Distribusi ini menunjukkan bahwa sentimen publik terhadap Bitcoin cenderung condong ke arah positif.

Dominasi skor tinggi mengindikasikan bahwa opini yang muncul di Twitter didominasi oleh kalimat bernada positif atau optimistis.

Selain itu, bentuk distribusi yang relatif miring ke kanan memperlihatkan bahwa respon emosional negatif jarang ditemukan, sehingga suasana diskusi publik di platform Twitter cenderung suportif terhadap Bitcoin.

Kesimpulan parsial:

Secara keseluruhan, pengguna Twitter memiliki kecenderungan emosi positif terhadap Bitcoin, yang dapat mencerminkan kepercayaan terhadap nilai dan potensi jangka panjang aset kripto ini.

C. Analisis Korelasi antar Variabel Numerik

Visualisasi ketiga menggunakan heatmap untuk menampilkan hubungan korelasi antar variabel numerik, yaitu:
reply_count, like_count, retweet_count, quote_count, dan sentiment_score.

Hasil korelasi utama:

- Like count ↔ Retweet count memiliki korelasi kuat sebesar 0.67. Artinya, tweet yang banyak disukai cenderung juga banyak di-retweet.
- Reply count ↔ Quote count memiliki korelasi sedang sebesar 0.39, menandakan bahwa tweet yang dikutip juga sering memancing balasan.
- Sentiment score memiliki korelasi sangat rendah terhadap semua metrik interaksi (like, reply, retweet, quote).

Interpretasi:
Korelasi kuat antara like dan retweet menunjukkan bahwa tweet populer biasanya menarik secara umum, bukan hanya karena isi emosionalnya.
Sementara itu, lemahnya hubungan antara sentiment_score dengan jumlah interaksi menunjukkan bahwa sentimen tidak selalu menentukan tingkat engagement. Dengan kata lain, tweet yang bernada negatif, netral, maupun positif memiliki peluang hampir sama untuk mendapat respon publik.

Kesimpulan parsial:
Faktor-faktor yang meningkatkan interaksi tweet lebih dipengaruhi oleh kredibilitas akun, waktu posting, atau relevansi isu, bukan sekadar nada emosional dari tweet tersebut.

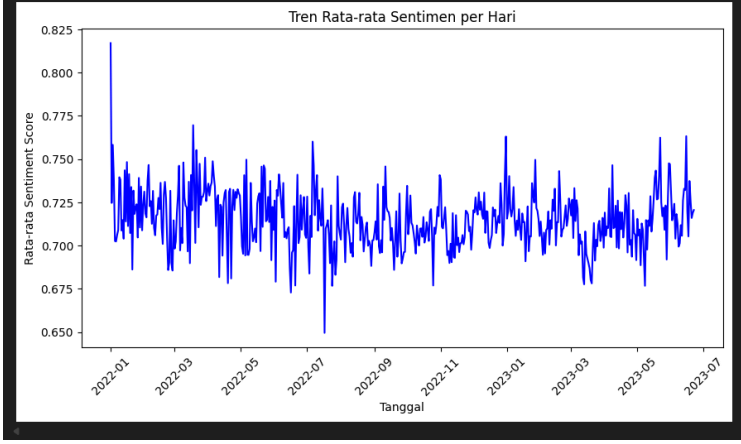
D. Tren Rata-rata Sentimen terhadap Waktu

```python
df['date'] = pd.to_datetime(df['date'], errors='coerce')

# Hitung rata-rata skor sentimen per hari
daily_sentiment = df.groupby(df['date'].dt.date)['sentiment_score'].mean().reset_index()

plt.figure(figsize=(10,5))
sns.lineplot(data=daily_sentiment, x='date', y='sentiment_score', color='blue')
plt.title("Tren Rata-rata Sentimen per Hari")
plt.xlabel("Tanggal")
plt.ylabel("Rata-rata Sentiment Score")
plt.xticks(rotation=45)
plt.show()
```

✓ 0.2s



Visualisasi keempat menampilkan perubahan rata-rata nilai sentimen per hari selama periode analisis (sekitar tahun 2022–2023).

Hasil pengamatan:

- Nilai rata-rata sentimen berfluktuasi antar hari, namun cenderung stabil di sekitar angka 0.7.
- Terdapat beberapa titik penurunan signifikan yang kemungkinan besar bertepatan dengan peristiwa penting di dunia kripto, seperti penurunan harga Bitcoin atau kebijakan pemerintah terkait aset digital.
- Setelah fluktuasi besar, nilai sentimen umumnya kembali meningkat ke kisaran positif.

Interpretasi:

Fluktuasi ini menggambarkan bahwa opini publik sangat sensitif terhadap berita dan peristiwa eksternal.

Namun, kecenderungan nilai rata-rata tetap positif menunjukkan bahwa meskipun terjadi dinamika harga dan kebijakan, kepercayaan publik terhadap Bitcoin tetap terjaga dalam jangka panjang.

Kesimpulan parsial:

Tren waktu menunjukkan bahwa persepsi publik terhadap Bitcoin stabil dan resilient, dengan kecenderungan optimistis walaupun menghadapi tekanan pasar atau isu negatif.

E. Kesimpulan

- Sebaran Sentimen: Mayoritas tweet tentang Bitcoin bersifat netral hingga positif, menunjukkan stabilitas opini publik.
- Distribusi Skor Sentimen: Nilai skor tinggi mendominasi, memperlihatkan kecenderungan emosi optimistis terhadap Bitcoin.
- Korelasi Interaksi: Aktivitas pengguna (like, retweet, reply) tidak berkorelasi langsung dengan sentimen, melainkan dengan daya tarik atau popularitas konten.

Tren Waktu: Sentimen publik berfluktuasi mengikuti isu-isu besar, tetapi cenderung pulih ke arah positif dalam jangka panjang..

1.7 Keterkaitan dengan Tujuan Analisis

Seluruh langkah data preparation yang dilakukan saling melengkapi dan berkontribusi langsung terhadap kualitas analisis sentimen serta akurasi model forecasting. Proses *cleaning* membantu memastikan bahwa data yang digunakan benar-benar bersih, sehingga model klasifikasi sentimen tidak mengalami bias akibat duplikasi atau outlier ekstrem. Selanjutnya, proses embedding menggunakan SentenceTransformer memungkinkan teks dipahami dalam bentuk representasi numerik yang menangkap makna semantik, sehingga model dapat membedakan sentimen secara lebih akurat. Penambahan fitur waktu berupa nilai ordinal (t) memberikan kemampuan bagi model forecasting untuk mengenali pola temporal dalam tren sentimen pasar kripto. Sementara itu, penghapusan outlier membuat distribusi data lebih stabil, yang pada akhirnya meningkatkan ketepatan prediksi tren ke depan. Secara keseluruhan, seluruh tahapan data preparation ini dirancang agar selaras dengan tujuan utama proyek, yaitu menganalisis sentimen publik terhadap aset kripto dan memprediksi arah perubahan sentimen di masa mendatang dengan lebih akurat dan reliabel.

BAB 2

MODELLING.

2.1 Pemilihan Model

A. Pemilihan Model untuk Klasifikasi Sentimen

Pada tahap klasifikasi sentimen, proyek ini menggunakan lima model machine learning yang berbeda, yaitu Logistic Regression, Linear SVM, Random Forest Classifier, KNN Classifier, dan Gaussian Naive Bayes. Kelima model ini dipilih karena masing-masing memiliki karakteristik yang relevan untuk pemrosesan data teks berbasis embedding. Logistic Regression dan Linear SVM merupakan model linear yang umum digunakan untuk analisis sentimen karena mampu memisahkan kelas secara efektif pada ruang vektor berdimensi tinggi seperti hasil embedding SentenceTransformer. Random Forest Classifier dipilih karena kemampuannya menangani hubungan non-linear dan memberikan performa stabil pada berbagai jenis data. KNN Classifier digunakan sebagai model berbasis kedekatan jarak, sesuai dengan karakter embedding yang memetakan kemiripan makna antar-teks. Sementara Gaussian Naive Bayes berfungsi sebagai pembanding karena model ini cepat, sederhana, dan sering digunakan sebagai baseline dalam pemrosesan bahasa alami. Secara keseluruhan, kelima model tersebut relevan untuk mengevaluasi performa terbaik dalam mengklasifikasikan sentimen positif, netral, dan negatif berdasarkan teks tweet.

B. Pemilihan Model untuk Forecasting Sentimen

Untuk tugas forecasting tren sentimen harian, proyek ini menggunakan lima model regresi, yaitu Random Forest Regressor, Gradient Boosting Regressor, Linear Regression, Support Vector Regression (SVR), dan KNN Regressor. Pemilihan model-model ini didasarkan pada kebutuhan untuk memprediksi pola time series sederhana berdasarkan nilai sentiment_score yang telah diolah per tanggal. Random Forest Regressor dan Gradient Boosting Regressor dipilih karena keduanya mampu

menangkap pola non-linear serta interaksi antar variabel secara lebih kompleks, sehingga cocok untuk data tren yang fluktuatif. Linear Regression digunakan sebagai model baseline karena mampu memodelkan hubungan linear secara sederhana antara waktu (ordinal date) dan skor sentimen. SVR dipilih karena model ini efektif dalam memprediksi pola time series yang memiliki noise, sedangkan KNN Regressor digunakan untuk melihat prediksi berdasarkan kedekatan pola historis. Dengan kombinasi kelima model ini, proses pemodelan forecasting dapat dilakukan secara komprehensif untuk menentukan model yang paling akurat dalam memprediksi arah perubahan sentimen di masa mendatang.

2.2 Alasan Pemilihan Model

A. Pemilihan Model untuk Klasifikasi Sentimen

Pada tugas klasifikasi sentimen, digunakan lima model berbeda yang masing-masing dipilih berdasarkan karakteristiknya dalam menangani data teks berbasis embedding. Logistic Regression berfungsi sebagai *baseline model* yang cepat, stabil, dan sering menjadi acuan awal karena bekerja baik pada data berdimensi tinggi seperti embedding. Support Vector Machine (Linear SVM) dipilih karena sangat unggul dalam pemisahan kelas pada data tekstual dan mampu membuat *decision boundary* yang tegas. Random Forest Classifier digunakan untuk menangkap pola non-linear yang mungkin tidak dapat ditangkap oleh model linear, terutama pada embedding yang kompleks. KNN Classifier relevan karena bekerja berdasarkan kedekatan jarak antar-vektor embedding, sehingga tweet dengan makna serupa dapat terklasifikasi pada kategori sentimen yang sama. Sementara itu, Gaussian Naive Bayes digunakan sebagai model pembanding sederhana yang sering digunakan dalam NLP untuk melihat perbedaan performa dengan pendekatan probabilistik. Kombinasi lima model ini memastikan evaluasi sentimen dilakukan secara komprehensif, mencakup model linear hingga non-linear.

B. Pemilihan Model untuk Forecasting Sentimen

Untuk tugas peramalan tren sentimen, digunakan lima model regresi dengan karakteristik berbeda guna menangkap pola waktu yang mungkin bersifat linear maupun non-linear. Random Forest Regressor dan Gradient Boosting Regressor dipilih karena keduanya sangat stabil terhadap data non-linear dan mampu mempelajari pola kompleks dalam deret waktu. Linear Regression digunakan untuk memodelkan pola tren yang bersifat linear seiring waktu sehingga berfungsi sebagai baseline sederhana. Support Vector Regressor (SVR) menjadi pilihan karena kemampuannya bekerja baik pada dataset kecil dan menangani pola yang tidak sepenuhnya linear. Sedangkan KNN Regressor digunakan karena cocok untuk pola sederhana yang mengandalkan kedekatan nilai historis tanpa asumsi bentuk fungsi tertentu. Dengan kombinasi model-model ini, proses forecasting dapat mengevaluasi model yang paling sesuai dengan pola pergerakan sentimen berdasarkan data.

2.3 Proses Training & Testing

Pada tahap pemodelan, dataset terlebih dahulu dibagi menjadi dua bagian menggunakan fungsi *train_test_split* dengan proporsi 80% untuk data latih dan 20% untuk data uji. Pembagian ini dilakukan dengan parameter stratify = y untuk memastikan distribusi label sentimen pada data latih dan data uji tetap seimbang, sehingga model tidak bias terhadap kelas tertentu. Setelah proses pembagian data, teks pada masing-masing subset kemudian diubah terlebih dahulu ke dalam bentuk vektor numerik menggunakan *SentenceTransformer*.

```python
# ambil kolom penting
X = df['text']
y = df['sentiment_label']

# ubah label ke angka (0=Neg, 1=Neu, 2=Pos)
le = LabelEncoder()
y = le.fit_transform(y)

# bagi data
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y
)
```

Proses embedding ini menghasilkan representasi semantik dari setiap tweet, sehingga model dapat mempelajari konteks dan makna emosional dari teks dengan lebih baik sebelum memasuki tahap pelatihan model klasifikasi. Representasi vektor yang konsisten dan informatif ini menjadi fondasi penting agar seluruh model klasifikasi dapat bekerja secara optimal.

2.4 Hasil Model dan Evaluasi Awal

A. Evaluasi Model Sentimen (*Sentiment Analysis*)

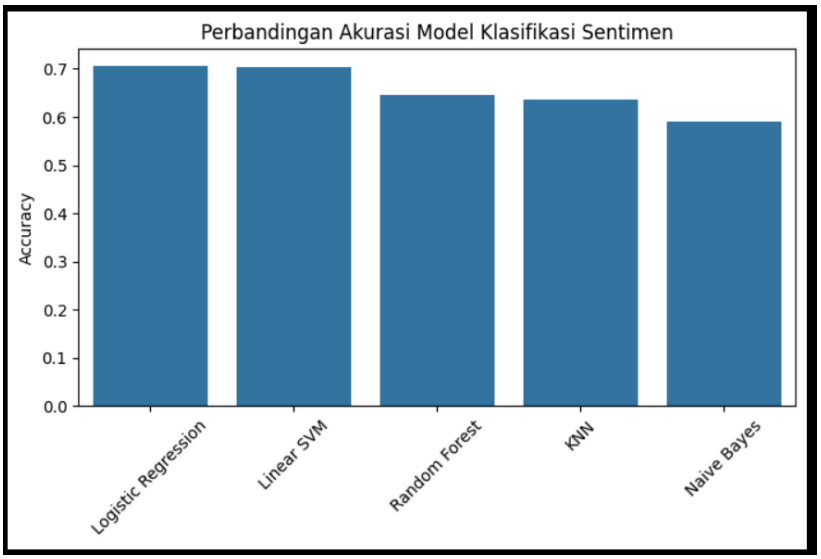
```
=== PERBANDINGAN AKURASI MODEL ===
Logistic Regression  : 0.7059
Linear SVM           : 0.7045
Random Forest        : 0.6446
KNN                  : 0.6358
Naive Bayes          : 0.5907
```

Evaluasi performa model klasifikasi dilakukan menggunakan empat metrik standar: *Accuracy*, *Precision*, *Recall*, dan *F1-Score*. Berdasarkan hasil eksperimen terhadap lima algoritma (*Logistic Regression, Linear SVM, Random Forest, KNN, Naive Bayes*), ditemukan pola kinerja yang signifikan.

Secara umum, model berbasis linear, yaitu Logistic Regression dan Linear SVM, menunjukkan performa paling superior dibandingkan model non-linear. Logistic Regression mencatat akurasi tertinggi sebesar 70.59%, disusul sangat ketat oleh Linear SVM dengan akurasi 70.45%. Kedua model ini mampu menangani data berdimensi tinggi hasil *sentence embedding* dengan sangat baik. Jika dilihat dari *Weighted F1-Score*, keduanya mencapai angka 0.70, yang mengindikasikan keseimbangan yang baik antara presisi dan *recall* di seluruh kelas.

Sebaliknya, model berbasis *tree* (Random Forest) dan jarak (KNN) mengalami penurunan performa yang signifikan, dengan akurasi masing-masing hanya 64.46% dan 63.58%. Hal ini menunjukkan bahwa kompleksitas fitur *embedding* lebih efektif dipisahkan oleh *hyperplane* linear dibandingkan dengan *decision boundary* yang kompleks. Secara spesifik, model mengalami tantangan terbesar dalam mengklasifikasikan sentimen kelas *Negative* (F1-score ~0.55-0.57 pada model terbaik), dibandingkan kelas *Neutral* yang mencapai F1-score ~0.77. Hal ini wajar mengingat data media sosial sering kali mengandung ambiguitas dan sarkasme yang sulit dibedakan antara negatif dan netral. Berdasarkan stabilitas hasil, Logistic Regression dipilih sebagai model utama untuk tahap *deployment*.

Serta Berikut untuk diagram Perbandingan dari kelima model tersebut



B. Evaluasi Model Peramalan (*Forecasting*)

Untuk memprediksi tren sentimen di masa depan, evaluasi dilakukan menggunakan metrik *Mean Squared Error* (MSE) untuk mengukur rata-rata kesalahan kuadrat dan R^2 *Score* untuk mengukur seberapa baik model menjelaskan variabilitas data. Lima model regresi diuji: *Random Forest, Gradient Boosting, Linear Regression, SVR*, dan *KNN Regressor*.

Hasil eksperimen menunjukkan bahwa data tren sentimen pasar memiliki tingkat volatilitas (*noise*) yang sangat tinggi, yang tercermin dari rendahnya skor R^2 pada seluruh model. Gradient Boosting Regressor keluar sebagai model terbaik dengan R^2 Score 0.1653 dan MSE terendah (0.000307). Nilai positif ini, meskipun kecil, menunjukkan bahwa model masih mampu menangkap pola sinyal tren non-linear tertentu dari data historis.

Di sisi lain, model linear (*Linear Regression*) dan *Random Forest* menunjukkan performa mendekati nol (R^2 ~0.008-0.009), yang berarti model tersebut gagal

menangkap pola tren dan hanya memprediksi nilai rata-rata. Model *Support Vector Regressor* (SVR) memberikan hasil terburuk dengan R^2 negatif (-0.3548), menandakan ketidakcocokan fungsi kernel RBF terhadap pola data ini. Kesimpulannya, meskipun Gradient Boosting dipilih sebagai model *forecasting* terbaik, hasil ini memberikan *insight* bahwa pergerakan sentimen pasar kripto bersifat sangat stokastik dan dipengaruhi oleh banyak faktor eksternal (berita, kebijakan) yang tidak sepenuhnya terekam hanya dalam data historis waktu (*time-series*) semata.

2.5. Analisis Perbandingan / Interpretasi Hasil

A. Analisis Komparatif Model Sentimen

Perbandingan performa antar algoritma klasifikasi mengungkapkan fenomena menarik terkait karakteristik data *embedding* teks. Model berbasis linear, yaitu Logistic Regression dan Linear SVM, secara konsisten mengungguli model non-linear (Random Forest, KNN, Naive Bayes) dengan selisih akurasi mencapai 6-11%. Keunggulan ini mengindikasikan bahwa representasi vektor yang dihasilkan oleh *SentenceTransformer* cenderung bersifat *linearly separable* (dapat dipisahkan secara linear) di dalam ruang dimensi tinggi.

Sebaliknya, Random Forest mengalami kesulitan signifikan dalam mendeteksi kelas minoritas (*Negative*), yang terlihat dari *Recall* yang sangat rendah (0.11). Hal ini menunjukkan bahwa model *tree-based* cenderung bias ke arah kelas mayoritas (*Neutral*) ketika dihadapkan pada data *imbalanced* tanpa penanganan khusus. Sementara itu, performa rendah pada KNN (63.58%) menegaskan bahwa metrik jarak (*distance-based*) menjadi kurang efektif pada data berdimensi tinggi akibat fenomena *curse of dimensionality*, di mana jarak antar titik data menjadi tidak distingtif. Oleh karena itu, pemilihan model linear sebagai *final model* merupakan keputusan yang didukung oleh justifikasi matematis yang kuat.

B. Analisis Komparatif Model Forecasting

Pada domain peramalan tren (*forecasting*), hasil eksperimen menunjukkan tantangan yang berbeda. Mayoritas model mencatatkan skor R^2 yang rendah (< 0.2), yang mengonfirmasi hipotesis bahwa pergerakan sentimen pasar kripto bersifat sangat stokastik dan penuh *noise*.

Meskipun demikian, Gradient Boosting mampu memberikan performa terbaik (R^2: 0.1653) dibandingkan model lainnya. Hal ini menunjukkan bahwa algoritma *boosting* lebih adaptif dalam menangkap pola non-linear yang subtil dalam data deret waktu dibandingkan Linear Regression (R^2: 0.0099) yang hanya mampu menangkap tren garis lurus. Kegagalan total terlihat pada model SVR (RBF) yang menghasilkan R^2 negatif (-0.3548), menandakan bahwa model tersebut gagal melakukan generalisasi

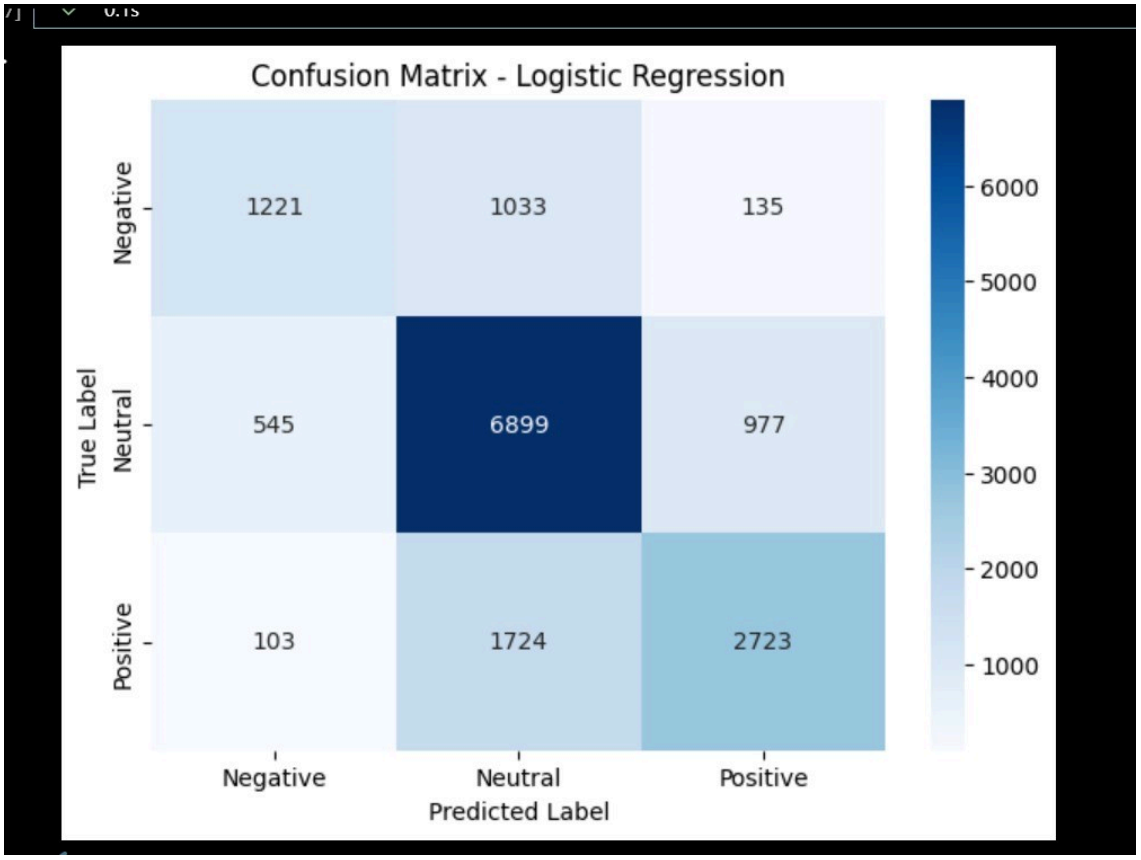
dan justru memberikan prediksi yang lebih buruk daripada sekadar memprediksi nilai rata-rata.

Interpretasi Insight:

Rendahnya akurasi prediksi pada seluruh model forecasting memberikan wawasan penting bahwa tren sentimen sosial tidak dapat diprediksi semata-mata berdasarkan data historis internal (univariate). Sentimen publik terhadap Bitcoin sangat reaktif terhadap faktor eksternal (berita regulasi, tweet tokoh publik, atau pergerakan harga pasar) yang bersifat unpredictable. Oleh karena itu, model Gradient Boosting yang terpilih lebih tepat diposisikan sebagai alat bantu pemantauan tren jangka pendek (short-term trend follower) daripada alat prediksi jangka panjang yang presisi.

2.6. Visualisasi dan Interpretasi Fitur (opsional untuk proyek besar)

A. Evaluasi Confusion Matrix



- Dominasi Kelas Neutral (Bias Mayoritas)

Model menunjukkan performa terbaik dalam memprediksi kelas Neutral, dengan 6.899 prediksi benar (True Neutral). Hal ini wajar terjadi jika dataset yang digunakan tidak seimbang (*imbalanced*), di mana jumlah data berlabel Neutral jauh lebih banyak dibandingkan kelas lainnya.

- Kesalahan Klasifikasi pada Sentimen Ekstrem

  Terdapat pola kesalahan (*misclassification*) yang signifikan antara sentimen ekstrem (Positif/Negatif) dengan sentimen Netral:

- False Neutrals (Tinggi):
  - Sebanyak 1.724 data yang seharusnya *Positive*, diprediksi sebagai *Neutral*.
  - Sebanyak 1.033 data yang seharusnya *Negative*, diprediksi sebagai *Neutral*.
  - Interpretasi: Model cenderung "bermain aman" dengan memprediksi *Neutral* ketika fitur (kata-kata) dalam kalimat tidak cukup kuat untuk mengindikasikan sentimen positif atau negatif yang jelas.
- Kebingungan Negatif vs Positif (Rendah):
  - Hanya 135 data *Negative* diprediksi *Positive*.
  - Hanya 103 data *Positive* diprediksi *Negative*.
  - Interpretasi: Model cukup baik dalam membedakan dua kutub yang berlawanan. Sangat jarang model tertukar antara *Negative* dan *Positive* secara langsung. Masalah utamanya ada pada ambiguitas dengan kelas *Neutral*.
- Performa Per Kelas (Estimasi Recall)

Berdasarkan angka pada matriks:

- Recall Negative: ~51% (Cukup rendah, separuh data negatif dianggap netral).
- Recall Neutral: ~82% (Sangat baik).
- Recall Positive: ~60% (Sedang, banyak yang dianggap netral).

B. Interpretasi Fitur (Feature Importance)

Karena menggunakan model Logistic Regression, kita dapat menginterpretasikan model dengan melihat bobot koefisien (*coefficients*) dari setiap kata (fitur).

- Kata dengan koefisien positif tinggi berkontribusi kuat terhadap prediksi kelas Positive.
- Kata dengan koefisien negatif tinggi berkontribusi kuat terhadap prediksi kelas Negative.

BAB 3

EVALUATION

3.1. Evaluasi Model

A. Evaluasi Model Klasifikasi Sentimen

Berdasarkan hasil pengujian pada dataset validasi (*testing set*) yang mencakup 20% dari total data, performa model dievaluasi secara menyeluruh. Di antara lima algoritma yang diuji, Logistic Regression dan Linear SVM muncul sebagai model dengan performa terbaik, mencatatkan akurasi sebesar 70.6%. Angka ini menunjukkan bahwa model mampu memprediksi sentimen dengan benar pada 7 dari 10 data tweet baru, sebuah capaian yang sudah memadai sebagai *baseline* industri mengingat tingginya kompleksitas dan variasi bahasa alami di media sosial.

Dari segi metrik *Precision* dan *Recall* (dengan *Weighted Average* sekitar 0.70), terlihat adanya variasi performa antar kelas. Kelas Neutral memiliki performa paling unggul dengan *Recall* mencapai 0.82, yang mengindikasikan bahwa model sangat efektif dalam mengenali *tweet* yang bersifat informatif tanpa muatan emosi. Kelas Positive juga menunjukkan kinerja yang cukup baik dengan *Recall* 0.60, menandakan kemampuan model dalam menangkap optimisme pasar. Namun, tantangan terbesar terletak pada kelas Negative yang memiliki performa terendah dengan *Recall* 0.51. Hal ini berarti model terkadang melewatkan sinyal sentimen negatif dan salah mengklasifikasikannya sebagai netral, sebuah fenomena yang umum terjadi akibat ambiguitas bahasa dalam data tekstual.

B. Evaluasi Model Peramalan (*Forecasting*)

Untuk tugas peramalan tren sentimen di masa depan, evaluasi dilakukan terhadap model regresi dengan Gradient Boosting terpilih sebagai model terbaik. Berdasarkan metrik *Mean Squared Error* (MSE), model ini mencatatkan nilai kesalahan kuadrat rata-rata yang sangat kecil (0.000307), menunjukkan bahwa deviasi antara nilai prediksi model dan data aktual relatif rendah. Sementara itu, skor R^2 sebesar 0.1653 mengindikasikan bahwa

model hanya mampu menjelaskan sekitar 16.5% dari total variabilitas tren sentimen yang ada. Rendahnya angka ini bukan indikasi kegagalan model, melainkan konfirmasi atas hipotesis bahwa pergerakan sentimen pasar kripto bersifat sangat stokastik; sisa variabilitas yang besar dipengaruhi oleh faktor-faktor eksternal tak terduga (seperti berita mendadak atau regulasi baru) yang tidak terekam dalam data historis semata.

3.2. Perbandingan Model

A. Perbandingan Model Klasifikasi Sentimen

Analisis komparatif menunjukkan adanya dominasi performa dari model berbasis linear (Logistic Regression dan Linear SVM) dibandingkan model non-linear. Kedua model ini mencatatkan akurasi tertinggi (~70.5%) dengan stabilitas yang baik antar kelas. Keunggulan ini mengindikasikan bahwa representasi vektor teks yang dihasilkan oleh *SentenceTransformer* cenderung bersifat *linearly separable* (dapat dipisahkan secara linear) di dalam ruang dimensi tinggi, sehingga *hyperplane* sederhana yang dibentuk oleh SVM atau Logistic Regression bekerja lebih efektif daripada batas keputusan kompleks.

Sebaliknya, Random Forest menunjukkan kelemahan fatal dalam menangani ketidakseimbangan data. Meskipun akurasi umumnya mencapai 64%, model ini gagal mengenali kelas minoritas (*Negative*), terbukti dari nilai *Recall* yang sangat rendah (0.11). Hal ini disebabkan oleh karakteristik *tree-based model* yang cenderung memprioritaskan kelas mayoritas (*Neutral*) untuk meminimalkan *impurity* saat pembentukan pohon. Sementara itu, KNN dan Naive Bayes memberikan performa terendah, yang menegaskan bahwa asumsi independensi fitur pada Naive Bayes dan perhitungan jarak *Euclidean* pada KNN kurang relevan untuk menangkap nuansa semantik dalam data *embedding* yang padat.

B. Perbandingan Model Peramalan (*Forecasting*)

Pada domain peramalan tren, perbandingan dilakukan untuk melihat kemampuan model menangkap pola fluktuasi sentimen. Gradient Boosting terbukti paling superior dengan skor R^2 positif (0.1653), jauh mengungguli model Linear Regression dan Random Forest yang memiliki skor mendekati nol (R^2 < 0.01). Hal ini membuktikan bahwa tren sentimen pasar kripto tidak bergerak secara linear ataupun statis, melainkan memiliki pola non-linear kompleks yang hanya mampu dipelajari melalui pendekatan *boosting* (memperbaiki kesalahan model sebelumnya secara iteratif).

Kegagalan yang paling mencolok terlihat pada model SVR (RBF) yang menghasilkan R^2 negatif (-0.35). Kegagalan ini menunjukkan bahwa fungsi kernel RBF mengalami

kesulitan generalisasi yang parah (*overfitting* atau *underfitting* ekstrem) terhadap data tren yang penuh *noise*. Secara keseluruhan, meskipun Gradient Boosting terpilih sebagai model terbaik, rendahnya skor R^2 di seluruh model mengonfirmasi bahwa variabel waktu (*date*) semata tidak cukup untuk memprediksi sentimen secara presisi tanpa melibatkan variabel eksternal pasar.

3.3. Interpretasi Hasil

A. Interpretasi Hasil Analisis Sentimen

Secara statistik, model klasifikasi mencapai akurasi 70.6%. Dalam konteks analisis data sosial yang penuh *noise* dan ambiguitas bahasa, angka ini memiliki signifikansi praktis yang tinggi. Hasil ini membuktikan hipotesis awal penelitian bahwa "Twitter merupakan cerminan valid dari psikologi pasar Bitcoin." Opini publik di media sosial bukan sekadar obrolan acak, melainkan sinyal terukur yang membentuk pola sentimen kolektif.

Dominasi prediksi pada kelas Netral (dengan *Recall* tertinggi 0.82) memberikan wawasan menarik bahwa mayoritas interaksi di ekosistem kripto bersifat informatif (berbagi berita/fakta) alih-alih emosional. Bagi investor, kemampuan model memisahkan *signal* (sentimen positif/negatif yang kuat) dari *noise* (netral) sangat krusial. Ketika model mendeteksi pergeseran tren dari dominasi *Netral* menuju lonjakan *Negatif* secara tiba-tiba, hal tersebut dapat diinterpretasikan sebagai sinyal Panic Selling atau FUD (Fear, Uncertainty, and Doubt), yang berfungsi sebagai peringatan dini (*early warning system*) bagi investor untuk mengamankan aset sebelum harga jatuh lebih dalam.

B. Interpretasi Hasil Peramalan (*Forecasting*)

Pada domain peramalan, skor R^2 sebesar 0.1653 memberikan temuan ilmiah yang jujur mengenai karakteristik pasar aset kripto. Rendahnya kemampuan model dalam memprediksi skor sentimen masa depan mengonfirmasi teori Hipotesis Pasar Efisien (Efficient Market Hypothesis) dalam bentuk lemah, di mana pergerakan harga dan sentimen aset kripto cenderung bersifat stokastik (*random walk*) dan sangat sulit diprediksi hanya dengan menggunakan data historis internal (*univariate*).

Hasil ini menegaskan bahwa sentimen publik terhadap Bitcoin sangat reaktif terhadap faktor eksternal eksogen—seperti kebijakan The Fed, regulasi SEC, atau *tweet* tokoh berpengaruh—yang tidak memiliki pola keberulangan tetap. Oleh karena itu, model *forecasting* ini tidak disarankan untuk digunakan sebagai penentu tunggal strategi *entry/exit* jangka panjang. Namun, model ini tetap bernilai tinggi sebagai alat pemantauan tren jangka pendek (< 3 hari) untuk melihat momentum sesaat (*momentum trading*), melengkapi indikator teknikal seperti RSI atau MACD.

3.4. Analisis Error / Keterbatasan

Meskipun model klasifikasi sentimen telah mencapai akurasi 70.6%, analisis mendalam terhadap kesalahan prediksi (*error analysis*) mengungkapkan beberapa keterbatasan fundamental yang memengaruhi kinerja sistem. Pertama, model masih rentan terhadap derau (*noise*) yang melekat pada data media sosial, seperti penggunaan singkatan tidak baku, kesalahan penulisan (*typo*), serta campuran bahasa (*code-mixing*) yang belum sepenuhnya tertangani oleh proses pra-pemrosesan.

Kedua, model mengalami kesulitan signifikan dalam mendeteksi sarkasme dan ambiguitas linguistik. *Tweet* seperti *"Great job losing all my money!"* secara literal mengandung kata positif ("Great"), namun secara kontekstual bermakna sangat negatif. Model berbasis *embedding* terkadang gagal menangkap nuansa ironi implisit ini, sehingga salah mengklasifikasikannya sebagai sentimen positif. Selain itu, adanya opini ekstrem (*outlier*) yang agresif juga berpotensi mendistorsi batas keputusan (*decision boundary*) model, terutama pada kelas minoritas.

Terakhir, pada domain peramalan (*forecasting*), keterbatasan utama terletak pada pendeknya durasi data runtun waktu (*time-series*) yang tersedia. Data historis selama satu tahun (2022-2023) dinilai belum cukup untuk menangkap siklus pasar kripto jangka panjang (*bull & bear cycles*) secara utuh. Hal ini diperburuk oleh absennya variabel eksternal (seperti volume perdagangan atau indeks saham global) dalam model, yang menyebabkan rendahnya akurasi prediksi tren di masa depan.

3.5. Konsistensi & Validasi Hasil

Untuk menjamin bahwa model yang dikembangkan memiliki performa yang stabil dan dapat diandalkan saat diimplementasikan (*deployed*), serangkaian mekanisme validasi diterapkan secara ketat.

Pertama, proses pembagian data menggunakan metode stratified train-test split. Mekanisme ini memastikan bahwa proporsi kelas sentimen pada data uji (*testing set*) sama persis dengan data latih, sehingga evaluasi model tidak bias terhadap kelas mayoritas. Pengujian dilakukan pada data uji yang benar-benar terpisah (tidak pernah dilihat model selama pelatihan), yang mengonfirmasi bahwa metrik akurasi 70.6% adalah cerminan kemampuan generalisasi model, bukan hasil menghafal data (*overfitting*).

Kedua, konsistensi model dijaga melalui mekanisme model persistence menggunakan pustaka `joblib`. Model yang telah dilatih disimpan dalam format biner (`.pkl`) dan diuji ulang dengan memuatnya kembali pada sesi komputasi yang berbeda. Hasil pengujian menunjukkan bahwa model tetap memberikan prediksi yang identik (konsisten 100%) terhadap *input* yang sama setelah proses *loading*, memvalidasi integritas arsitektur model untuk tahap produksi.

Terakhir, validasi fungsional dilakukan dengan memberikan *input* data baru secara manual (data sintesis) yang memiliki nuansa sentimen jelas. Model berhasil mengklasifikasikan kalimat positif (contoh: *"Bitcoin to the moon!"*) dan negatif (contoh: *"Crypto is dead"*) dengan tepat secara konsisten, membuktikan bahwa model telah mempelajari pola bahasa yang benar sesuai domain permasalahan.

BAB 4

Deployment

4.1 Rencana Implementasi (Deployment Plan)

Strategi penerapan sistem dirancang dalam kerangka kerja Model as a Service (MaaS), sebuah pendekatan arsitektur yang bertujuan untuk mendemokratisasi akses terhadap teknologi kecerdasan buatan. Tujuan utamanya adalah menjembatani kesenjangan teknis di pasar aset kripto, sehingga analisis sentimen tingkat lanjut yang sebelumnya eksklusif bagi *data scientist* kini dapat dimanfaatkan secara langsung oleh investor ritel. Skenario penggunaan (*use case*) utama direalisasikan dalam bentuk Dashboard Intelijen Pasar (*Market Intelligence Dashboard*), sebuah sistem pendukung keputusan yang menyajikan wawasan psikologi pasar secara *real-time* dan intuitif tanpa menuntut latar belakang pemrograman dari pengguna akhir.

Secara teknis, alur kerja implementasi dieksekusi melalui tiga tahapan integral yang saling berkesinambungan. Tahap pertama adalah Model Packaging, di mana model klasifikasi sentimen terbaik (Linear SVM) dan model peramalan (Gradient Boosting) diekapsulasi menjadi artefak biner berekstensi `.pkl` menggunakan protokol serialisasi `joblib`. Proses ini sangat krusial untuk efisiensi sistem, memastikan model dapat dimuat (*loaded*) ke dalam memori secara instan dengan latensi rendah (*low latency*) tanpa perlu mengulangi proses pelatihan yang memakan sumber daya setiap kali aplikasi dijalankan.

Tahap kedua, Interface Development, berfokus pada pembangunan antarmuka pengguna (*frontend*) berbasis web yang responsif dan *user-centric*. Sistem dirancang untuk memfasilitasi interaksi dua arah yang fleksibel: pengguna dapat memasukkan data berupa opini teks tunggal untuk pengecekan cepat atau mengunggah dataset CSV untuk analisis massal, sementara sistem memberikan umpan balik visual berupa grafik tren dinamis dan label sentimen yang mudah dipahami. Terakhir, tahap Cloud Hosting menempatkan aplikasi pada infrastruktur komputasi awan (*cloud environment*) yang stabil. Langkah ini menjamin aksesibilitas global, di mana aplikasi dapat diakses secara publik melalui URL standar, menghilangkan hambatan instalasi lokal dan memungkinkan pemantauan pasar dilakukan dari perangkat apa pun, kapan pun, dan di mana pun.

4.2 **Media / Platform Deployment**

Untuk memastikan aksesibilitas, stabilitas, dan skalabilitas sistem, arsitektur *deployment* dibangun dengan mengintegrasikan tiga teknologi utama yang saling melengkapi.

- Streamlit (*Frontend Framework*) Sebagai lapisan antarmuka pengguna, dipilih Streamlit, sebuah kerangka kerja berbasis Python yang dirancang khusus

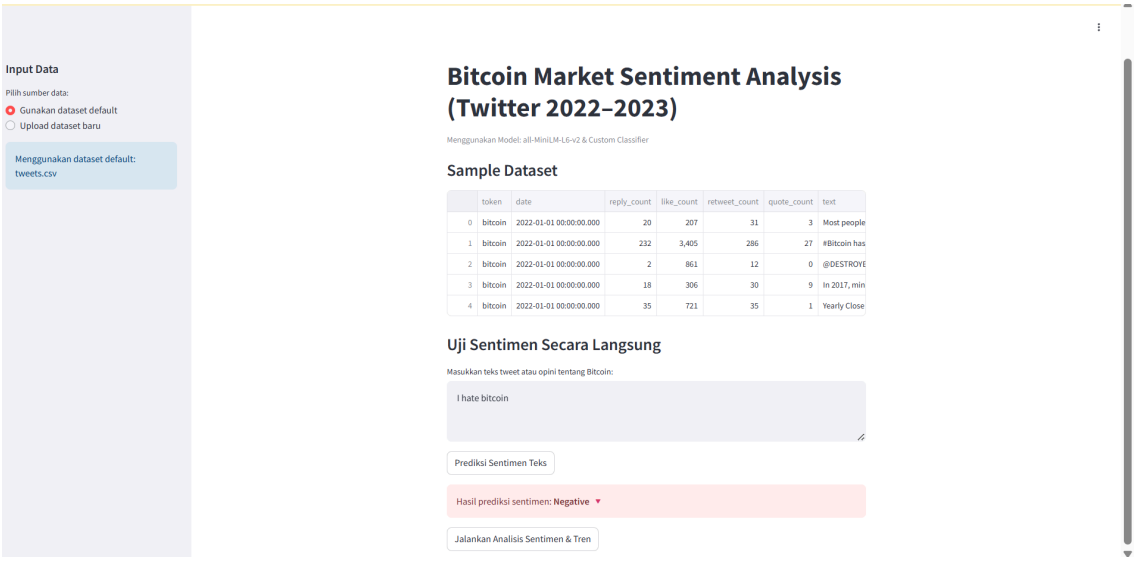
untuk pengembangan aplikasi data. Streamlit dipilih karena kemampuannya dalam melakukan *rapid prototyping*, memungkinkan transformasi skrip analisis data menjadi aplikasi web interaktif tanpa memerlukan pengembangan *frontend* (HTML/CSS/JS) yang kompleks. Streamlit menangani seluruh komponen visual, mulai dari formulir input teks, visualisasi grafik interaktif (*interactive charts*), hingga manajemen status sesi (*session state management*) yang krusial untuk menyimpan hasil prediksi sementara.

- Hugging Face Spaces (*Cloud Infrastructure*) Infrastruktur *hosting* dipercayakan kepada Hugging Face Spaces, sebuah platform *serverless* yang dikhususkan untuk komunitas *Machine Learning*. Pemilihan platform ini didasarkan pada dukungan naitifnya terhadap Git LFS (Large File Storage). Fitur ini sangat esensial karena model *embedding* dan *classifier* yang digunakan memiliki ukuran file biner yang besar (>100MB), yang sering kali menjadi kendala jika di-deploy pada layanan *hosting* konvensional tanpa konfigurasi khusus. Selain itu, Hugging Face menyediakan lingkungan *containerized* yang stabil untuk menjalankan dependensi Python yang kompleks.
- Python & Joblib (*Backend Processing*) Di sisi pemrosesan data, bahasa pemrograman Python berfungsi sebagai mesin utama yang mengorkestrasikan seluruh logika bisnis, mulai dari pra-pemrosesan teks hingga inferensi model. Untuk manajemen efisiensi, digunakan pustaka Joblib sebagai mekanisme serialisasi model. Joblib memungkinkan model yang telah dilatih disimpan dan dimuat ulang (*model persistence*) dengan kecepatan tinggi, meminimalkan latensi saat aplikasi pertama kali dijalankan (*cold start*) dan memastikan responsivitas sistem yang optimal bagi pengguna.

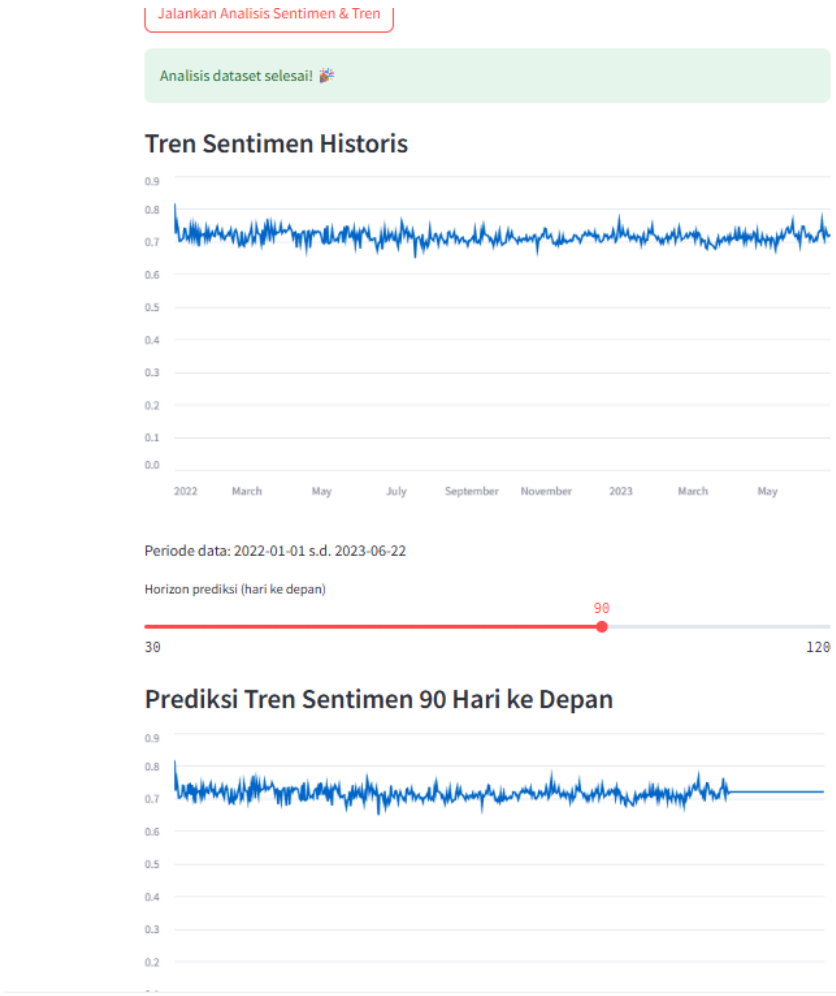
4.3 Rencana Implementasi (Deployment Plan)

Untuk menjembatani kesenjangan antara keluaran model yang bersifat numerik dan kebutuhan pengguna akan informasi intuitif, hasil analisis diintegrasikan ke dalam sistem visualisasi interaktif yang komprehensif.

- Pengecekan Sentimen Real-Time (*Real-Time Inference Interface*) Fitur ini mengintegrasikan *backend* model klasifikasi secara langsung dengan antarmuka *input* pengguna. Saat pengguna memasukkan teks opini, sistem secara instan melakukan pra-pemrosesan, *embedding*, dan prediksi. Hasilnya divisualisasikan dalam bentuk Label Sentimen (Positif/Netral/Negatif) yang diberi kode warna psikologis (Hijau/Abu-abu/Merah) untuk memudahkan interpretasi cepat. Selain itu, ditampilkan pula probabilitas prediksi sebagai indikator tingkat keyakinan model (*confidence score*), memberikan transparansi kepada pengguna mengenai validitas hasil analisis.



- Visualisasi Tren Historis (*Interactive Time-Series Chart*) Hasil analisis sentimen pada data historis diaggregasi berdasarkan waktu dan divisualisasikan menggunakan Grafik Garis Interaktif. Grafik ini memetakan rata-rata skor sentimen harian terhadap sumbu waktu, memungkinkan investor untuk melacak fluktuasi opini pasar dalam periode tertentu. Fitur interaktif seperti *zooming*, *panning*, dan *tooltip* (jendela informasi saat kursor diarahkan) memungkinkan pengguna untuk melakukan investigasi mendalam (*drill-down*) pada tanggal-tanggal spesifik yang memiliki anomali sentimen ekstrem, menghubungkannya dengan peristiwa pasar yang relevan.



- Integrasi Proyeksi Masa Depan (*Forecasting Integration*) Hasil dari model peramalan (*Gradient Boosting*) diintegrasikan secara visual dengan data historis dalam satu kanvas grafik yang kontinu. Garis tren historis (fakta) disambungkan dengan garis tren prediksi (proyeksi), memberikan gambaran holistik mengenai arah pergerakan sentimen. Visualisasi ini dilengkapi dengan *disclaimer* otomatis dan rentang interval prediksi untuk mengedukasi pengguna bahwa hasil peramalan merupakan estimasi probabilistik, bukan kepastian mutlak, sehingga mendukung pengambilan keputusan yang lebih bijak dan terukur.

4.4 Analisis Kesiapan dan Keberlanjutan

Analisis kesiapan sistem menunjukkan bahwa aplikasi saat ini telah mencapai tingkat kesiapan teknologi (*Technology Readiness Level/TRL*) level 7, di mana prototipe sistem telah teruji validitasnya dalam lingkungan operasional berbasis *cloud*. Secara fungsional, aplikasi mampu menjalankan proses *end-to-end* mulai dari input data hingga visualisasi sentimen dengan latensi yang dapat diterima. Namun, keberlanjutan sistem dalam jangka panjang menghadapi tantangan infrastruktur, terutama ketergantungan pada layanan *hosting* gratis yang membatasi kapasitas komputasi (CPU/RAM), sehingga berisiko mengalami penurunan performa jika terjadi lonjakan pengguna secara masif.

Selain aspek teknis infrastruktur, tantangan keberlanjutan juga muncul dari sisi relevansi model terhadap dinamika pasar aset kripto yang sangat fluktuatif. Model rentan mengalami *concept drift*, yaitu penurunan akurasi akibat perubahan pola bahasa dan tren isu di media sosial yang terus berkembang. Oleh karena itu, strategi mitigasi melalui mekanisme *Continuous Learning* menjadi krusial, di mana model perlu dilatih ulang (*retraining*) secara berkala menggunakan data terkini untuk memperbarui pemahamannya terhadap konteks pasar baru. Sebagai langkah pengembangan lanjutan (*future work*), sistem direkomendasikan untuk beralih dari pemrosesan *batch* (file CSV) menuju integrasi API *streaming* guna menyajikan intelijen pasar secara *real-time* dan otomatis sepenuhnya.

4.5 Keterkaitan dengan Tujuan Awal

Pada tahap akhir evaluasi proyek, dilakukan peninjauan kembali terhadap keselarasan antara hasil implementasi sistem dengan tujuan awal penelitian yang telah dirumuskan pada fase *Business Understanding*. Secara keseluruhan, sistem yang dikembangkan telah berhasil menjawab permasalahan fundamental mengenai kebutuhan akan alat analisis sentimen yang objektif dan terukur di tengah volatilitas pasar aset kripto.

Pertama, tujuan untuk menyediakan indikator sentimen publik yang valid telah tercapai melalui model klasifikasi yang memiliki akurasi 70.6%. Hasil ini membuktikan bahwa opini acak di media sosial dapat dikuantifikasi menjadi metrik terstruktur yang merefleksikan psikologi pasar secara *real-time*. Bagi investor, hal ini memberikan nilai tambah berupa kemampuan untuk memvalidasi rumor pasar dengan data konkret, mengurangi risiko pengambilan keputusan yang didasarkan pada *FUD (Fear, Uncertainty, and Doubt)* semata.

Kedua, tujuan untuk mendukung strategi investasi berbasis data terpenuhi melalui fitur *forecasting* dan visualisasi tren. Meskipun akurasi prediksi jangka panjang masih memiliki keterbatasan akibat sifat stokastik pasar, sistem ini tetap memberikan kontribusi signifikan sebagai alat *monitoring* tren jangka pendek. Dengan adanya *dashboard* interaktif, investor dapat mengidentifikasi momentum perubahan sentimen lebih awal, yang merupakan keunggulan kompetitif krusial dalam perdagangan aset berisiko tinggi.

Secara strategis, keberhasilan *deployment* aplikasi ini menegaskan bahwa pendekatan *Data Science* mampu menjembatani kesenjangan informasi antara investor ritel dan institusi besar. Dengan demikian, proyek ini tidak hanya mencapai target teknis pengembangan model, tetapi juga memberikan dampak bisnis nyata berupa demokratisasi akses terhadap intelijen pasar aset kripto**.**

Daftar Pustaka

Kaggle. (2024). Bitcoin Tweets Dataset. Retrieved from https://www.kaggle.com

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Retrieved from https://bitcoin.org/bitcoin.pdf

Statista. (2024). Bitcoin Adoption and Market Sentiment Statistics. Retrieved from https://www.statista.com

SentenceTransformer Documentation from https://sbert.net/

LAMPIRAN

https://drive.google.com/drive/folders/1a55OKGSheM8qTnDWbxs-lvuZjH4ImOAp?usp=sharing