

**Prediksi berbasis Time Series berdasarkan Data Jumlah
Penumpang Bandara APT Pranoto Tahun 2018 - 2023**



Dosen Pengampu:

Dr. Wiwik Anggraeni, S.Si., M.Kom.

Permodelan & Analitika Prediktif (C)

Kelompok 6:

Aulisa Rizki Amanda	(5026221024)
Shof Watun Niswah	(5026221043)
Satria Dwi Nugraha	(5026221119)
Muhammad Daffa Alvinoer Rahman	(5026221180)

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER SURABAYA
2024**

DAFTAR ISI

DAFTAR ISI	1
DAFTAR TABEL	2
DAFTAR GAMBAR	3
BAB I	4
PENDAHULUAN	4
1.1 Latar Belakang	4
1.2 Rumusan Masalah	4
1.3 Tujuan	4
BAB II	5
METODOLOGI	5
2.1 Model SARIMA	5
2.2 Model Prophet	5
2.3 Model GRU	5
2.4 Pra-Pemrosesan Data	6
2.5 Pelatihan Model dan Evaluasi Kinerja	6
Proses pelatihan model dilakukan dengan menggunakan tiga pendekatan, yaitu SARIMA, Prophet, dan Hybrid Prophet-GRU. Tahap pelatihan ini mencakup pembagian data, proses pelatihan, dan evaluasi kinerja model untuk memastikan akurasi dan keandalan prediksi.	6
BAB III	8
IMPLEMENTASI METODE	8
3.2 Implementasi Model SARIMA	9
3.3 Implementasi Model Prophet	11
3.4 Implementasi Model Hybrid (Prophet - GRU)	11
BAB IV	12
HASIL DAN PEMBAHASAN	12
4.1 Hasil Forecast	12
4.2 Metrik Evaluasi	18
4.3 Interpretasi Hasil	19
KESIMPULAN	20
DAFTAR PUSTAKA	21
LAMPIRAN	22

DAFTAR TABEL

Tabel 3.1 Uji Augmented Dicky-Fuller

9

DAFTAR GAMBAR

Gambar 1. Pembuatan Dataframe Baru	8
Gambar 2. Grafik ACF & PACF	9
Gambar 3.	16
Gambar 4.	16
Gambar 5.	17
Gambar 6.	17

BAB I

PENDAHULUAN

1.1 Latar Belakang

Analisis *time series* merupakan metode yang umum digunakan untuk memprediksi data masa depan berdasarkan pola data sebelumnya. Metode ini dapat diterapkan dalam berbagai bidang, seperti ekonomi, transportasi, dan kesehatan, karena dapat memberikan informasi yang cukup akurat untuk membantu pengambilan keputusan secara strategis. Salah satu contoh penerapannya adalah membuat prediksi jumlah penumpang dalam suatu bandara pada periode tertentu. Dengan tren pertumbuhan jumlah penumpang yang terus meningkat, bandara membutuhkan sistem prediksi yang efektif untuk mengelola sumber daya dan perencanaan operasional.

Kami menggunakan dataset jumlah penumpang Bandara APT Pranoto selama periode 2018-2023 untuk dilakukan analisis dan pembuatan prediksi periode selanjutnya. Pemilihan data ini dikarenakan kami merasa data pada tahun 2018-2023 cukup untuk membuat prediksi pada periode selanjutnya. Selain itu, Bandara ini merupakan satu-satunya bandara di Kota Samarinda, sehingga perlu diperhatikan juga performa pelayanannya. Pendekatan yang akan dilakukan menggunakan model *Seasonal AutoRegressive Integrated Moving Average* (SARIMA), *Prophet*, dan *Gated Recurrent Unit* (GRU). Selain itu, dilakukan juga pendekatan dengan metode gabungan antara *Prophet* dan GRU untuk meningkatkan akurasi prediksi. Pemilihan model ini telah disesuaikan berdasarkan tipe data yang ada, dimana data jumlah penumpang ini bertipe *seasonal* dikarenakan seperti yang kita ketahui bahwa bandara cenderung mengalami lonjakan penumpang ketika musim liburan tiba dan akan kembali normal pada waktu-waktu tertentu.

1.2 Rumusan Masalah

1. Bagaimana memanfaatkan model prediksi deret waktu untuk memperkirakan jumlah penumpang di masa depan?
2. Model apa yang memberikan hasil prediksi paling akurat dalam konteks data penumpang Bandara APT Pramono?

1.3 Tujuan

1. Menerapkan model prediksi time series (SARIMA, Prophet, dan Hybrid Prophet-GRU) untuk data jumlah penumpang Bandara APT Pramono
2. Mengevaluasi kinerja model berdasarkan metrik evaluasi seperti MSE, MAE, MAPE, dan R^2
3. Memberikan hasil prediksi untuk perencanaan jumlah penumpang selama satu tahun ke depan

BAB II METODOLOGI

2.1 Model SARIMA

Model SARIMA merupakan pengembangan dari model ARIMA yang dirancang untuk menangani data deret waktu yang memiliki pola musiman (Suwandi, G. 2021). Model ini terdiri dari dua komponen, yaitu bagian tidak musiman dan bagian musiman. Komponen tidak musiman dalam metode ini menggunakan model ARIMA. Model SARIMA dapat ditulis dengan notasi berikut (Makridakis et al., 1999):

$$ARIMA(p,d,q)(P,D,Q)S$$

Dengan:

(p, d, q): bagian tidak musiman dari model

(P, D, Q): bagian musiman dari model

P: orde musiman AR

D: tingkat differencing musiman

Q: orde musiman MA

S: jumlah periode dalam satu siklus musiman

2.2 Model Prophet

Prophet adalah model peramalan data deret waktu yang dibuat oleh Facebook pada tahun 2017. Prophet memiliki kemampuan untuk memodelkan data deret waktu dengan pola dan tren musiman yang kuat, dan dia mampu mengoptimalkan model adaptasi dalam waktu yang singkat (Giovany Syuhada & Helmi Setyawan, 2023). Prophet hanya menggunakan 2 komponen data utama, yaitu komponen waktu dan komponen jumlah data yang dilambangkan dengan 'ds' dan 'y' (Auliya et al., 2023). Prophet didefinisikan sebagai berikut:

$$y(t) = g(t) + s(t) + \varepsilon \quad (7)$$

dengan:

$g(t)$: model tren yang menggambarkan kenaikan atau penurunan jangka panjang dalam data.

$s(t)$: model deret Fourier yang menggambarkan bagaimana data dipengaruhi oleh faktor musiman seperti waktu dalam setahun.

ε : model yang mewakili istilah kesalahan yang tidak dapat dikurangi.

2.3 Model GRU

GRU (Gated Recurrent Unit) adalah varian dari LSTM (Long Short-Term Memory) yang lebih sederhana. Struktur berulang ini dirancang untuk mencapai keseimbangan yang baik antara kinerja dan efisiensi, sehingga GRU banyak digunakan baik di bidang akademik maupun industri. GRU pertama kali diperkenalkan oleh Chung et al. pada tahun

2014 dengan tujuan utama membuat unit berulang yang mampu secara adaptif menangkap ketergantungan data pada berbagai skala waktu.

Sebagai ilustrasi, manusia tidak perlu selalu menggunakan semua informasi dari masa lalu untuk membuat keputusan saat ini. Misalnya, ketika kita memutuskan untuk membeli makanan sekarang, informasi tentang jadwal ujian tengah semester di masa lalu kemungkinan besar tidak terlalu relevan untuk keputusan tersebut. Dalam GRU, mekanisme untuk mengatur aliran informasi ini disebut sebagai *gate*. GRU memiliki dua jenis *gate*, yaitu *reset gate* dan *update gate*. Dalam konteks analogi tadi, *reset gate* bertugas mengatur bagaimana input baru digabungkan dengan informasi dari masa lalu, sedangkan *update gate* menentukan seberapa banyak informasi dari masa lalu yang perlu dipertahankan.

2.4 Pra-Pemrosesan Data

Salah satu tahap penting dalam proses ini adalah pra-pemrosesan, yaitu serangkaian langkah yang dirancang untuk membersihkan, mengubah, dan mempersiapkan data sebelum digunakan oleh model. Tahap ini memiliki peran yang sangat penting, karena data yang tidak terstruktur, mengandung anomali, atau tidak lengkap dapat memengaruhi kinerja model dan menghasilkan prediksi yang kurang andal. Oleh sebab itu, langkah-langkah pra-pemrosesan menjadi fondasi yang kuat untuk menciptakan model machine learning yang akurat, efisien, dan mudah diinterpretasi. Hasil dari tahap pra-pemrosesan data dapat mencakup perubahan kualitas atau karakteristik data yang memengaruhi langkah-langkah berikutnya dalam proses pembelajaran mesin. Salah satu hasil dari pra-pemrosesan adalah dataset yang lebih bersih dan lengkap setelah penanganan nilai yang hilang, sehingga menghilangkan celah informasi yang dapat memengaruhi analisis. Selain itu, pasar kerja saat ini sangat membutuhkan keterampilan dalam bidang data science dan analitik. Oleh karena itu, mengajarkan langkah-langkah pra-pemrosesan membantu siswa memahami cara membersihkan data mentah sehingga siap digunakan untuk keperluan model.

2.5 Pelatihan Model dan Evaluasi Kinerja

Proses pelatihan model dilakukan dengan menggunakan tiga pendekatan, yaitu SARIMA, Prophet, dan Hybrid Prophet-GRU. Tahap pelatihan ini mencakup pembagian data, proses pelatihan, dan evaluasi kinerja model untuk memastikan akurasi dan keandalan prediksi.

2.5.1 Pembagian Data

Data historis dibagi ke dalam dua set:

- Set pelatihan digunakan untuk melatih model.
- Set pengujian digunakan untuk mengevaluasi kemampuan model dalam membuat prediksi.

Tiga skema pembagian data yang digunakan adalah:

- 80:20: 80% data untuk pelatihan dan 20% untuk pengujian.
- 70:30: 70% data untuk pelatihan dan 30% untuk pengujian.
- 60:40: 60% data untuk pelatihan dan 40% untuk pengujian.

Pembagian ini bertujuan untuk mengevaluasi kinerja model pada berbagai proporsi

data pelatihan dan pengujian.

2.5.2 Proses Pelatihan Model

1. SARIMA: model SARIMA dilatih dengan menyesuaikan parameter musiman dan non-musiman berdasarkan analisis Autocorrelation Function (ACF) dan Partial Autocorrelation Function (PACF). Parameter optimal dipilih untuk meminimalkan error prediksi.
2. Prophet: Model Prophet dilatih dengan data terformat khusus menggunakan kolom waktu sebagai ds dan target sebagai y. Model ini memanfaatkan komponen tren dan musiman untuk menangkap pola data.
3. Hybrid Prophet-GRU: Pendekatan hybrid ini menggabungkan kemampuan Prophet dalam menangkap pola musiman dengan GRU yang efisien dalam mempelajari pola non-linearitas. Data diproses menggunakan MinMaxScaler, kemudian dilatih menggunakan pendekatan sliding window untuk menghasilkan prediksi yang lebih baik.

2.5.3 Evaluasi Kinerja Model

Kinerja model dievaluasi menggunakan empat metrik berikut:

- Mean Absolute Error (MAE): Mengukur rata-rata kesalahan absolut dalam satuan asli data.
- Mean Absolute Percentage Error (MAPE): Mengukur kesalahan prediksi dalam persentase relatif terhadap nilai aktual.
- Root Mean Squared Error (RMSE): Mengukur akar rata-rata kesalahan kuadrat untuk memberikan bobot lebih besar pada kesalahan besar.
- R-squared (R^2): Mengukur proporsi variabilitas data yang dapat dijelaskan oleh model.

BAB III IMPLEMENTASI METODE

3.1 Tahap Pra-Pemrosesan Data

Pra-pemrosesan data bertujuan untuk menyusun data agar lebih terstruktur dan siap dianalisis. Dalam analisis berbasis waktu, data waktu sering diformat ulang, seperti menggabungkan tahun dan bulan menjadi tanggal lengkap, serta menyaring kolom yang relevan.

```
# Membuat kolom date dengan hanya tahun dan bulan
df['date'] = pd.to_datetime(df['Tahun'].astype(str) + '-' + df['Bulan'].astype(str))

# Assuming the correct column name is 'Jumlah'

# Membuat DataFrame baru dengan kolom date dan data jumlah penumpang
new_df = df[['date', 'Jumlah']].copy() # Changed the column name here

# Menampilkan info
new_df.info()
print(new_df.dtypes)
```

Gambar 1. Pembuatan DataFrame Baru

Kode di atas adalah bagian dari tahap pra-pemrosesan data yang bertujuan untuk mempersiapkan data agar lebih terstruktur dan siap digunakan dalam analisis atau pemodelan.

1. Menggabungkan Kolom Tahun dan Bulan menjadi Format Tanggal:

- Kode `pd.to_datetime(df['Tahun'].astype(str) + '-' + df['Bulan'].astype(str))` menggabungkan kolom *Tahun* dan *Bulan* untuk membuat kolom baru bernama `date` dalam format *datetime*. Ini membantu menyajikan informasi waktu secara terstruktur dan mempermudah analisis berbasis waktu, seperti tren bulanan atau musiman.

2. Membuat DataFrame Baru yang Lebih Ringkas:

- Kode `new_df = df[['date', 'Jumlah']].copy()` membuat DataFrame baru hanya dengan kolom `date` (tanggal) dan `Jumlah` (yang diasumsikan mewakili jumlah penumpang). Ini bertujuan untuk fokus hanya pada data yang relevan, menyederhanakan dataset, dan mengurangi kebisingan dari kolom yang tidak diperlukan.

3. Memastikan Informasi dan Tipe Data:

- Kode `new_df.info()` dan `print(new_df.dtypes)` digunakan untuk menampilkan informasi struktur DataFrame, seperti jumlah baris, kolom, tipe data, dan apakah terdapat nilai yang hilang (*missing values*). Hal ini penting untuk memastikan data telah terformat dengan benar sebelum melanjutkan ke analisis lebih lanjut.

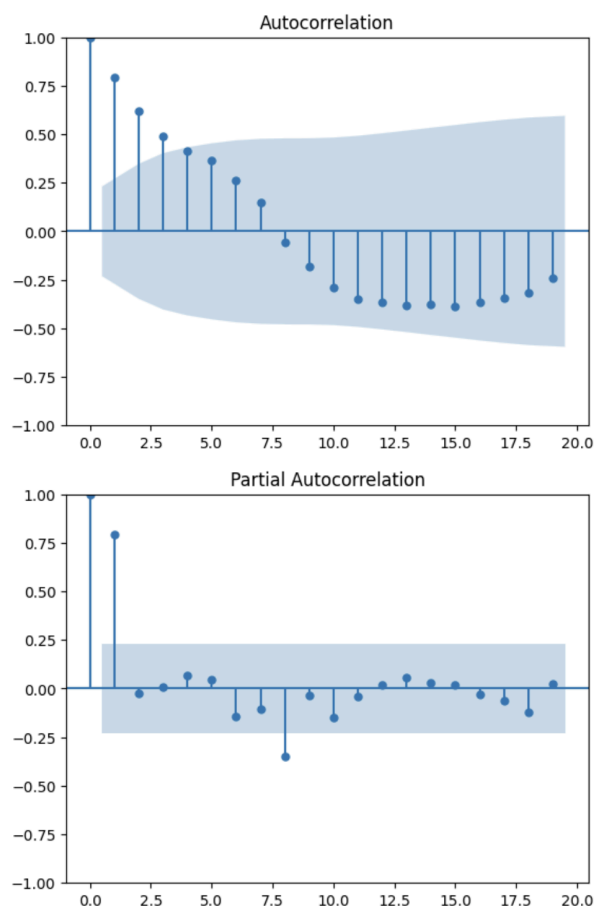
Secara keseluruhan, kode ini adalah langkah awal untuk mengubah data mentah menjadi lebih terorganisir dan siap untuk digunakan dalam analisis berbasis waktu, seperti peramalan atau pengenalan pola.

3.2 Implementasi Model SARIMA

Dalam melakukan implementasi model SARIMA dalam *forecasting*, dataset yang digunakan harus stasioner dengan nilai p-value di bawah nilai alpha 0,05. Untuk mengetahui stasioneritas dataset, dilakukan *Augmented Dicky-Fuller Test* dan analisis ACF dan PACF. Hasil dari ADF dan ACF-PACF dapat dilihat pada tabel dan gambar di bawah ini.

Tabel 3.1 Uji *Augmented Dicky-Fuller*

p-value	symbol	alpha
0.1100349461893812	>	0.05



Gambar 2. Grafik ACF & PACF

Berdasarkan hasil uji Augmented Dickey-Fuller (ADF), diperoleh p-value sebesar 0.0481 yang lebih kecil dari alpha (0.05), sehingga hipotesis awal (H_0) yang menyatakan bahwa data tidak stasioner ditolak. Dengan demikian, data dapat dianggap stasioner tanpa

perlu differencing tambahan. Berdasarkan grafik ACF dan PACF, berikut parameter awal yang dapat dipertimbangkan:

- **Order AR (p):** Nilai signifikan pada PACF berhenti di lag 1 atau 2, sehingga $p=1$ atau $p=2$.
- **Order MA (q):** Berdasarkan ACF, autokorelasi signifikan hingga beberapa lag, sehingga q juga dapat dimulai dari 1 atau 2.

Parameter Musiman (P,D,Q,s):

- $P=1$: Mengindikasikan satu lag musiman dalam komponen autoregresif musiman. Nilai ini ditentukan dari analisis PACF yang menunjukkan pola musiman signifikan.
- $D=0$: Menunjukkan bahwa sudah stasioner dan tidak diperlukan diferensiasi
- $Q=1$: Menunjukkan satu lag musiman dalam komponen moving average musiman. Nilai ini diambil dari analisis ACF musiman.
- $s=12$: Siklus musiman diperkirakan terjadi setiap 12 periode (misalnya, data bulanan dengan pola tahunan).

Asumsi

Model SARIMA yang digunakan dalam penelitian ini memiliki parameter $(p,d,q)=(4,1,1)$ untuk komponen non-musiman dan (P,D,Q,s) untuk komponen musiman. Pemilihan parameter ini didasarkan pada analisis ACF dan PACF serta evaluasi performa model. Berikut adalah asumsi yang mendasari penggunaan parameter tersebut:

1. Komponen Autoregresif (AR):

Nilai $p=4$ dipilih karena diperlukan untuk menangkap hubungan jangka pendek yang signifikan hingga lag ke-4. Meskipun PACF menunjukkan penurunan eksponensial, uji coba menunjukkan bahwa komponen AR hingga lag 4 memberikan hasil prediksi yang lebih optimal dibandingkan nilai p yang lebih rendah.

2. Komponen Diferensiasi (Differencing):

Data menunjukkan adanya pola musiman dan tren non-musiman terutama pengaruh adanya pandemi COVID-19 membuat pola tren musiman dari data berubah. Oleh karena itu, model SARIMA dengan parameter musiman dan non-musiman digunakan untuk menangkap karakteristik ini. Nilai $d=1$ digunakan untuk membuat data menjadi stasioner. Hal ini sesuai dengan analisis awal yang menunjukkan adanya tren pada data sebelum proses diferensiasi.

3. Komponen Moving Average (MA):

Nilai $q=1$ dipilih berdasarkan analisis ACF, di mana terdapat pola cut-off pada lag pertama, yang menunjukkan pentingnya koreksi kesalahan prediksi menggunakan lag sebelumnya.

Model SARIMA yang digunakan memiliki notasi lengkap sebagai berikut:

$$SARIMA(4,1,1)(1,0,1,12)$$

3.3 Implementasi Model Prophet

Proses implementasi model prophet pada dataset ini dilakukan dalam tiga kasus pembagian data pelatihan dan pengujian, yaitu 80-20, 70-30, dan 60-40. Proses ini dimulai dengan membagi data menjadi 2 bagian yaitu, data pelatihan dan data untuk pengujian sesuai dengan kasus yang sedang dilakukan. Dilanjutkan dengan melakukan uji stasioner untuk data pelatihan pada kolom jumlah menggunakan uji *Augmented Dickey-Fuller*. Sebelum memulai pelatihan model prophet, dilakukan penyetelan ulang terhadap index pada data training sehingga date menjadi kolom biasa dan mengubah nama kolom sesuai dengan format yang dibutuhkan model prophet, yaitu date menjadi 'ds' dan jumlah menjadi 'y'. Setelah data sudah sesuai, dilakukan pelatihan model prophet dan dilanjutkan dengan penyesuaian data test agar sesuai dengan format model prophet. Model prophet yang sudah dilatih digunakan untuk melakukan prediksi menggunakan data test dan ditampilkan dalam bentuk plot. Selain hasil prediksi, komponen-komponen dari model prophet, seperti tren dan musiman juga ditampilkan dalam bentuk grafik. Setelah itu, dilakukan pembuatan plot yang membandingkan antara prediksi dan data aktual serta pengecekan nilai MAE, MAPE, RMSE, dan R-squared dengan tujuan mengukur seberapa baik model yang sudah dibuat. Terakhir membuat visualisasi residuals untuk memeriksa residuals antara nilai prediksi dan nilai aktual.

3.4 Implementasi Model Hybrid (Prophet - GRU)

Model Hybrid menggabungkan Prophet yang digunakan untuk menangkap pola musiman dan tren dengan GRU yang efektif untuk menangkap pola non-linearitas dalam data. Hasil dari Implementasi ini adalah kombinasi rata-rata dari output Prophet dan GRU.

Pertama, Prophet dilatih menggunakan data historis, dengan data diubah ke format yang sesuai, yaitu kolom waktu sebagai ds dan target sebagai y. Prophet kemudian menghasilkan prediksi untuk data validasi atau data masa depan. Kedua, data yang sama dinormalisasi menggunakan MinMaxScaler agar sesuai dengan format masukan GRU. Model GRU dilatih dengan pendekatan sliding window, menggunakan 12 bulan terakhir (look-back) sebagai input untuk memprediksi nilai bulan berikutnya. Setelah model Prophet dan GRU memberikan prediksi, kedua hasil digabungkan dengan cara menghitung rata-rata (averaging) untuk menghasilkan prediksi hybrid yang memadukan keunggulan keduanya. Pendekatan hybrid ini dievaluasi menggunakan data yang dibagi menjadi beberapa split (80%, 70%, 60%) untuk pelatihan dan pengujian.

Evaluasi metrik dilakukan untuk mengukur performa model Hybrid dengan pengukuran MAE, RMSE, MAPE, dan R2. Model Hybrid kemudian digunakan untuk melakukan prediksi selama 1 tahun ke depan.

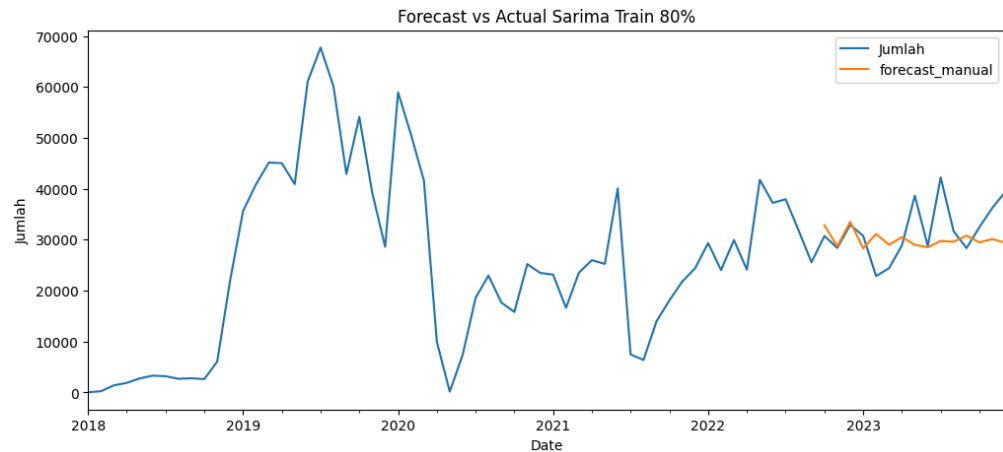
BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Forecast

4.1.1 Model SARIMA

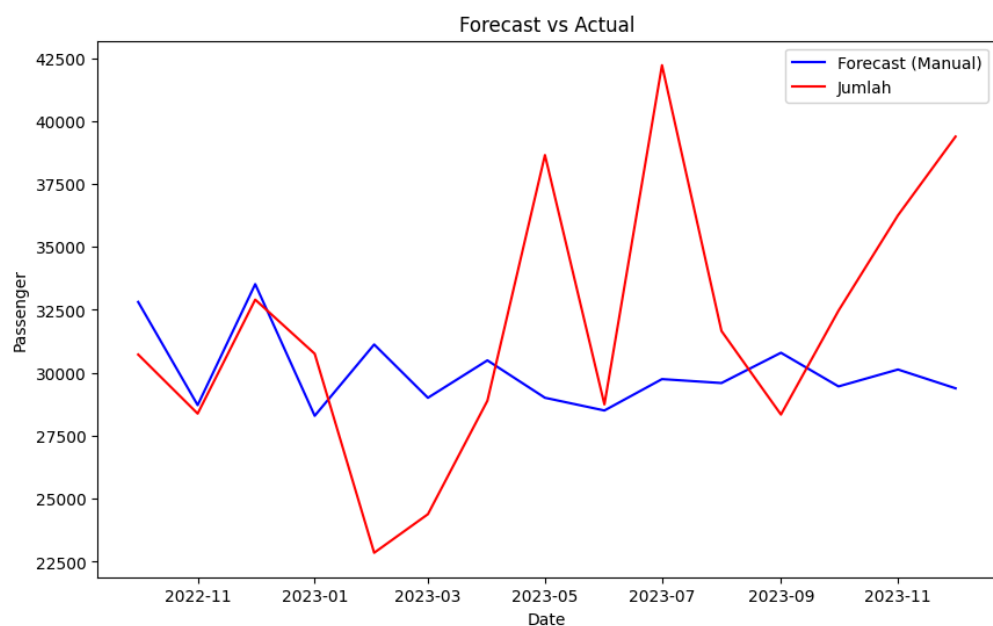
a. Train Split 80%



Gambar 3. SARIMA Forecast (Train Data 80%)

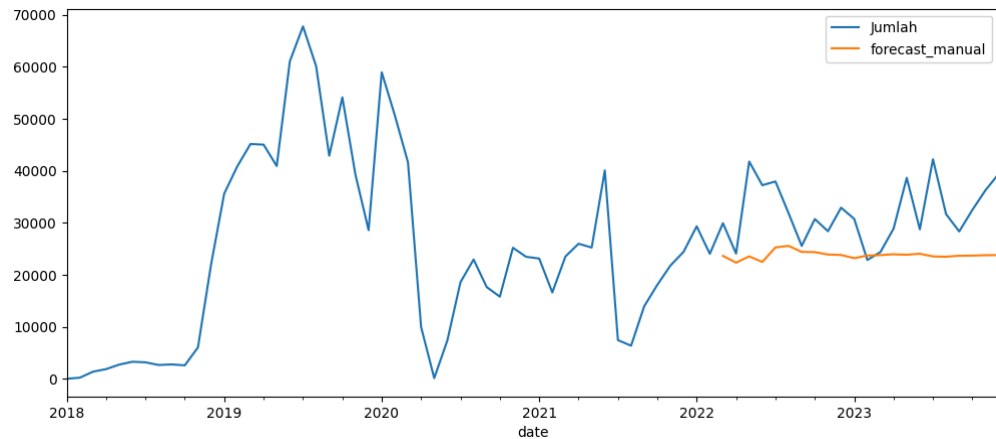
Grafik di atas menunjukkan hasil prediksi menggunakan model **SARIMA** dengan pembagian data **80% untuk pelatihan** dan **20% untuk pengujian**. Garis biru merepresentasikan data aktual jumlah penumpang, sedangkan garis oranye menunjukkan hasil prediksi model SARIMA untuk data pengujian.

Pada data pengujian prediksi model terlihat cenderung lebih stabil dan kurang mampu mengikuti fluktuasi yang tajam pada data aktual. Hal ini disebabkan oleh sifat SARIMA yang lebih fokus pada pola musiman dan tren umum daripada menangkap fluktuasi yang tidak terduga.



Gambar 4. SARIMA Forecast (Train Data 80%)

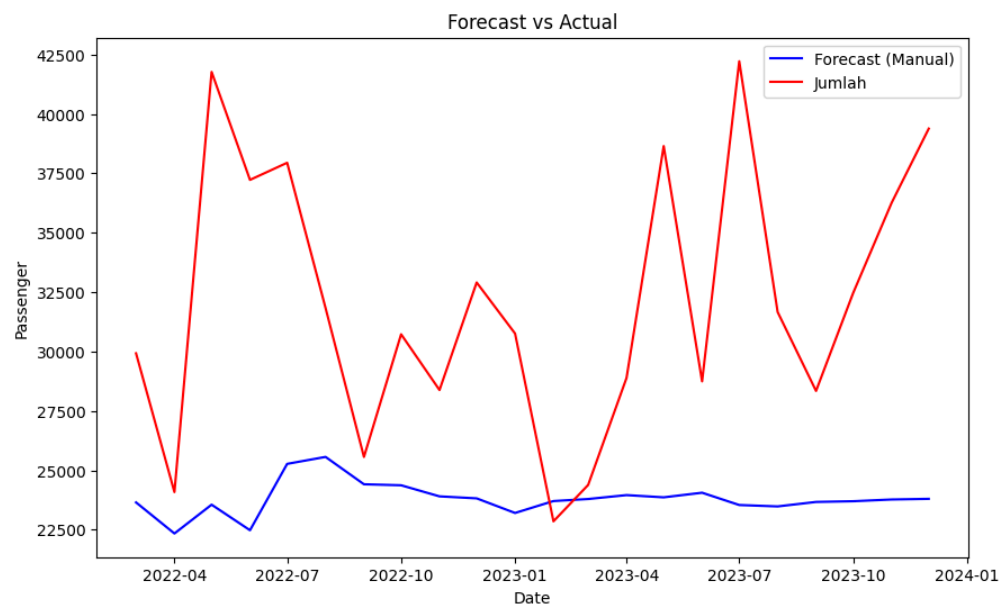
b. Train Split 70%



Gambar 5.

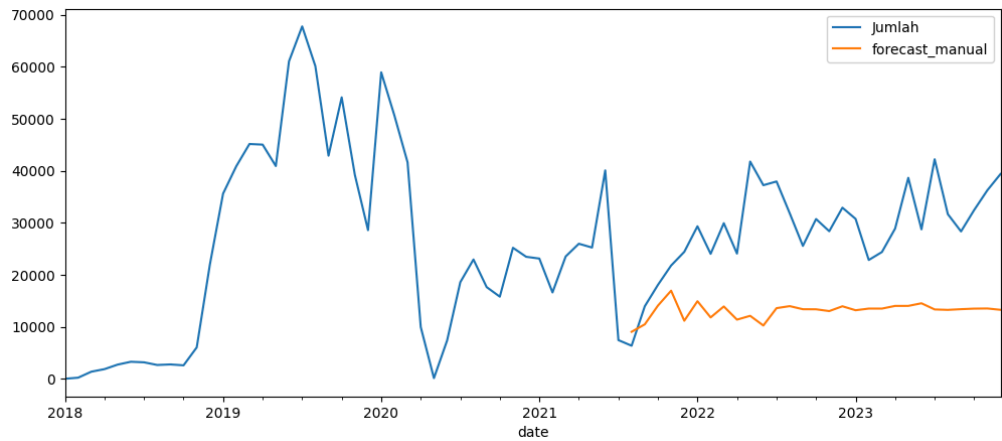
Grafik di atas menunjukkan hasil prediksi **SARIMA** dengan pembagian data **70% untuk pelatihan** dan **30% untuk pengujian**. **Garis biru** merepresentasikan data aktual jumlah penumpang, sedangkan **garis oranye** menunjukkan hasil prediksi model SARIMA pada data pengujian (test).

Pada data pengujian, model SARIMA menghasilkan prediksi yang cenderung stabil dan kurang mampu menangkap fluktuasi tajam pada data aktual. Garis oranye menunjukkan pola prediksi yang relatif datar tanpa mencerminkan variasi musiman yang terlihat jelas pada data aktual, seperti lonjakan atau penurunan yang signifikan.



Gambar 6.

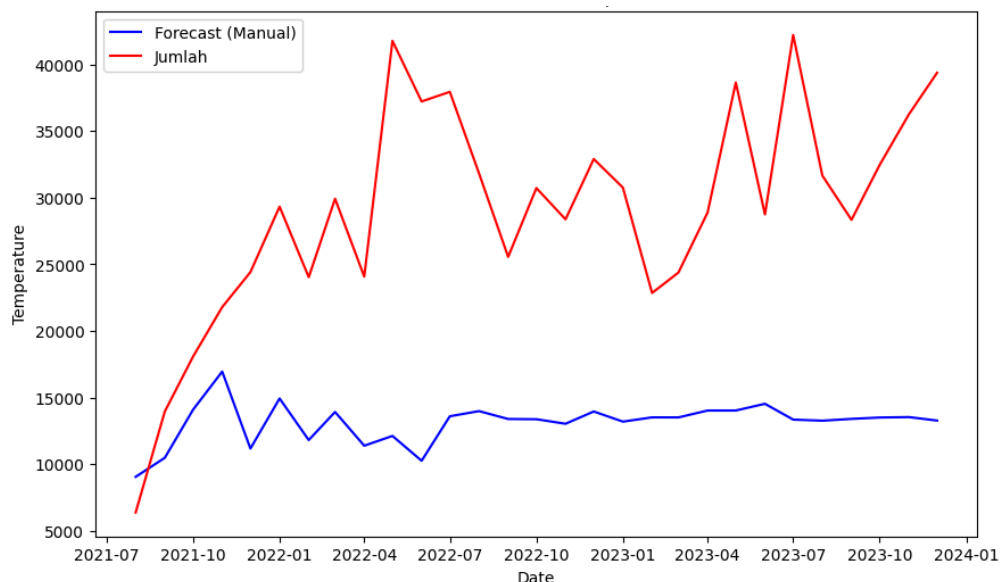
c. Train Split 60%



Gambar 7.

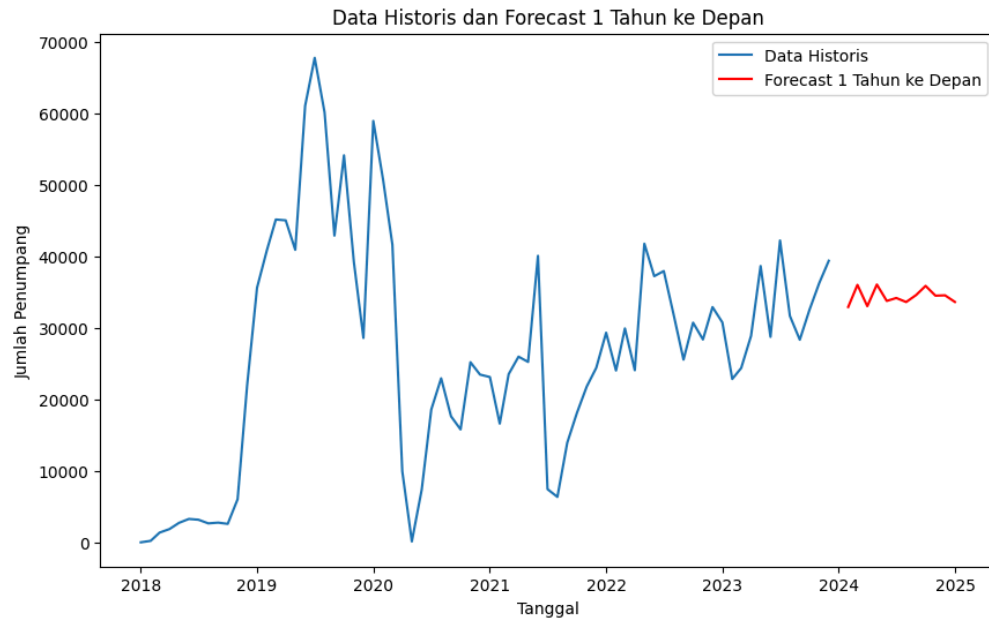
Grafik di atas menampilkan hasil prediksi model **SARIMA** dengan **60% data digunakan untuk pelatihan** dan **40% untuk pengujian**. Garis biru menunjukkan data aktual jumlah penumpang, sedangkan garis oranye menunjukkan hasil prediksi dari model SARIMA pada data pengujian.

Hasil prediksi model SARIMA pada data pengujian menunjukkan pola yang relatif datar dan stabil. Hal ini mengindikasikan bahwa model SARIMA kesulitan dalam menangkap fluktuasi musiman yang signifikan, terutama pada periode dengan kenaikan dan penurunan tajam pada data aktual. Prediksi cenderung berkonsentrasi di sekitar rata-rata jumlah penumpang, sehingga tidak mampu mereplikasi variasi musiman dan dinamika fluktuasi yang terlihat pada data aktual.



Gambar 8.

d. Forecast 1 Tahun ke depan



Gambar 9.

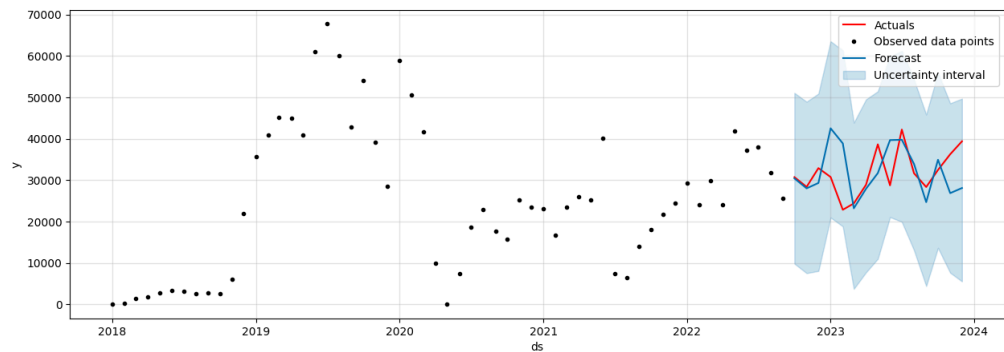
Grafik di atas menampilkan hasil prediksi 1 tahun ke depan (2024) menggunakan model SARIMA, dengan seluruh data historis (2018–2023) digunakan sebagai data pelatihan. Garis biru merepresentasikan data historis jumlah penumpang, sementara garis merah menunjukkan prediksi model SARIMA untuk periode 2024.

Hasil prediksi menunjukkan bahwa model SARIMA memproyeksikan jumlah penumpang dengan pola yang relatif stabil selama tahun 2024. Tidak ada fluktuasi tajam dalam prediksi, yang mengindikasikan bahwa SARIMA menghasilkan proyeksi berdasarkan pola musiman rata-rata dari data historis. Model menangkap pola musiman yang berulang dari data historis, namun fluktuasi atau kejadian ekstrem seperti lonjakan atau penurunan tajam yang terjadi pada data historis tidak tercermin dalam hasil prediksi.

4.1.2 Model Prophet

a. Train Split 80%

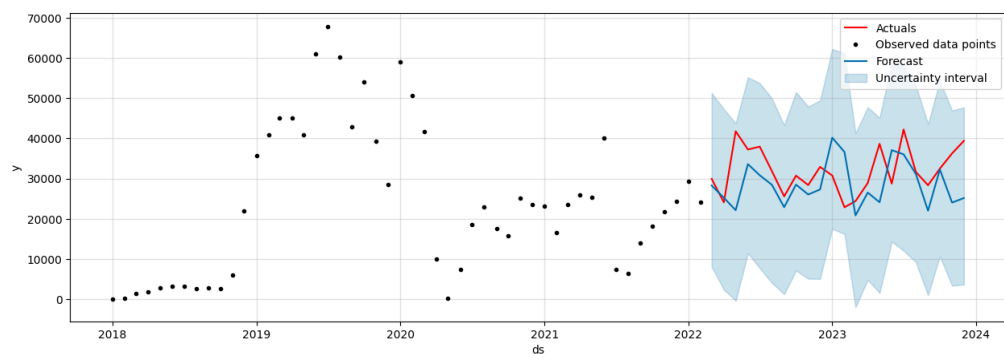
Data dibagi menjadi 80% untuk pelatihan dan 20% untuk pengujian. Garis merah menunjukkan hasil prediksi model Prophet, sementara titik-titik hitam merepresentasikan data aktual. Model Prophet mampu menangkap pola musiman dan tren dengan baik pada data pelatihan. Pada data pengujian, prediksi cukup akurat, dengan interval ketidakpastian (area biru) yang relatif sempit, menunjukkan kepercayaan model terhadap prediksi. Meski demikian, terdapat sedikit penyimpangan pada beberapa fluktuasi ekstrem.



Gambar 10. SARIMA Forecast (Train Data 80%)

b. Train Split 70%

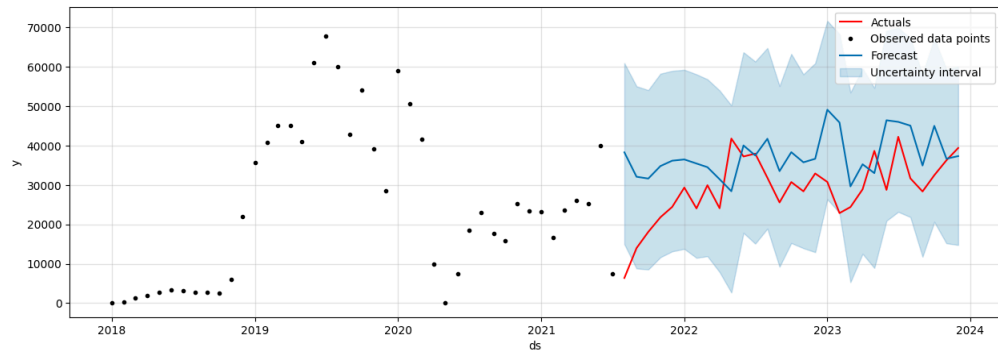
Data pelatihan hanya mencakup 70% dari total data, dengan 30% untuk pengujian. Model Prophet tetap menangkap pola musiman dan tren secara keseluruhan, tetapi prediksi pada data pengujian menunjukkan tingkat ketidakakuratan yang lebih tinggi dibandingkan dengan split 80%. Hal ini terlihat dari interval ketidakpastian (area biru) yang lebih lebar, mencerminkan peningkatan ketidakpastian akibat jumlah data pelatihan yang lebih sedikit. Penyimpangan terlihat lebih sering terjadi pada periode dengan fluktuasi tajam.



Gambar 11. SARIMA Forecast (Train Data 70%)

c. Train Split 60%

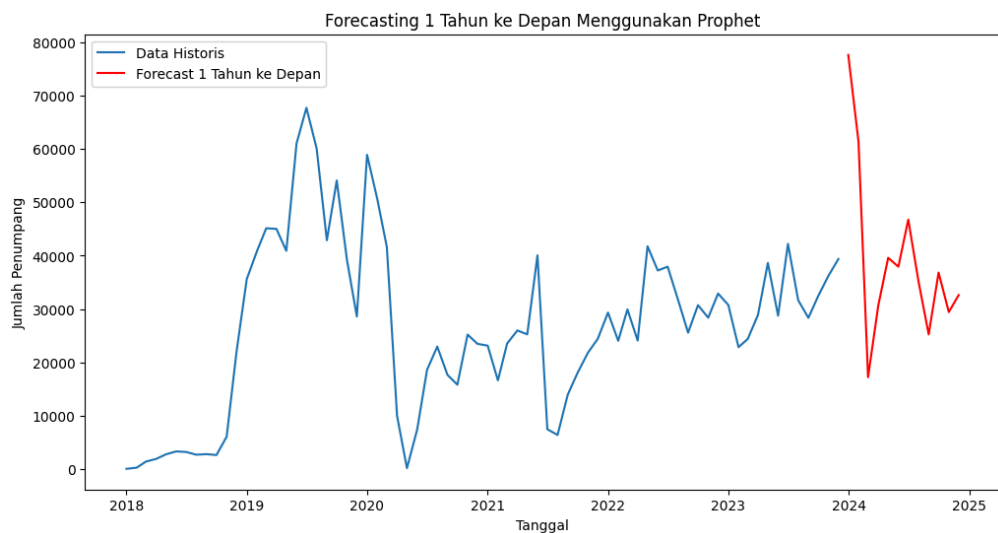
Data pelatihan hanya mencakup 60% dari total data, dengan 40% untuk pengujian. Dengan semakin sedikitnya data pelatihan, model mengalami kesulitan yang lebih besar dalam menangkap pola musiman dan tren pada data pengujian. Interval ketidakpastian menjadi semakin lebar, yang menunjukkan bahwa model kurang percaya diri terhadap prediksi pada data pengujian. Beberapa penyimpangan besar antara prediksi (garis merah) dan data aktual (titik hitam) juga terlihat jelas, terutama pada periode dengan pola fluktuasi yang signifikan.



Gambar 12. SARIMA Forecast (Train Data 60%)

d. Forecast 1 tahun ke depan

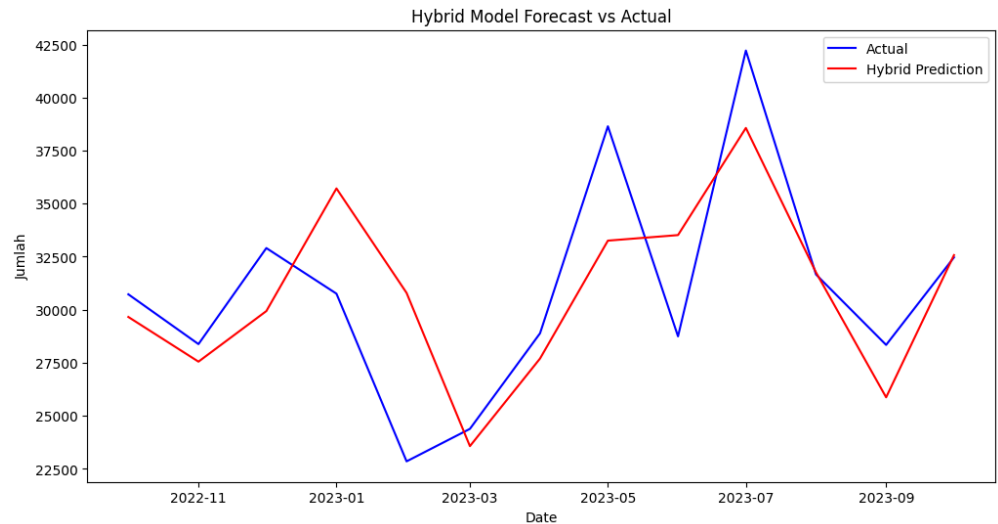
Pada hasil prediksi untuk tahun 2024, model memproyeksikan adanya fluktuasi musiman yang mirip dengan pola yang muncul pada data historis. Misalnya, model memperkirakan adanya kenaikan jumlah penumpang pada pertengahan tahun, yang sesuai dengan pola musiman sebelumnya. Namun, proyeksi model menunjukkan adanya puncak yang tajam pada awal tahun 2024, diikuti dengan fluktuasi yang lebih stabil pada sisa tahun tersebut.



Gambar 13. SARIMA Forecast Next-Year

4.1.3 Model Hybrid Prophet GRU

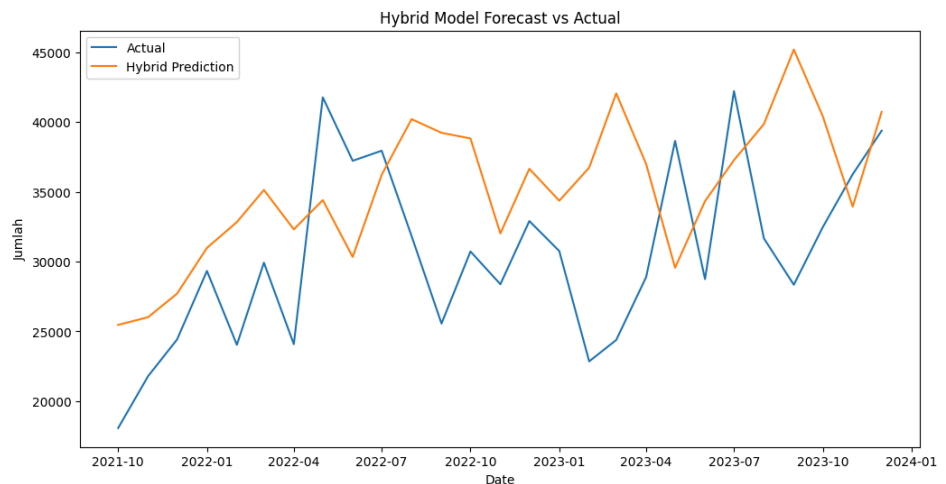
a. Train 80%



Gambar 14. Hybrid Forecast (Train Data 80%)

Gambar di atas merupakan grafik hasil prediksi menggunakan model Hybrid Prophet - GRU dengan data test 20%. Garis biru menggambarkan data aktual sementara garis merah menggambarkan hasil forecast dengan hybrid prediction yang dilakukan pada data test 20%.

b. Train 70%

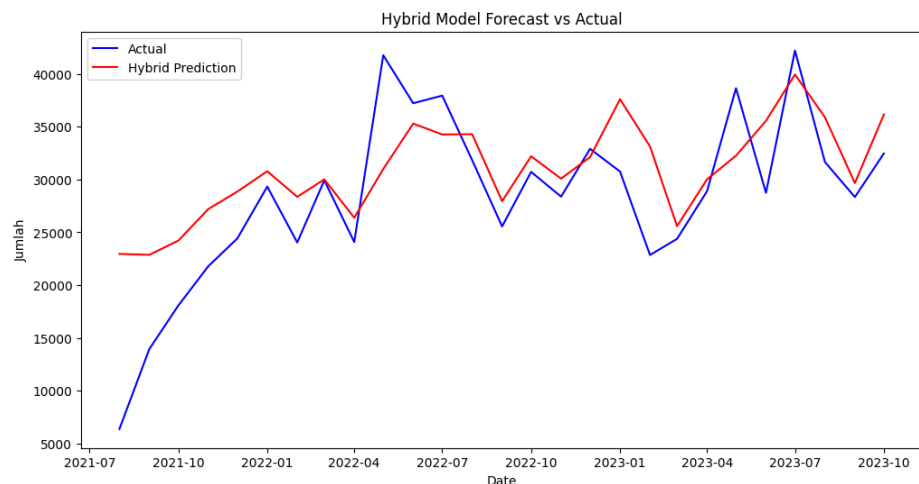


Gambar 15. Hybrid Forecast (Train Data 70%)

Gambar di atas merupakan grafik hasil prediksi menggunakan model Hybrid Prophet - GRU dengan data test 20%. Garis biru menggambarkan data aktual sementara garis merah menggambarkan hasil forecast dengan hybrid prediction yang dilakukan pada data test 20%.

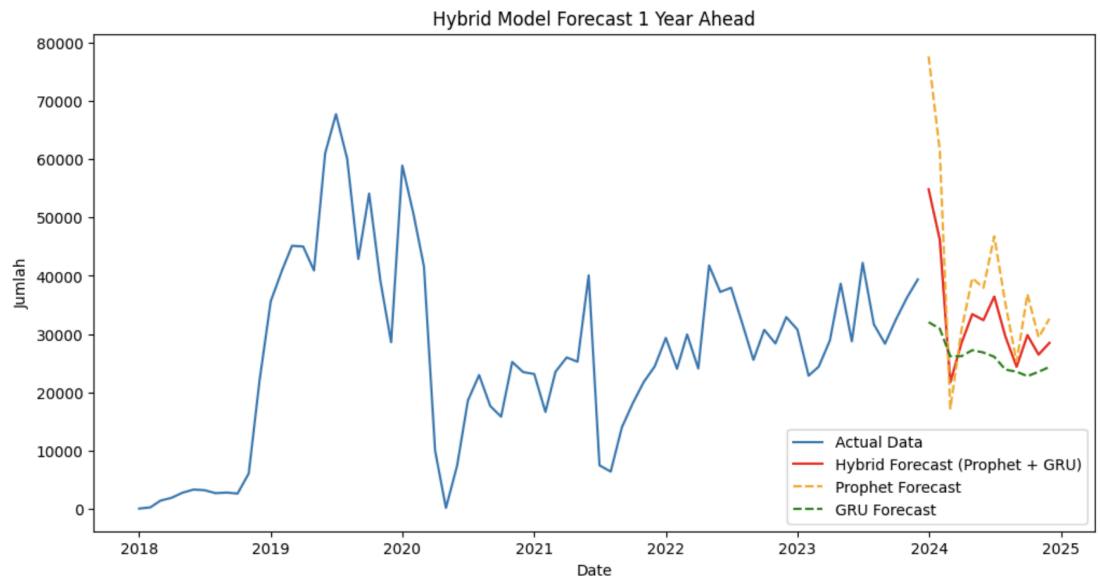
Pada bagian ini, grafik menunjukkan hasil prediksi (Hybrid Prediction) dibandingkan dengan data aktual (Actual). Garis biru merepresentasikan data aktual, sedangkan garis oranye merepresentasikan hasil prediksi model. Hasil prediksi menunjukkan adanya fluktuasi yang cukup besar antara prediksi dan data aktual. Beberapa titik prediksi tampak cukup akurat mengikuti pola data aktual, namun di beberapa bagian terdapat deviasi yang signifikan. Hal ini mengindikasikan bahwa dengan 70% data training, model dapat menangkap pola umum dari data, tetapi mungkin masih kurang baik dalam mengadaptasi fluktuasi data aktual

c. Train 60%



Pada grafik ini, pola yang dihasilkan dari prediksi dengan 60% data training menunjukkan kesesuaian yang lebih baik dibandingkan dengan hasil pada 70% data training. Meskipun masih terdapat beberapa deviasi, pola prediksi tampak lebih mengikuti pola data aktual, terutama pada tren naik dan turun yang lebih halus. Dengan data training yang lebih sedikit (60%), model tampaknya lebih stabil dalam memprediksi tren, namun mungkin mengorbankan detail fluktuasi tertentu.

d. Forecast 1 Tahun ke depan



Model Hybrid Prophet-GRU mencoba menggabungkan keunggulan Prophet dalam menangkap pola tren dan musiman serta kekuatan GRU dalam memahami pola kompleks data non-linear, menghasilkan prediksi yang berada di antara Prophet (terlalu fluktuatif) dan GRU (lebih stabil). Prediksi hybrid cukup responsif terhadap fluktuasi historis namun kurang akurat dalam menangkap perubahan mendadak dibandingkan GRU. Hasil prediksi 1 tahun ke depan menunjukkan bahwa model hybrid memberikan hasil yang lebih halus dan stabil dibanding Prophet, namun tetap tidak mampu menangkap fluktuasi signifikan yang ada pada data sebenarnya, seperti yang dilakukan GRU. Model hybrid lebih cocok untuk pola data yang stabil, tetapi dalam kasus ini, performanya kurang optimal untuk data dengan dinamika tinggi, karena harus mengorbankan akurasi detail fluktuasi demi prediksi yang lebih stabil.

4.2 Metrik Evaluasi

4.2.1 Model SARIMA

Model	MAE	MAPE	RMSE	R-squared
SARIMA (80% split)	4399.793534431 93	13.37339521788 6067	5821.674299538 3465	-0.230757156574 5299
SARIMA (70% split)	8306.106864553 425	23.77065472373 145	9935.086297142 334	-2.192167251248 5317
SARIMA (60% split)	16122.01567474 8078	52.56339015609 461	17676.21759331 8906	-3.927469343622 4055

Dari hasil evaluasi, dapat disimpulkan bahwa performa model SARIMA sangat bergantung pada proporsi data pelatihan. Skenario 80% split memberikan hasil terbaik dengan tingkat kesalahan yang lebih rendah (MAE, MAPE, dan RMSE) dan kemampuan model yang lebih baik dalam menjelaskan pola data pengujian

dibandingkan skenario lainnya. Penurunan performa terlihat jelas pada split 70% dan semakin buruk pada split 60%, menunjukkan bahwa SARIMA membutuhkan jumlah data pelatihan yang lebih besar untuk menghasilkan prediksi yang akurat.

4.2.2 Model Prophet

Model	MAE	MAPE	RMSE	R-squared
Prophet (80% split)	5568.379017701071	0.18236882978342359	7409.559988851517	-0.9937072926046748
Prophet (70% split)	6415.81287701007	0.19498366126298147	8318.609251651947	-1.2379160950740329
Prophet (60% split)	9896.001094384203	0.5153791861283752	12094.171427575593	-1.3067338739646788

Tabel di atas menunjukkan performa model Prophet berdasarkan pembagian data 80%, 70%, dan 60% untuk pelatihan. Pada split 80%, model menunjukkan performa terbaik dengan nilai MAE sebesar 5568, MAPE 18.24%, dan RMSE 7409, meskipun nilai R2 negatif (-0.99) menunjukkan keterbatasan dalam menjelaskan variabilitas data. Pada split 70%, kesalahan meningkat dengan MAE 6415 dan MAPE 19.49%, sementara pada split 60% terjadi penurunan performa yang signifikan dengan MAE 9896, MAPE 51.54%, dan R2 -1.30. Hasil ini menegaskan bahwa performa Prophet bergantung pada jumlah data pelatihan, dengan split 80% memberikan hasil paling akurat.

4.2.3 Model Hybrid GRU

Model	MAE	MAPE	RMSE	R-squared
Hybrid (80% split)	6154.598145686479	0.20055996370672283	7657.581941578691	-0.9108898171370621
Hybrid (70% split)	6777.196456543721	0.20396656467648677	8217.605563469751	-1.223772837862616
Hybrid (60% split)	7104.085526586817	0.2539820691835167	8290.192182915247	-0.7793002632010746

Hasil evaluasi ini menunjukkan bahwa pembagian data 80% memberikan performa terbaik dibandingkan pembagian 70% dan 60% pada semua metrik evaluasi (MAE, MAPE, RMSE, dan R-squared). MAE, MAPE, dan RMSE terendah pada pembagian 80% menunjukkan bahwa model lebih akurat dalam memprediksi dengan lebih banyak data pelatihan, sedangkan pembagian 60% menghasilkan kesalahan tertinggi. Namun, nilai R-squared negatif pada semua skenario menunjukkan bahwa model kurang mampu menjelaskan variabilitas data target secara efektif, yang mungkin disebabkan oleh kompleksitas data atau ketidaksesuaian model.

4.3 Interpretasi Hasil

KESIMPULAN

DAFTAR PUSTAKA

- Auliya, Y. A., Nurdiansyah, Y. & Astuti, A. P., 2023. Peramalan jumlah pengunjung objek wisata Gumul Paradise Island Kabupaten Kediri menggunakan metode Prophet. *INFORMAL: Informatics Journal*, 8(1), pp. 37–43. Available at: <https://doi.org/10.19184/isj.v8i1.35605>.
- Giovany Syuhada, E. & Helmi Setyawan, M. Y., 2023. Analisis komparasi metode Prophet dan metode Exponential Smoothing dalam peramalan jumlah pengangguran di Jawa Barat: Systematic Literature Review. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(2), pp. 1369–1377. Available at: <https://doi.org/10.36040/jati.v7i2.6827>.
- Makridakis, S., Wheelwright, S.C., and McGee, V.E. 1999. *Metode dan Aplikasi Peramalan*. 2nd ed. Translate by Ir. Hari Suminto. Binarupa Aksara, Jakarta.
- Suryawan, I. G. T., Putra, I. K. N., Meliana, P. M., and Sudipa, I. G. I. 2024. *Performance Comparison of ARIMA, LSTM, and Prophet Methods in Sales Forecasting*. *Sinkron : jurnal dan penelitian teknik informatika*, 8(4), pp. 2410-2421. Available at: <https://doi.org/10.33395/sinkron.v8i4.14057>
- Suwandi, G. (2021). *Model Seasonal Autoregressive Integrated Moving Average (SARIMA) Pada Peramalan Metode Fuzzy Time Series Markov Chain (FTS-MC)*. Fakultas Matematika Dan Ilmu Pengetahuan Alam, Universitas Lampung. pp. 1. Available at: <http://digilib.unila.ac.id/61743/>
- Wardana, R. 2020. *Penerapan Model Gated Recurrent Unit Untuk Peramalan Jumlah Penumpang Kereta Api Di Pt. Kai (Persero)*. Fakultas Ilmu Sains dan Teknologi, Universitas Islam Negeri Syarif Hidayatullah. Available at: <https://repository.uinjkt.ac.id/dspace/handle/123456789/51047>

LAMPIRAN