AITI: Jurnal Teknologi Informasi, Volume 20 No. 2 Agustus 2023, 135-149
ISSN 1693-8348 E-ISSN 2615-7128

JURNAL TEKNOLOGI INFORMASI

Prediksi dan visualisasi penyakit COVID-19 menggunakan kombinasi Prophet dan GeoPandas

Ardito Laksono Suryoputro[1)], Sri Yulianto Joko Prasetyo[2)]
[1,2)]Program Studi Teknik Informatika, Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Jl. Dr. O. Notohamidjodjo Blotongan, Salatiga
Email : 672019057@student.uksw.edu

Riwayat artikel:
Received: 18-04-2023 Revised: 15-05-2023 Accepted: 24-05-2023

Abstract

Covid-19 is spreading very rapidly. Indonesia is one of the countries with the highest cases in Southeast Asia. The purpose of this research is to use machine learning models with the help of tools such as Prophet to predict the trend of the Covid-19 outbreak in Indonesia. Obtained data will be visualized using a Geographic Information System (GIS) with Geopandas, which is used to visualize the spread of Covid-19 in Indonesia. Predictions with three tuning methods using Prophet with trend flexibility and holiday effects scored the best, with 0.68 for RMSLE and 1070 for MAE. Based on the use of Geopandas for Covid-19 cases in Indonesia, Geopandas can be used to visualize geospatial data effectively.

Keywords: *COVID-19, Prophet, Forecasting, Seasonal, GIS*

Abstrak

COVID-19 menyebar dengan sangat pesat. Indonesia menjadi salah satu negara dengan kasus tertinggi di Asia tenggara. Tujuan penelitian ini adalah dengan memanfaatkan model dalam *machine learning* menggunakan bantuan *tools* seperti Prophet untuk memprediksi tren wabah COVID-19 di Indonesia. Data yang diperoleh divisualisasikan menggunakan Sistem Informasi Geografis (SIG) dengan Geopandas untuk memvisualisasikan persebaran COVID-19 di Indonesia. Prediksi dengan tiga metode *tuning* yang dilakukan Prophet dengan *trend flexibility* dan *holiday effect* mendapat skor yang paling baik 0.68 untuk RMSLE dan 1070 untuk MAE. Berdasarkan penggunaan Geopandas untuk kasus COVID-19 di Indonesia, Geopandas dapat digunakan untuk memvisualisasikan data geospasial dengan efektif.

Kata kunci: COVID-19, Prophet, Prediksi, Musiman, SIG

Pendahuluan

Prediksi banyak digunakan untuk membuat keputusan dengan menggunakan data masa lalu dan analisis yang tepat dapat berguna untuk membuat keputusan dan perencanaan strategis. Salah satu contoh kasus yang dapat dapat dibuat adalah prediksi COVID-19. COVID-19 atau juga dikenal sebagai virus corona merupakan pandemi yang sedang melanda dunia saat ini. Virus ini pertama kali ditemukan di Wuhan, China pada akhir tahun 2019 dan sejak saat itu telah menyebar ke seluruh dunia. Virus ini sangat menular dan dapat menyebar melalui tetesan udara yang dihasilkan saat seseorang bersin atau batuk [1]. Indonesia menjadi negara dengan jumlah kasus tertinggi di Asia Tenggara. Hal ini didukung oleh jumlah populasi rakyat Indonesia yang lebih dari 200 juta jiwa. Berdasarkan data yang ada penyakit ini menunjukkan adanya sebuah pola yaitu lonjakan yang tinggi paska libur panjang.

Penelitian ini memiliki tujuan untuk membuat sebuah model *Machine Learning* dengan Prophet untuk melakukan prediksi pertumbuhan kasus COVID-19 di Indonesia dan memberikan visualisasi dengan memanfaatkan Sistem Informasi Geografis. Data yang diolah merupakan data yang didapat dari Kaggle. Data tersebut terdiri atas 31823 baris data yang mencakup data harian kasus COVID-19 di Indonesia dari tahun 2020 hingga 2022. Kemudian dilakukanlah *Exploratory Data Analysis* (EDA) untuk melihat data yang ada. Model akan diuji untuk melihat kinerjanya dengan tes matriks evaluasi. Dari sini kita dapat mengetahui apakah model yang dihasilkan sudah cukup baik dalam membuat prediksi pada kasus COVID-19 di Indonesia. Kemudian, dari hasil prediksi tersebut akan dibuat visualisasi dalam bentuk peta dengan memanfaatkan Sistem Informasi Geografis. Adanya prediksi dan visualisasi tersebut dapat menjadi gambaran ke depannya terkait kasus COVID-19 di Indonesia dan menjadi acuan untuk perencanaan strategi di masa yang akan datang.

Kajian Pustaka

Penelitian yang dilakukan oleh Aditya Satrio, dkk. (2021) [1] bertujuan untuk membandingkan seberapa baik ARIMA dan PROPHET menangani data deret waktu tanpa adanya *tuning seasonality*, memiliki pola acak, dan minim pengamatan dengan menggunakan data kasus COVID-19. Perkiraannya dilakukan dalam jangka waktu 30 hari untuk kedua model dari 22 April 2020 hingga 21 Mei 2020. Baik ARIMA maupun PROPHET ditemukan cukup tidak akurat dalam peramalan seiring berjalannya waktu. PROPHET memiliki akurasi yang baik dalam memprediksi kasus yang dikonfirmasi dengan presisi 91%, sedangkan ARIMA tidak melewati setengah presisi. Hasil penelitian menunjukkan bahwa Prophet secara umum mengungguli ARIMA, meskipun jauh dari data aktual perkiraannya.

Studi empiris yang dilakukan oleh Siami-Namini, dkk. (2019) [2] menunjukkan bahwa algoritma berbasis *deep learning* seperti LSTM mengungguli

ARIMA. Lebih khusus lagi, rata-rata penurunan tingkat kesalahan yang diperoleh LSTM antara 84-87 persen jika dibandingkan dengan ARIMA menunjukkan keunggulan LSTM terhadap ARIMA.

Beberapa studi literatur dilakukan dengan memanfaatkan metode Prophet untuk melakukan *forecasting*. Penelitian yang dilakukan Ye, Z. (2019) [3] menggabungkan metode ARIMA dan Prophet untuk *training* data dari 11 stasiun pemantauan kualitas udara di Shanghai, China. Hasil akhir menunjukkan metode hibrida ini dapat mencapai hasil yang baik dalam memprediksi konsentrasi polutan udara jangka pendek. Studi yang dilakukan oleh Chafiq, dkk. (2020) [4] memberikan mekanisme tren seperti yang dilihat oleh Prophet pada kasus penyebaran pandemi COVID-19 mulai tanggal 21 Januari 2020 hingga 23 September 2020, yang dapat memberikan wawasan yang berharga bagi otoritas kompeten nasional. Hasil dari studi ini juga cukup baik untuk melihat tren dalam jangka pendek.

Penelitian yang dilakukan oleh G. A. Papacharalampous dan H. Tyralis (2018) [5] untuk melakukan evaluasi perbandingan antara metode *Random Forest* dengan metode Prophet. Penelitian ini dilakukan dengan melakukan prediksi terhadap aliran air sungai di Amerika dalam tujuh hari yang akan datang. Hasil dari studi ini menunjukkan *random forest* meramalkan fluktuasi aliran sungai yang tiba-tiba lebih memuaskan dari tiga metode lainnya.

Penelitian yang dilakukan oleh Bashir, dkk. (2021) [6] mengusulkan metode hibrida untuk ramalan beban jangka pendek menggunakan model *Prophet* dan *Long Short Term Memory* (LSTM) untuk mengatasi keterbatasan konvergensi lambat pada model konvensional dan kompleksitas tinggi pada model AI. Hasil menggunakan data beban listrik *Elia Grid* berbasis seperempat jam real time dari tahun 2014 hingga tahun 2021 menunjukkan bahwa metode hibrida mengungguli performa model *standalone* (ARIMA, LSTM, dan Prophet) dengan kesalahan yang lebih sedikit dan waktu komputasi yang lebih cepat. Emir Žunić, dkk. (2020) [7]. Mengaplikasikan penggunaan Prophet untuk prediksi data asli penjualan pada perusahaan ritel terbesar di Bosnia, hasilnya prediksinya dianggap memuaskan.

Vinay Kumar, Lei Zhang (2020) [8] melakukan penelitian dengan menggunakan metode LSTM *networks* untuk melakukan prediksi terhadap penyebaran COVID-19 di Kanada. Penelitian ini mengatakan kebijakan pemerintah setempat sangat berpengaruh pada angka penyebaran COVID-19.Neo Wu, dkk. (2020) [9] menggunakan metode *Deep Transformer* untuk melakukan prediksi deret waktu pada penyakit flu. Penelitian tersebut memanfaatkan parameter sebagai fitur tambahan dan hasilnya fitur tambahan tersebut meningkatkan kinerja model. Metode *Deep Transformer* memberikan hasil ramalan yang cukup baik dibandingkan dengan metode konvensional seperti ARIMA , LSTM.

Jurnal penelitian yang ditulis Shih, dkk. (2019) [10] mengusulkan penggunaan *recurrent neural networks* (RNNs) dengan mekanisme perhatian (*attention mechanism*) untuk meramalkan data *time series multivariat* seperti konsumsi listrik, produksi energi surya, dan lagu piano polifonik. Sourabh Shastri, dkk. (2020) [11] melakukan kajian untuk prediksi COVID-19 dengan studi kasus perbandingan antara kasus di India dengan kasus di Amerika menggunakan metode LSTM seperti *Stacked LSTM, Bi-directional LSTM*, dan *Convolutional LSTM*. Selain itu, tren naik/turun dari ramalan kasus COVID-19 juga divisualisasikan secara grafis. Jurnal oleh Bohdan M. Pavlyshenko(2019)[12] menggunakan beberapa metode *Machine Learning* berbasis XGboost untuk melakukan prediksi terhadap data penjualan. Hasil penelitian menunjukkan bahwa penggunaan teknik *stacking* dapat meningkatkan kinerja model prediktif untuk meramalkan seri waktu penjualan.

Penelitian yang ditulis oleh Dârdală, M., Furtună, T. F., dan Ioniță, C. (2019) [13] membahas pengembangan dan implementasi komponen perangkat lunak yang memungkinkan visualisasi data *geospasial* di dalam perangkat lunak Microsoft Excel. Pada artikel ini Geopandas digunakan pada *script* Python untuk menghasilkan grafik dalam *format png* berdasarkan data yang telah diolah pada *sheet* Excel. Para penulis menyoroti manfaat potensial dari perangkat lunak ini bagi bisnis dan peneliti yang perlu menganalisis dan menyajikan data dalam konteks spasial.

Penelitian yang dilakukan oleh Rojas, dkk. (2023) [14]. Penelitian ini membahas tentang pengaruh positif sistem penyewaan sepeda berbagi yang bergerak bebas pada mobilitas pusat perkotaan, dan pentingnya strategi lokalisasi yang efisien untuk menghindari kerumunan pada jam sibuk dan meningkatkan ketersediaan layanan. Pendekatan ini diimplementasikan dengan menggunakan Python, Geopandas, dan LocalSolver untuk menentukan lokasi stasiun sepeda virtual yang memaksimalkan cakupan sistem.

Penelitian yang dilakukan oleh C. Kavuma, dkk. (2021) [15]. Penelitian ini bertujuan untuk mengevaluasi kemungkinan pengurangan biaya melalui peningkatan pembangkit listrik tenaga listrik bukan melalui pembangkit listrik dengan menggunakan Geopandas dan teknik analisis Geospasial. Data stasiun listrik yang mencakup koordinat dan *rating* daya, data untuk distrik, jalan utama dan kota di Uganda serta populasi diimpor ke dalam jupyter notebook menggunakan bahasa pemrograman Python dan diplot untuk menghasilkan peta stasiun listrik di Uganda. Data pembangkit listrik, *substation*, distrik, dan populasi disimpan dalam Jupyter Notebook dan divisualisasikan dalam GIS.

Berdasarkan penelitian-penelitian yang telah ada sebelumnya, penggunaan Prophet dan Geopandas bisa dikombinasikan di mana Prophet dapat digunakan sebagai *library* Python untuk membuat model untuk prediksi kasus COVID-19 dan Geopandas sebagai *library* untuk memvisualisasikan data hasil prediksi. Prophet

menggunakan gabungan metode *General Additive Model* [16] dan *decomposable time series model* dengan tiga model komponen, yaitu tren, seasonal, dan irregular components [17] yang dapat dirumuskan pada Persamaan (1).

$$y(t) = g(t) + s(t) + h(t) + \varepsilon(t) \quad (1)$$

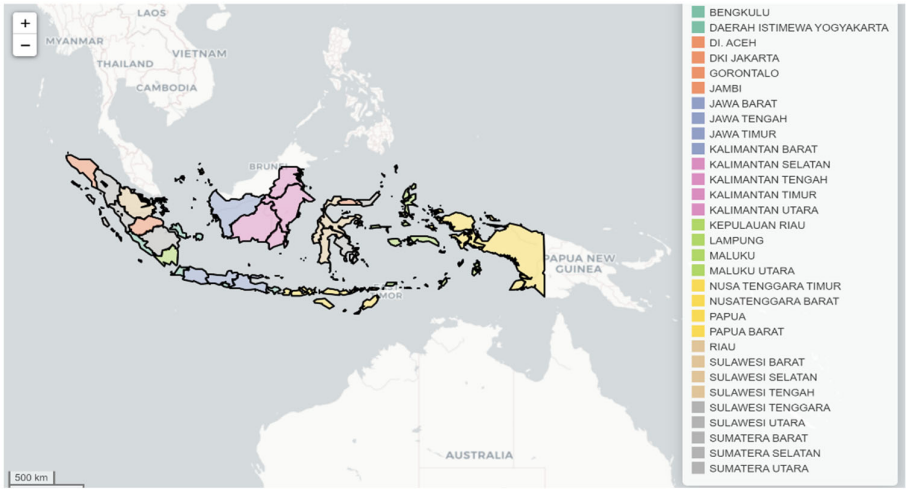
di mana:

- $g(t)$ adalah kurva pertumbuhan linear atau logistik untuk pemodelan perubahan non-periodik dalam seri waktu.
- $s(t)$ menyatakan perubahan periodik musiman seperti mingguan,bulanan,dan tahunan.
- $h(t)$ mewakili efek liburan yang mana perlu ditentukan secara manual oleh pengguna.
- $\varepsilon(t)$ mewakili setiap perubahan yang tidak biasa tidak diakomodasi oleh model.

Metode Penelitian

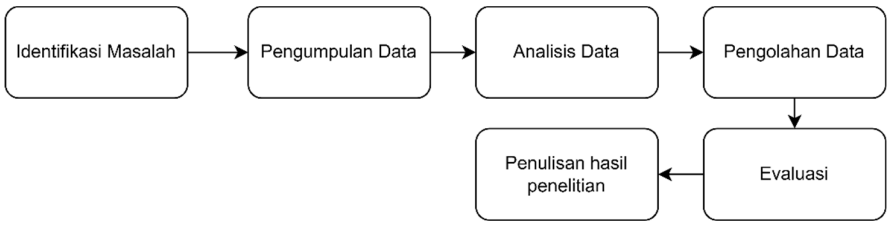
1. Lokasi Studi Kasus

Lokasi dari studi kasus penelitian ini adalah di Indonesia. Indonesia merupakan sebuah negara kepulauan yang terletak di Asia Tenggara. Indonesia juga menjadi negara dengan penduduk terbesar ke-4 di dunia dengan jumlah penduduk lebih dari 270 juta jiwa pada tahun 2020 dan luas wilayah 7,8 juta km^2. Jumlah penduduk yang sangat banyak dan juga luas wilayah yang kecil menjadikan kasus penyebaran COVID-19 di Indonesia menyebar dengan sangat pesat, terlebih hal ini didukung dengan sifat masyarakat yang masih acuh pada saat awal penyebaran kasus COVID-19. Gambar 1 menunjukkan wilayah Indonesia sebagai lokasi studi kasus.



Gambar 1 Wilayah Indonesia sebagai lokasi studi kasus

2. Tahapan Penelitian



Gambar 2 Tahapan penelitian

Tahapan penelitian pada Gambar 2 dapat dijelaskan sebagai berikut.

2.1. Identifikasi Masalah

Peneliti akan melakukan kajian dan analisa permasalahan yang terkait dengan penyakit COVID-19 di Indonesia yang kemudian akan divisualisasikan dan dibuat prediksi di masa depan.

2.2. Pengumpulan Data

Dataset pertama yang digunakan diambil dari Kaggle, *dataset* tersebut terdiri dari 20816 baris dan 38 kolom. *Dataset* ini berisi data kasus COVID-19 di Indonesia yang dimulai dari tanggal 1 Maret 2020 hingga yang paling akhir merupakan tanggal 16 September 2022. *Dataset* tersebut diolah menggunakan layanan Google Colab. Berikut *dataset* yang digunakan yang ditunjukkan pada Gambar 3.

	Date	Location ISO Code	Location	New Cases	New Deaths	New Recovered	New Active Cases	Total Cases	Total Deaths	Total Recovered	Total Active Cases	Location Level	City or Regency	Province	Country
0	3/1/2020	ID-JK	DKI Jakarta	2	0	0	2	39	20	39	-20	Province	NaN	DKI Jakarta	Indonesia
1	3/2/2020	ID-JK	DKI Jakarta	2	0	0	2	41	20	39	-18	Province	NaN	DKI Jakarta	Indonesia
2	3/2/2020	IDN	Indonesia	2	0	0	2	2	0	0	2	Country	NaN	NaN	Indonesia
3	3/2/2020	ID-RI	Riau	1	0	0	1	2	0	0	2	Province	NaN	Riau	Indonesia
4	3/3/2020	ID-JK	DKI Jakarta	2	0	0	2	43	20	39	-16	Province	NaN	DKI Jakarta	Indonesia

Gambar 3 *Preview dataset* COVID-19 di Indonesia

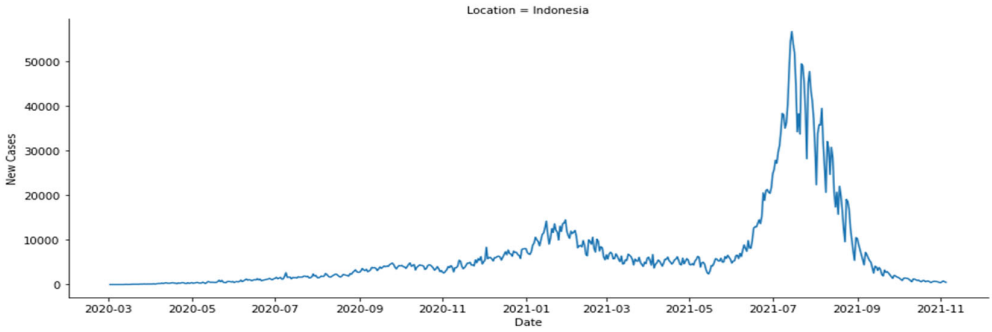
Dataset kedua yang digunakan merupakan data *open source* dengan *format* GeoJson yang berisikan 34 provinsi di Indonesia berikut kode wilayah, letak geografis masing-masing provinsi dalam koordinat *multipolygon* yang digunakan untuk visualisasi wilayah ke dalam bentuk peta. Berikut *preview dataset* yang digunakan pada Gambar 4.

	ID	kode	Propinsi	SUMBER	geometry
0	2	52	NUSATENGGARA BARAT	Peta Dasar BAKOSURTANAL Skala 1 : 250.000	MULTIPOLYGON (((117.62720 -8.50640, 117.63470 ...
1	3	75	GORONTALO	Peta Dasar BAKOSURTANAL Skala 1 : 250.000	POLYGON ((122.18814 1.04530, 122.22627 1.00335...
2	4	74	SULAWESI TENGGARA	Peta Dasar BAKOSURTANAL Skala 1 : 250.000	MULTIPOLYGON (((120.98423 -2.83534, 121.07834 ...
3	5	34	DAERAH ISTIMEWA YOGYAKARTA	Peta Dasar BAKOSURTANAL Skala 1 : 250.000	POLYGON ((110.01183 -7.88690, 110.04295 -7.892...
4	6	33	JAWA TENGAH	Peta Dasar BAKOSURTANAL Skala 1 : 250.000	MULTIPOLYGON (((108.82934 -6.74608, 108.85489 ...

Gambar 4 *Preview dataset* koordinat provinsi di Indonesia

2.3. Analisis Data

Dataset yang dipakai kemudian dicoba untuk dipahami dan dianalisa, agar diperoleh atribut yang tepat untuk digunakan dalam proses pada tahap berikutnya. Proses *Exploratory Data Analysis* (EDA) dilakukan agar lebih mudah untuk memahami dataset yang tersedia. Hal ini juga digunakan di antaranya untuk mengoptimalkan pengetahuan mengenai data, menghasilkan variabel yang penting, mendeteksi *outlier* dan anomali pada data, dan menguji asumsi awal. Berikut hasil visualisasi untuk melihat kurva perkembangan kasus COVID-19 di Indonesia



Gambar 5 Grafik perkembangan kasus COVID-19 di Indonesia

2.4. Pengolahan Data

Dataset kemudian akan diolah melalui beberapa tahap, sehingga datanya lebih rapi dan dapat menghasilkan performa model yang baik. Peneliti mungkin tidak akan menggunakan keseluruhan data dari *dataset* yang ada. Karena kalau dilihat sebelumnya terdapat lonjakan kasus mulai dari bulan Juni 2021 yang mana jika dilihat dari tanggal tersebut merupakan lonjakan kasus pasca lebaran. Tentunya hal ini akan mempengaruhi hasil prediksi model apabila data dari tahun 2020 diikutsertakan.

Beberapa kolom yang dirasa tidak perlu akan didrop, kemudian beberapa baris yang mengandung nilai *NULL* juga akan dihilangkan. Setelah itu *dataset* yang diperoleh memiliki format tanggal *mm-dd-yyyy* akan dikonversi ke format tanggal *yyyy-mm-dd*. Hal ini dilakukan karena Prophet hanya bisa membaca dengan format tanggal *yyyy-mm-dd*.

Terdapat beberapa *library/*modul yang digunakan untuk membantu proses pembuatan model, berikut *library/*modul yang digunakan dalam penelitian yaitu:

a. Prophet adalah *library* yang membantu melakukan olah data untuk keperluan forecasting.
b. Pandas merupakan *library* yang dapat digunakan untuk keperluan analisis data seperti membuat tabel, mengubah dimensi data, mengecek data, dan lain sebagainya.
c. Matplotlib merupakan *library* yang dapat membantu dalam operasi penghitungan matematika.
d. Sklearn adalah *library* yang dapat membantu *clustering* dan klasifikasi.
e. Geopandas merupakan *library* yang digunakan untuk membantu visualisasi data geospasial dalam bentuk peta.
f. Folium adalah *library* Python yang digunakan untuk memvisualisasikan data geospasial dalam bentuk peta interaktif.

2.5. Evaluasi

Pada tahap ini model akan dievaluasi untuk didapati hasil performa dari prediksi yang dihasilkan. Proses ini memisahkan data untuk dibagi menjadi dua bagian yaitu:

a. Data Training
   Data *training* merupakan data yang digunakan untuk melatih algoritma untuk membuat prediksi. Pada penelitian ini data *training* digunakan untuk melatih data *time series* untuk mendapatkan pola dasar seperti *tren* dan *seasonal.*
b. Data Testing
   Data *testing* merupakan data yang digunakan untuk menguji kinerja model yang dihasilkan. Data testing digunakan untuk melakukan *cross validation* untuk mendapati hasil performa dari model pada tahap evaluasi.

Untuk mengukur tingkat *error* dari hasil prediksi, perlu diperhitungkan perbedaan antara hasil aktual dan hasil prediksi. Ada beberapa matriks evaluasi yang dapat digunakan. Pada penelitian ini matriks yang digunakan adalah MAE dan RMSLE. MAE atau *Mean Absolute Error* merupakan salah satu metode pengukuran untuk mengukur rata-rata selisih mutlak antara nilai prediksi dengan nilai aktual. Secara formula MAE dapat dirumuskan pada Persamaan (2).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|A_i - F_i| \quad (2)$$

di mana:

- n adalah panjang sampel/total sampel dari data *time series* yang dipakai
- A_i adalah nilai aktual dari data ke- i
- F_i adalah nilai hasil prediksi dari data ke- i

Secara formula RMSLE atau *Root Mean Squared Logarithmic Error* dapat dijabarkan pada Persamaan (3).

$$RMSLE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\log(p_i + 1) - \log(a_i + 1))^2} \quad (3)$$

di mana:

- n adalah panjang/total dari data *time series* yang diobservasi
- p_i adalah nilai prediksi untuk waktu ke- i
- a_i adalah nilai aktual untuk waktu ke- i

RMSLE digunakan karena memberikan efek *penalty* yang yang lebih besar untuk perkiraan yang terlalu rendah dari nilai aktual daripada yang terlalu tinggi [18]. RMSLE tidak dapat digunakan ketika target/nilai prediksi bernilai negatif. Selain itu alasan RMSLE dipakai pada penelitian ini adalah karena memberikan nilai toleransi dengan lebih baik apabila terdapat *outlier* pada hasil nilai prediksi.

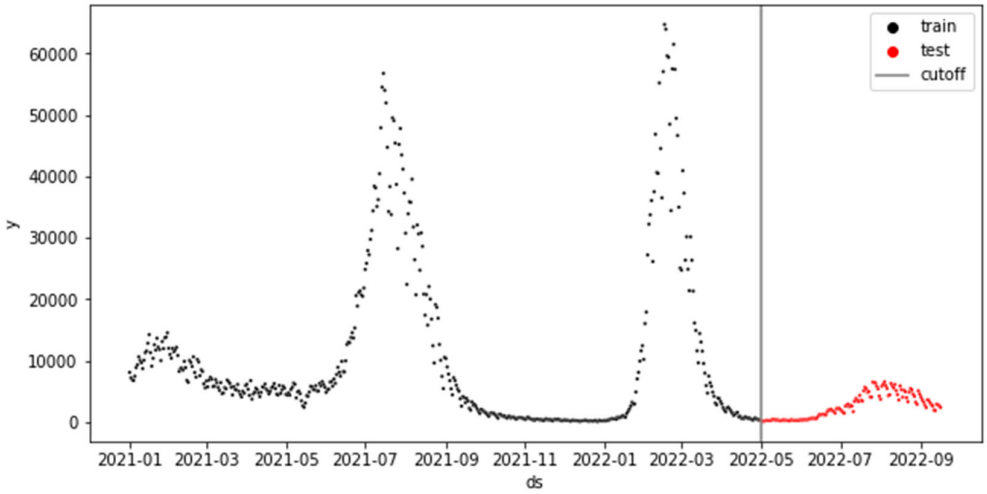
Hasil dan Pembahasan

Metode *machine learning* dengan Prophet mengandalkan data yang bersifat *timeseries* atau data deret waktu. Untuk kasus COVID-19 data yang diolah merupakan data deret waktu harian, seperti yang dijelaskan pada metode penelitian. Data yang diolah terdiri dari 31822 baris data yang mencakup data harian kasus COVID-19 untuk 35 provinsi di Indonesia. Data yang digunakan menggunakan data harian dari tahun 1 Januari 2021 hingga 16 September 2022. Data yang digunakan untuk pengujian skor performa nilai matriks evaluasi merupakan data seluruh kasus aktif baru COVID-19 di Indonesia. Prophet terdapat *tuning parameter* yang dapat digunakan untuk mendapatkan hasil model yang lebih baik.

Data yang telah dibersihkan akan dibagi menjadi dua bagian yaitu data *training* dan data *testing*. Hal ini digunakan untuk mengukur seberapa baik performa model dalam menghasilkan prediksi.

Gambar 6 merupakan visualisasi dari pembagian data *training* dan data *testing*. Data *training* yang digunakan merupakan data untuk tanggal sebelum 1 Mei 2022 dan data *testing* merupakan data setelah tanggal 1 Mei 2022. Visualisasi untuk pembagian data *training* dan data *testing* ditampilkan pada Gambar 6. Data *training* terdiri dari 485 baris dan data *testing* sejumlah 139 baris.

Hasil *output* dari model Prophet selanjutnya akan dievaluasi dengan matriks evaluasi MAE dan RMSLE. Terdapat tiga *parameter* yang digunakan untuk menguji nilai evaluasi untuk prediksi. Tabel 1 menampilkan hasil matriks evaluasi untuk data training dari tiap-tiap model yang dihasilkan serta parameter yang digunakan dalam pembuatan model.



Gambar 6 Grafik pembagian data training dan data testing

Tabel 1 Nilai hasil evaluasi dengan matriks MAE dan RMSLE

Model	Nilai RMSLE	Nilai MAE
Prophet tanpa parameter	1.80	10064
Prophet dengan *trend flexibility* dan *holiday effect*	0.68	1070
Prophet dengan *trend flexibility*, *holiday effect, yearly seasonality, interval width*	0.75	1152

Tabel 1 menunjukkan nilai yang masih kurang akurat seiring dengan makin panjangnya jumlah hari yang diprediksi [1]. Hasil evaluasi yang ditunjukkan mendekati hasil evaluasi yang dilakukan Kumaresan, et.al untuk kasus COVID-19 dengan metode *autofitting* mendapatkan nilai RMSLE 0.778 [18]. Hasil prediksi yang memiliki nilai paling baik ditunjukan pada model Prophet dengan *tuning parameter trend flexibility* dan *holiday effect*. Hal ini menunjukkan, dengan memberikan pengaturan manual pada tanggal liburan dapat menghasilkan nilai evaluasi yang lebih akurat dibandingkan dengan tanpa parameter. Tren *flexibility* merupakan salah satu fitur dalam model time series *forecasting* yang disebut Prophet yang dikembangkan oleh Facebook. *Trend flexibility* mengacu pada kemampuan model untuk menyesuaikan bentuk tren yang muncul pada pengolahan data deret waktu. *Holiday effect* atau efek liburan adalah fenomena di mana data deret waktu dipengaruhi oleh keberadaan hari libur hal ini menyebabkan terjadinya lonjakan yang signifikan contohnya pada kasus COVID-19 di Indonesia terjadi pada musim liburan panjang seperti libur hari raya Idul Fitri dan libur tahun baru terjadi lonjakan kasus aktif baru yang signifikan pada tanggal-tanggal tersebut. Kode program 1 pada Python yang digunakan untuk *setting parameter* efek liburan.

Kode Program 1 *Setting parameter* efek liburan

```
Begin;

1. lebaran= pd.DataFrame({
2.  'holiday': 'liburlebaran',
3.  'ds': pd.to_datetime(['2021-07-17']),
4.  'lower_window': -30,
5.  'upper_window': 35 })
6. nataltahunbaru = pd.DataFrame({
7.  'holiday': 'nataltahunbaru',
8.  'ds': pd.to_datetime(['2022-02-16']),
9.  'lower_window': -30,
10.  'upper_window': 35 })
11. holiday = pd.concat((lebaran, nataltahunbaru))

End;
```

Pada Kode program 1, terdapat *lower_window* dan *upper_window* yang merupakan parameter untuk mengatur jarak waktu sebelum dan setelah hari libur. Misalnya untuk memasukkan efek liburan untuk hari raya Idul Fitri, dengan memasukkan nilai *lower_window* -30 dan *upper_window* 35 artinya model akan memperhitungkan data pada 30 hari sebelum hari H dan 35 hari setelah hari H. Parameter disimpan dalam sebuah *list* yang nantinya akan digunakan sebagai parameter tambahan saat pembuatan model.

```
forecast_test = forecast[forecast['ds'] >= cutoff]
test_rmsle = mean_squared_log_error(y_true=test['y'],
                                    y_pred=forecast_test['yhat']) ** 0.5
test_mae = mean_absolute_error(y_true=test['y'],
                               y_pred=forecast_test['yhat'])
print(test_rmsle)
print(test_mae)
```

```
0.6825022606142913
1070.2025863801653
```

Gambar 7 Nilai hasil evaluasi RMSLE dan MAE dengan Sklearn

Gambar 7 merupakan nilai hasil evaluasi yang diperoleh model dengan Matriks RMSLE 0.68, sedangkan untuk nilai hasil dengan matriks MAE 1070. Metode yang digunakan untuk mendapatkan hasil nilai RMSLE dan nilai MAE di atas adalah *library* Python Sklearn untuk model Prophet dengan *tuning parameter trend flexibily* dan *holiday effect*. Dengan melakukan penyesuaian terhadap parameter ini, model dapat ditingkatkan performanya dalam melakukan prediksi. Hasil evaluasi ini menjadi dasar untuk menyimpulkan bahwa penggunaan model Prophet dengan tuning parameter *trend flexibility* dan *holiday effect* melalui implementasi di library Sklearn dapat meningkatkan performa model.

Dari nilai hasil evaluasi pada Tabel 1, maka model Prophet dengan *tuning parameter trend flexibility* dan *holiday effect* digunakan untuk melakukan pembuatan model selanjutnya untuk prediksi per provinsi di Indonesia. Kode program 2 digunakan untuk pembuatan model tiap provinsi.

Kode Program 2 Pembuatan Model untuk Prediksi tiap Provinsi

```
df_list = []
for i in lokasi:
 #looping per provinsi
 df2 = df[df['Location'] == i]
 #ambil kolom date dan new case kemudian rename m
 kasus = df2[['Date', 'New Cases']].rename(
   columns={'Date': 'ds',
            'New Cases': 'y'})
 #ambil data dari tahun 2021
 kasus = kasus[(kasus['ds'] >= '2021-01-01')]

 # fitting model
 model_tuning_trend = Prophet(
   n_changepoints=25, # default = 25
   changepoint_range=0.8, # default = 0.8
   changepoint_prior_scale=0.05, # default = 0.05
   holidays = holiday,
   interval_width = 0)
 model_tuning_trend.fit(kasus)

 # forecasting
 future = model_tuning_trend.make_future_dataframe(periods=60, freq='D')
 forecast = model_tuning_trend.predict(future)
 forecast = forecast[['ds','yhat']]
 forecast = forecast.rename(columns={'yhat': i})
 df_list.append(forecast)
```

Pada Kode program 2, terdapat perulangan sejumlah nilai pada *list* yang menampung nama-nama provinsi di Indonesia. Model prediksi yang terbuat disimpan pada *list* tampungan yang kemudian akan dilakukan operasi *merge* untuk menggabungkan angka prediksi kasus aktif COVID-19 di Indonesia berdasarkan data tanggal. Model lalu memprediksi hasil untuk 60 hari ke depan dan menyimpan hasil dalam *list dataframe* Pandas.

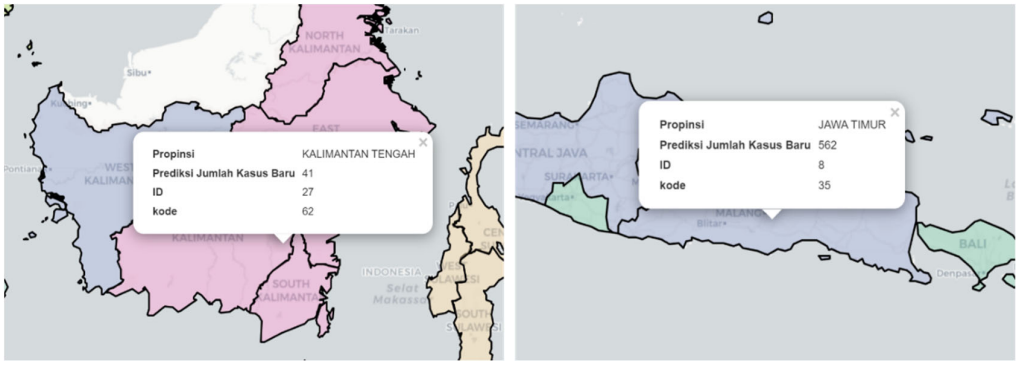
Dataframe Pandas yang menampung hasil prediksi kasus aktif COVID-19 tiap provinsi selanjutnya di *merge* dengan *Dataframe* Geopandas yang menampung nilai geometri masing-masing provinsi di Indonesia yang digunakan untuk pembuatan peta visualisasi prediksi kasus aktif baru COVID-19.

Kode program 3 digunakan untuk memvisualisasikan data geospasial dan data prediksi kasus aktif baru COVID-19 di Indonesia. Hasil visualisasi pada peta dari kode program tersebut dapat dilihat pada Gambar 8.

Kode Program 3 Pembuatan visualisasi peta dengan Geopandas

```
choropleth = indo2.explore(
   column="Propinsi",
   tooltip="Propinsi",
   popup=True,
   tiles="CartoDB Positron",
   cmap="Set2",
   style_kwds=dict(color="black")

choropleth.add_child(folium.map.LayerControl())
choropleth.get_root().html.add_child(folium.Element(f"<h4>Prediksi Jumlah Kasus
Aktif Baru COVID-19 di Indonesia per 13 November 2022</h4>"))
choropleth
```



Gambar 8 Hasil visualisasi prediksi jumlah kasus aktif baru COVID-19

Gambar 8 menunjukkan prediksi jumlah kasus aktif baru COVID-19 di Indonesia untuk tanggal 13 November 2022 pada wilayah provinsi Kalimantan Tengah dan wilayah provinsi Jawa Timur. Visualisasi menggunakan Geopandas dapat menghasilkan peta interaktif yang dapat membantu dalam memahami prediksi penyebaran COVID-19 di berbagai wilayah di Indonesia.

Simpulan

Hasil prediksi untuk data *time series* dengan menggunakan pemodelan *machine learning* Prophet untuk jumlah kasus aktif baru COVID-19 di Indonesia untuk hasil prediksi yang dihasilkan akurasinya sendiri masih kurang akurat .Model masih mengalami kesulitan untuk memprediksi lebih akurat khususnya dikarenakan terdapat anomali dari bulan Juli hingga pertengahan September dan juga pada awal Januari hingga Maret. Hal ini disebabkan oleh efek liburan panjang yang berdampak pada kenaikan yang signifikan pada kasus COVID-19 di Indonesia, terlebih lagi terdapat faktor *human behaviour* yang tentunya akan berpengaruh besar ke dalam perkembangan kasus COVID-19 di Indonesia ke depannya. Hal ini dibuktikan dengan adanya kenaikan skor matriks evaluasi yang lebih akurat ketika diberikan *parameter* tambahan berupa efek liburan. Dengan tiga metode *tuning* yang dilakukan sebagai perbandingan, Prophet dengan *trend flexibility* dan *holiday effect* mendapat skor yang paling baik 0.68 untuk RMSLE dan 1070 untuk MAE. Berdasarkan penggunaan Geopandas untuk kasus COVID-19 di Indonesia, Geopandas dapat digunakan untuk memvisualisasikan data geospasial dengan cara yang efektif dan efisien.

Daftar Pustaka

[1] C. B. Aditya Satrio, W. Darmawan, B. U. Nadia, and N. Hanafiah, “Time series analysis and forecasting of coronavirus disease in Indonesia using ARIMA model and PROPHET,” in *Procedia Computer Science*, 2021. doi: 10.1016/j.procs.2021.01.036.

[2] S. Siami-Namini, N. Tavakoli, and A. Siami Namin, “A Comparison of ARIMA and LSTM in Forecasting Time Series,” in *Proceedings - 17th IEEE International*

Conference on Machine Learning and Applications, ICMLA 2018, 2019. doi: 10.1109/ICMLA.2018.00227.

[3] Z. Ye, "Air Pollutants Prediction in Shenzhen Based on ARIMA and Prophet Method," in *E3S Web of Conferences*, 2019. doi: 10.1051/e3sconf/201913605001.

[4] T. Chafiq, M. Quadoud, and K. Elboukhari, "Covid-19 forecasting in Morocco using FBprophet Facebook's Framework in Python," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 5, p. 8654-8660, Oct. 2020, doi: 10.30534/ijatcse/2020/251952020.

[5] G. A. Papacharalampous and H. Tyralis, "Evaluation of random forests and Prophet for daily streamflow forecasting," *Advances in Geosciences*, vol. 45, pp. 201–208, Aug. 2018, doi: 10.5194/adgeo-45-201-2018.

[6] T. Bashir, C. Haoyong, M. F. Tahir, and Z. Liqiang, "Short term electricity load forecasting using hybrid prophet-LSTM model optimized by BPNN," *Energy Reports*, vol. 8, pp. 1678–1686, Nov. 2022, doi: 10.1016/j.egyr.2021.12.067.

[7] E. Žunić, K. Korjenić, K. Hodžić, and D. Đonko, "Application of Facebook's Prophet Algorithm for Successful Sales Forecasting Based on Real-world Data," *International Journal of Computer Science and Information Technology*, vol. 12, no. 2, 2020, doi: 10.5121/ijcsit.2020.12203.

[8] V. K. R. Chimmula and L. Zhang, "Time series forecasting of COVID-19 transmission in Canada using LSTM networks," *Chaos Solitons Fractals*, vol. 135, 2020, doi: 10.1016/j.chaos.2020.109864.

[9] N. Wu, B. Green, X. Ben, and S. O'Banion, "Deep Transformer Models for Time Series Forecasting: The Influenza Prevalence Case," Jan. 2020, [Online]. Available: http://arxiv.org/abs/2001.08317

[10] S. Y. Shih, F. K. Sun, and H. yi Lee, "Temporal pattern attention for multivariate time series forecasting," *Mach Learn*, vol. 108, no. 8–9, 2019, doi: 10.1007/s10994-019-05815-0.

[11] S. Shastri, K. Singh, S. Kumar, P. Kour, and V. Mansotra, "Time series forecasting of Covid-19 using deep learning models: India-USA comparative case study," *Chaos Solitons Fractals*, vol. 140, 2020, doi: 10.1016/j.chaos.2020.110227.

[12] B. M. Pavlyshenko, "Machine-learning models for sales time series forecasting," *Data (Basel)*, vol. 4, no. 1, 2019, doi: 10.3390/data4010015.

[13] M. DÂRDALĂ, F. T. FURTUNĂ, and C. IONIȚĂ, "DESIGN AND IMPLEMENTATION OF A SOFTWARE COMPONENT FOR GEOSPATIAL DATA VISUALIZATION IN EXCEL," in *Proceedings of the 18th International Conference on INFORMATICS in ECONOMY Education, Research and Business Technologies*, 2019. doi: 10.12948/ie2019.04.22.

[14] C. Rojas, R. Linfati, R. F. Scherer, and L. Pradenas, "Using Geopandas for locating virtual stations in a free-floating bike sharing system," *Heliyon*, vol. 9, no. 1, 2023, doi: 10.1016/j.heliyon.2022.e12749.

[15] C. Kavuma, D. Sandoval, and H. K. Jean de Dieu, "Analysis of power generating plants and substations for increased Uganda's electricity grid access," *AIMS Energy*, vol. 9, no. 1, 2021, doi: 10.3934/ENERGY.2021010.

[16] S. J. Taylor and B. Letham, "Forecasting at Scale," *American Statistician*, vol. 72, no. 1, 2018, doi: 10.1080/00031305.2017.1380080.

[17] C. Chandra and S. Budi, "Analisis Komparatif ARIMA dan Prophet dengan Studi Kasus Dataset Pendaftaran Mahasiswa Baru," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 2, 2020, doi: 10.28932/jutisi.v6i2.2676.

[18] V. Kumaresan *et al.*, "Fitting and validation of an agent-based model for COVID-19 case forecasting in workplaces and universities," *PLoS One*, vol. 18, no. 3 March, 2023, doi: 10.1371/journal.pone.0283517.