

Peramalan Jumlah Kasus COVID-19 Menggunakan *Joint Learning*

<http://dx.doi.org/10.28932/jutisi.v8i1.4469>

Riwayat Artikel

Received: 31 Januari 2022 | Final Revision: 24 Maret 2022 | Accepted: 24 Maret 2022

Muhammad Rizqi Nur[✉]#1, Faris Mushlihul Amin^{#2}, Ahmad Yusuf^{#3}

Prodi Sistem Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sunan Ampel Surabaya
Jl. Ahmad Yani No.117, Surabaya, Indonesia

¹rizqinur2010@gmail.com

²faris@uinsby.ac.id

³ahmadyusuf@uinsby.ac.id

Abstract— COVID-19 is a dangerous illness because it spreads quickly and easily. Vaccines are already available but the pandemic isn't likely to end soon. Forecasting is hoped to help handle the pandemic. Deep learning, especially Long-Short Term Memory (LSTM), has been used to forecast COVID-19 case count in some regions. However, deep learning models generally need a lot of training data while COVID-19 daily data are scarce. However, COVID-19 pandemic happens in many regions. This research aims to use joint learning with data from other regions to improve model performance with fewer data and to use the model to forecast until 9 months since the date of last data taken. Joint learning was done by making models share some parts and training the models together. To overcome the different data scale and pandemic age in the regions, the data was first transformed into discrete Susceptible-Infected-Recovered-Dead (SIRD) Model variables and was evaluated using Root Mean Square Scaled Error (RMSSE). The joint learning done failed to improve the model performance in this research. The proposed model performance was significantly better than the SIRD and the Autoregressive Integrated Moving Average (ARIMA)-SIRD model but wasn't better than the normal encoder-decoder LSTM. The models only reached RMSSE below one occasionally. Additionally, it was found that doing joint learning with all regions without selecting them by clustering can make the model performance worse instead. It was also found that RMSSE is too sensitive to a mostly stagnant time-series due to its division by the error of one-step naïve forecast.

Keywords— COVID-19; Forecasting; Joint Learning; Root Mean Square Scaled Error (RMSSE); Susceptible-Infected-Recovered-Dead (SIRD) Model

I. PENDAHULUAN

COVID-19 adalah penyakit pernapasan yang berbahaya karena dapat menyebar dari manusia ke manusia lain dengan cepat, luas, dan sulit dilacak [1]–[5]. Pada 17 Juni 2020, Indonesia telah menjadi negara dengan jumlah kasus kumulatif COVID total, aktif, harian, dan meninggal tertinggi di ASEAN [6]. Pada 16 April 2021, Jawa Timur adalah provinsi dengan jumlah total kasus tertinggi keempat dengan 143.900 (9.0%) total kasus positif [7].

Meskipun vaksin sudah mulai tersedia, pandemi COVID-19 tidak akan segera berakhir. Vaksin COVID-19 sudah tersedia di Indonesia dan dibagikan secara gratis dan bertahap [8], [9], tapi distribusinya lebih lambat dari rencana [10]. Selain kesulitan distribusi vaksin, golongan yang menolak vaksinasi juga akan menghambat pembangunan *herd immunity* dan penyelesaian pandemi [11]–[14]. Virus COVID-19 juga telah bermutasi menjadi beberapa *strain* baru yang lebih mudah menular dan dapat menginfeksi kembali orang sebelumnya yang telah sembuh dari COVID-19 sehingga vaksin mungkin tidak seefektif yang diharapkan [15], [16].

Indonesia sebagai negara berkembang dengan jumlah populasi terbesar nomor 4 di dunia bisa menerima dampak yang sangat besar dari COVID-19 jika pandemi tidak ditangani dengan baik dan berlangsung lama [17]. Kapasitas layanan kesehatan dan jumlah tes di Indonesia kurang memadai, sedangkan *lockdown* total tidak memungkinkan. Pemerintah dan rumah sakit perlu mengantisipasi lonjakan jumlah kasus agar tidak terjadi lonjakan kematian seperti di Italia [18]. Peramalan jumlah kasus COVID-19 dapat membantu mengantisipasi lonjakan kasus sehingga persiapan bisa lebih matang dan meminimalkan dampaknya [19], [20].

Perkembangan pandemi cukup sulit diramalkan karena banyaknya faktor yang terlibat dan penanganan yang terus ditingkatkan. Pola yang rumit ini akan sulit ditemukan oleh model *machine learning* biasa dengan kapasitas yang terbatas,

tapi *deep learning* memiliki kapasitas yang dapat diatur dan mampu menangkap pola yang rumit dan tidak *linear* [21]. *Deep learning*, khususnya *Long Short Term Memory* (LSTM), telah digunakan untuk meramal perkembangan pandemi COVID-19 dan menunjukkan kinerja yang lebih baik daripada model klasik [20], [22].

Model *deep learning* mampu menangkap pola yang rumit dengan banyaknya parameter di dalamnya. Namun, banyaknya parameter itu mengharuskannya belajar dari data dalam jumlah besar [23], [24], sedangkan data jumlah kasus harian COVID-19 tidak banyak. *Transfer learning* berusaha mengatasi keterbatasan data dengan membuat model *machine learning* menggunakan pengetahuan dari *dataset* atau *model* lain [21], [23].

Transfer learning telah digunakan untuk meramalkan jumlah kasus positif COVID-19 [25], [26]. Namun, *transfer learning* hanya dapat dilakukan dengan dua domain saja, yaitu sumber dan target. Mungkinkah model dilatih dengan lebih banyak domain data lagi untuk meningkatkan kinerja model lebih jauh? *Joint learning* dapat digunakan untuk melatih model dengan data dari lebih dari dua domain. *Joint learning* adalah bentuk *transfer learning* dimana beberapa model dilatih bersamaan dan berbagi sebagian *layer* untuk mempelajari pola umum pada *layer* tersebut [27].

Penelitian ini bertujuan meningkatkan kinerja model LSTM untuk peramalan jumlah kasus COVID-19 pada daerah dengan data relatif sedikit menggunakan *joint learning*. Dengan *joint learning*, model diharapkan dapat menemukan suatu pola umum jumlah kasus pandemi COVID-19 dari banyak daerah yang memiliki dinamika pandemi yang mirip. Untuk mengatasi skala dan tren data serta umur pandemi yang berbeda pada tiap daerah, data ditransformasikan menjadi variabel SIRD diskrit [28], [29]. Agar *joint learning* tidak justru memperburuk model, daerah yang diikutkan dalam *joint learning* dipilih dengan cara *clustering* menggunakan algoritma *K-Means DTW Barycenter Averaging* [30]. Model dibuat untuk meramalkan jumlah kasus positif aktif, sembuh, dan meninggal sejauh 14 hari ke depan.

II. METODE

Alur penelitian ini dibuat berdasarkan lima langkah dasar peramalan [31] dan metodologi praktis *deep learning* [21].

A. Pengumpulan Data

Data utama yang digunakan dalam penelitian ini adalah data perkembangan harian jumlah kumulatif kasus positif aktif, sembuh, dan meninggal COVID-19 di kota atau kabupaten di Jawa Timur; diperoleh dari *website* JATIM TANGGAP COVID-19 milik Pemerintah Provinsi Jawa Timur [32]. Untuk menghitung variabel SIRD dari jumlah kasus, diperlukan data populasi daerah yang diperoleh dari *website* BPS Jawa Timur [33]. Selain jumlah kasus sebagai variabel endogen, digunakan empat variabel *eksogen*:

1. Jumlah tes COVID-19 harian karena jumlah kasus positif yang tercatat tergantung pada jumlah tes yang dilakukan [29]; diperoleh dari *repository* Github Our World In Data (OWID) [34]
2. Jumlah kumulatif orang dengan vaksinasi COVID-19 dosis penuh karena vaksin mengurangi penyebaran virus; diperoleh dari *repository* Github OWID [35]
3. Periode Pembatasan Sosial Berskala Besar (PSBB) sebagai *local lockdown* karena efektif mengurangi penyebaran COVID-19 [18], [25], [36], [37]; direkap secara manual dari situs berita
4. Tanggal penting lain yang mempengaruhi mobilitas, misalnya libur panjang atau hari raya idul fitri, karena mobilitas mempengaruhi tingkat penyebaran COVID-19 [38], [39], direkap secara manual dari kalender akademik [40] dan situs berita.

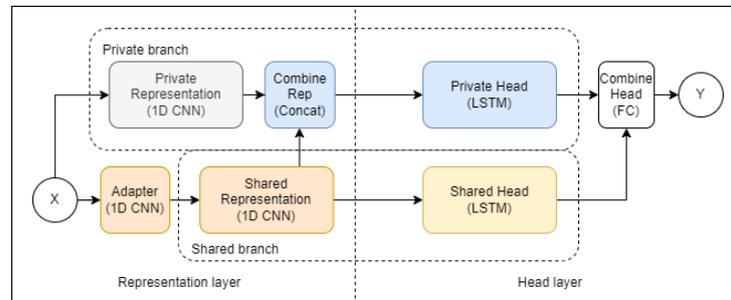
B. Eksplorasi Data

Eksplorasi data dilakukan untuk memahami data. Eksplorasi data *time series* dilakukan dengan grafik. Matriks korelasi Kendall-Tau digunakan untuk mengetahui korelasi variabel *eksogen*, khususnya yang non-*biner*, yaitu jumlah tes dan vaksinasi COVID-19; variabel dengan koefisien korelasi diatas 0.3 digunakan [41]. Untuk menyeleksi tanggal penting, dilakukan observasi pada grafik jumlah kasus positif aktif, perubahannya (*delta/differencing*), dan variabel SIRD beta. Observasi dilakukan untuk melihat pola kasat mata antara tanggal dan jumlah kumulatif kasus positif aktif, perubahannya, dan variabel beta dari semua kabupaten/kota. Observasi hanya dilakukan pada ketiga grafik tersebut karena tanggal tersebut hanya diperkirakan atau terbukti mempengaruhi mobilitas atau perilaku penduduk dan, sebagai akibatnya, jumlah kasus positif aktif. Tanggal yang terlihat berpengaruh untuk lebih dari 50% kabupaten/kota digunakan dalam penelitian ini.

C. Implementasi Model

Dalam tahap ini, model diimplementasikan untuk memungkinkan *joint learning*, namun *training* belum dilakukan sehingga yang dibuat hanya kerangkanya saja (*class*). Model dibuat dengan bahasa Python menggunakan *framework* PyTorch [42]. Model dibuat untuk meramalkan jumlah kasus positif aktif, sembuh, dan meninggal sejauh 14 hari ke depan. Jangka waktu peramalan ini cukup bagi rumah sakit untuk meningkatkan kapasitas ICU dan alih fungsi ruangan dan peralatan [43], serta bagi pemerintah untuk menentukan periode pembatasan sosial. *Long Short Term Memory* (LSTM) dilaporkan mampu menghasilkan ramalan akurat hingga 30 hari [44].

Joint learning dipilih sebagai bentuk *transfer learning* yang dapat menggunakan lebih dari satu sumber untuk memaksimalkan penggunaan data daerah-daerah terjangkau pandemi. Tiap daerah yang diikuti dalam *joint learning* memiliki model masing-masing, namun berbagi *shared branch* yang sama [45]. *Joint learning* [27] dilakukan dengan melatih model-model tersebut secara bersamaan sehingga *shared branch* akan mempelajari representasi atau pola umum. Untuk memfasilitasi itu, terdapat *representation layer* yang dibuat dengan *one-dimensional convolution* [46], [47], karena *convolution* merupakan *neural network* yang baik untuk membangun representasi data [46]–[48] dan telah digunakan untuk membangun representasi untuk *Recurrent Neural Network* (RNN) [49]. Arsitektur keseluruhan model digambarkan pada Gambar 1.



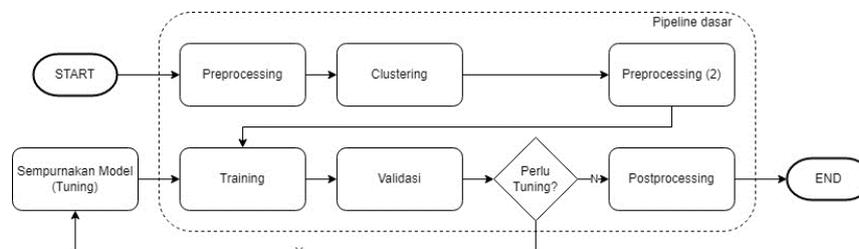
Gambar 1. Arsitektur model yang diajukan

Percabangan *private branch* dan *shared branch* dilakukan agar model dapat mempelajari pola khusus pada *private branch* sehingga meringankan beban *shared branch* yang harus belajar dari banyak domain [45]. Percabangan dalam model ini didesain sedemikian rupa sehingga *shared branch* tidak terpengaruh *private branch* tapi *private branch* dapat menggunakan *output* dari *shared branch* dengan menggabungkannya pada blok *combine*. Pada blok *combine*, data dari *private branch* diboboti lebih kecil untuk memaksa model mengutamakan untuk belajar pada *shared branch*. Bobot ini adalah *learnable parameter* sehingga akan membesar jika model memang mempelajari pola khusus pada *private branch*.

Sebelum input diberikan pada *shared representation*, data ditransformasikan terlebih dahulu dengan blok *adapter*, yang juga merupakan *one-dimensional convolution*, untuk sedikit melakukan *domain adaptation*. Semua *convolution layer* terhubung secara *residual* agar kedalaman model tidak memperburuk model [50].

D. Pembangunan Pipeline

Pipeline secara keseluruhan digambarkan pada Gambar 2.



Gambar 2. Pipeline keseluruhan

1) Preprocessing

Data perlu disiapkan dulu agar dapat digunakan dengan baik; ini dilakukan dalam tahap *preprocessing* [51]. Pertama-tama, data dimuat dulu untuk tiap kota/kabupaten. Lalu, data yang kosong (semuanya nol) di awal atau akhir *time series* dihilangkan. Data kosong di awal *time series* berarti pandemi belum dimulai untuk daerah itu, sedangkan data kosong di akhir *time series* dianggap hilang. Kemudian, data kosong yang berada di tengah-tengah dan akhir *time series* diisi dengan interpolasi linear. Setelah itu, kasus positif aktif dihitung dengan mengurangi kasus positif total dengan kasus positif sembuh dan meninggal. Data jumlah tes kumulatif diubah menjadi jumlah tes harian. Data periode PSBB dan tanggal penting diubah menjadi *time series* biner. Data jumlah kasus COVID-19 lalu ditransformasikan menjadi variabel SIRD (*susceptible-infected-recovered-dead*) diskrit untuk menghilangkan tren dan menghilangkan bias populasi dan lama pandemi daerah. Transformasi ke variabel SIRD dilakukan dengan rumus pada persamaan (1).

$$\begin{aligned} \gamma_t &= \frac{\Delta R_t}{I_{t-1}} \\ \delta_t &= \frac{\Delta D_t}{I_{t-1}} \\ \beta_t &= \frac{N \times (\Delta I_t + \Delta R_t + \Delta D_t)}{S_{t-1} I_{t-1}} \end{aligned} \quad (1)$$

Keterangan:

- S jumlah orang yang belum pernah sakit
- I jumlah kasus positif aktif, orang yang sakit dan dapat menginfeksi
- R jumlah kasus sembuh
- D jumlah kasus meninggal
- β kecepatan perpindahan dari kompartemen S ke I
- γ kecepatan perpindahan dari kompartemen I ke R
- δ kecepatan perpindahan dari kompartemen I ke D
- t waktu (hari) ke-t
- ΔI_t perbedaan nilai I pada waktu t dengan waktu t-1

Data dibagi menjadi empat kelompok pembatasan data, yaitu 3 bulan (90 hari), 6 bulan (180 hari), 1 tahun (366 hari), serta data sebelum ada vaksinasi (hingga 21 Januari 2021). Ini dilakukan untuk melihat kinerja model dalam tingkat keterbatasan data yang berbeda. Kelompok 3 bulan mewakili fase awal pandemi ketika pandemi baru mulai menyebar dan kesadaran masyarakat masih rendah. Kelompok 6 bulan mewakili perkembangan pandemi hingga fase menengah ketika pandemi memuncak lalu menurun dengan digalakkannya kebijakan-kebijakan pemerintah untuk menekan angka pandemi dan masyarakat yang mulai sadar dan patuh. Kelompok 1 tahun mewakili perkembangan pandemi hingga ia mulai reda dan, jika ada, munculnya gelombang baru. Kelompok sebelum vaksinasi ditambahkan karena vaksin mungkin mempengaruhi perkembangan pandemi sedangkan datanya masih sedikit ketika penelitian dilakukan.

Data tiap daerah dari tiap kelompok dibagi lagi menjadi 50% data *training*, 25% data validasi, dan 25% data *testing* [52]. Data *training* digunakan untuk melatih model, data validasi digunakan untuk penyempurnaan model (*tuning*), dan data *evaluasi* digunakan untuk mengukur kinerja model untuk data yang belum pernah dilihat seperti ketika penggunaan sesungguhnya nanti.

2) Clustering

Clustering dilakukan dilakukan pada data *time series* variabel SIRD diskrit menggunakan algoritma *K-Means DTW* (*Dynamic Time Warping*) *Barycenter Averaging* (DBA) [30] dari *library tslearn* [53]. Jumlah *cluster* (n) dicoba dari 2 hingga 10 dan diambil nilai yang menghasilkan rata-rata koefisien *silhouette* tertinggi sebagai ukuran kebaikan *clustering* [54]. Nilai *silhouette* dapat dihitung dengan rumus pada persamaan (2) *Clustering* dapat menghasilkan *cluster* yang hanya memiliki satu anggota; *cluster* ini berikutnya akan disebut *single cluster*. *Single cluster* dianggap sebagai *outlier* dan tidak digunakan. Daerah dengan jumlah data paling sedikit (yaitu dengan data pertama paling akhir) dari tiap *cluster* diambil sebagai target *joint learning*.

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad (2)$$

Keterangan:

- s(i) nilai *silhouette* data ke-i
- a(i) jarak rata-rata antara data ke-i dengan data lain dalam cluster yang sama
- b(i) nilai minimum jarak rata-rata antara data ke-i dengan data lain dalam cluster lain

3) Preprocessing (2)

Setelah *clustering*, data tiap daerah dinormalisasikan ke dalam jangkauan 0 hingga 1 bersamaan dengan daerah-daerah dalam *cluster* yang sama. Data tiap daerah lalu dipotong menjadi beberapa sampel dengan panjang 44 atau lebih (masih dalam bentuk pandas DataFrame), yaitu 14 label sebagai jarak peramalan dan setidaknya 30 prediktor sebagai dasar peramalan. Data lalu dipisahkan label dan variabel *eksogen*-nya. Kemudian, data dijadikan numpy *array* dan dimuat dalam PyTorch DataLoader untuk digunakan dengan model PyTorch.

4) Training

Joint learning dilakukan dengan melatih model dari *cluster* yang sama secara bersamaan. Semua model dari *cluster* yang sama berbagi *shared branch* yang sama. *Training* dilakukan sebanyak jumlah *batch* paling sedikit dari semua daerah untuk menghindari bias jumlah data, sehingga tidak semua *batch* digunakan dari dataset dengan banyak *batch* pada sebuah iterasi. *Batch* diambil secara acak. *Loss* daerah sumber diboboti lebih kecil agar pelatihan model fokus pada target. *Training* dilakukan hingga nilai validasi mulai naik.

Training dilakukan dengan *Mean Square Scaled Error* (MSSE) sebagai *loss function*. MSSE didapat dari menghilangkan akar dari *Root Mean Square Scaled Error* (RMSSE) [55], [56] untuk memudahkan perhitungan gradien. MSSE adalah MSE (*mean squared error*) peramalan dibagi MSE peramalan naif satu langkah dari prediktor (persamaan (3)).

$$MSSE = \frac{\frac{1}{h} \sum_{t=n+1}^{n+h} (y_t - \hat{y}_t)^2}{\frac{1}{n-1} \sum_{t=2}^n (y_t - y_{t-1})^2} \quad (3)$$

Keterangan:

- h panjang *horizon forecast* (panjang peramalan)
- n jumlah data prediktor
- y_t nilai sebenarnya pada waktu t
- \hat{y}_t nilai ramalan pada waktu t

5) Validasi

Validasi model ramalan dilakukan dengan menghitung nilai MSSE dari variabel SIRD diskrit. Validasi dilakukan untuk mengukur kinerja model pada data di luar *training set* dan untuk penyempurnaan model (*tuning*). Validasi dilakukan menggunakan data validasi yang telah disisihkan sebelumnya. Validasi MSSE digunakan untuk mengidentifikasi *overfitting* serta menghindarinya dengan menghentikan *training* ketika *loss* validasi mulai naik [52] atau jika tidak ada perkembangan dalam waktu yang lama. Nilai MSSE validasi diusahakan serendah mungkin dan diharapkan di bawah satu.

6) Post processing

Hasil ramalan model *deep learning* tidak bisa langsung digunakan atau dievaluasi karena label juga ditransformasikan pada tahap *preprocessing*. Tahap *post processing* meliputi langkah-langkah yang perlu dilakukan untuk mengembalikan ramalan ke bentuk semula, yaitu jumlah kasus COVID-19. Proses ini dilakukan dengan melakukan kebalikan dari tiap langkah transformasi variabel SIRD dengan urutan yang juga berkebalikan. Variabel SIRD dikembalikan menjadi jumlah kasus menggunakan persamaan (4).

$$\begin{aligned} \Delta R_t &= \gamma_t I_{t-1} \\ \Delta D_t &= \delta_t I_{t-1} \\ \Delta S_t &= -\frac{\beta_t S_{t-1} I_{t-1}}{N} \\ \Delta I_t &= \frac{\beta_t S_{t-1} I_{t-1}}{N} - (\Delta R_t + \Delta D_t) \end{aligned} \quad (4)$$

Keterangan:

- S jumlah orang yang belum pernah sakit
- I jumlah kasus positif aktif, orang yang sakit dan dapat menginfeksi
- R jumlah kasus sembuh
- D jumlah kasus meninggal
- β kecepatan perpindahan dari kompartemen S ke I
- γ kecepatan perpindahan dari kompartemen I ke R
- δ kecepatan perpindahan dari kompartemen I ke D
- t waktu (hari) ke-t
- ΔI_t perbedaan nilai I pada waktu t dengan waktu t-1

E. Penyempurnaan Model (Tuning)

Penyempurnaan model dilakukan dengan penyesuaian *hyperparameter* menggunakan Optuna [57]; sebagian *hyperparameter* mempengaruhi arsitektur model. Model untuk semua target berbagi *hyperparameter* yang sama. Optuna

mengoptimasi *hyperparameter* dengan rata-rata nilai *Mean Square Scaled Error* (MSSE) validasi sebagai nilai objektif yang diminimalkan. Keseluruhan *pipeline* dieksekusi untuk setiap percobaan *hyperparameter* atau desain; model dilatih ulang setiap kali eksekusi. Penyempurnaan model dilakukan selama dua minggu. Proses ini dilakukan secara paralel menggunakan 3x2 proses CPU dengan *backend storage* PostgreSQL.

F. Evaluasi

Evaluasi dilakukan untuk mengukur kinerja model untuk data baru [21], mengetahui kebergunaan tiap bagian model [25], dan membandingkan model dengan model lain. Untuk mengukur kinerja model untuk data baru, model dilatih menggunakan *training set* hingga nilai validasi mulai naik untuk menghindari *overfitting* atau setidaknya 100 *epoch*, lalu dievaluasi menggunakan *test set* yang belum pernah dilihat oleh model [52]. Kinerja yang diukur ini menjadi estimasi kinerja model untuk peramalan sesungguhnya.

Evaluasi model ramalan dilakukan dengan menghitung nilai *Root Mean Square Scaled Error* (RMSSE) [55], [56] rata-rata dari ramalan jumlah total kasus positif, sembuh, dan meninggal. RMSSE adalah metode pengukuran kinerja ramalan yang tidak terpengaruh skala, kokoh, dan dapat digunakan dalam segala situasi [56]. RMSSE didefinisikan sebagai akar dari pembagian MSE (*mean squared error*) peramalan dengan MSE peramalan naif satu langkah dari prediktor. Rumus RMSSE dicantumkan pada persamaan (5).

$$RMSSE = \sqrt{\frac{\frac{1}{h} \sum_{t=n+1}^{n+h} (y_t - \hat{y}_t)^2}{\frac{1}{n-1} \sum_{t=2}^n (y_t - y_{t-1})^2}} \quad (5)$$

Keterangan:

- h panjang *horizon forecast* (panjang peramalan)
- n jumlah data prediktor
- y_t nilai sebenarnya pada waktu t
- \hat{y}_t nilai ramalan pada waktu t

Model dievaluasi untuk tiap daerah target dari tiap kelompok jumlah data, yaitu dengan data 3 bulan (90 hari), 6 bulan (180 hari), 1 tahun (366 hari), serta data sebelum ada vaksinasi (hingga 21 Januari 2021). Pembagian data menjadi beberapa kelompok dengan jumlah data berbeda dilakukan untuk mengukur kinerja model dengan tingkat keterbatasan data yang berbeda. Model hanya diharapkan mencapai nilai RMSSE di bawah satu (lebih baik daripada model *naïve* untuk peramalan dalam sampel) dan memiliki kinerja lebih baik daripada model perbandingan.

Signifikansi perbedaan model diuji dengan rata-rata *rank* menggunakan *Adjusted Friedman test* dan *Bonferroni-Dunn post-hoc test* [58]. *Friedman Test* (persamaan (6)) lebih kuat daripada *Analysis of Variance* (ANOVA) ketika asumsi ANOVA tidak terpenuhi, khususnya untuk *rank* yang merupakan data ordinal. Karena statistik Friedman dinilai terlalu konservatif, disarankan menggunakan *Adjusted Friedman Statistics* (persamaan (7)) oleh Iman dan Davenport. Jika H_0 Friedman Test ditolak dan hanya dilakukan perbandingan satu model kontrol dengan banyak model (model lainnya tidak dibandingkan dengan satu sama lain), pengujian dilanjutkan dengan *Bonferroni-Dunn post-hoc test* untuk melihat dimana perbedaan signifikan berada. Kinerja dua model berbeda secara signifikan jika rata-rata *rank* berbeda setidaknya sebesar *critical difference* CD (persamaan (8) dan Tabel 1).

$$\chi_F^2 = \frac{12N}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right] \quad (6)$$

$$F_F = \frac{(N-1) \chi_F^2}{N(k-1) - \chi_F^2} \quad (7)$$

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \quad (8)$$

Keterangan:

- χ_F^2 Statistik *Friedman*

- F_F Adjusted Friedman Statistic
- CD Critical Difference, perbedaan rata-rata *rank* minimum untuk dapat dinilai signifikan
- R_j Rata-rata *rank* algoritma ke-j
- N jumlah dataset yang digunakan dalam perbandingan
- k jumlah algoritma atau model yang dibandingkan
- α tingkat kesalahan uji statistik

TABEL 1
NILAI q_α UNTUK *BONFERRONI-DUNN TEST* DENGAN $\alpha = 0.05$ [58]

Jumlah Model	2	3	4	5	6	7	8	9	10
$q_{0.05}$	1.960	2.241	2.394	2.498	2.576	2.638	2.690	2.724	2.773

Model dibandingkan dengan enam model *baseline* untuk mengetahui kebergunaan tiap bagian model [25]. Seperti model utama, model *baseline* juga dilatih dengan Optuna untuk menyesuaikan *hyperparameter*. Model dibandingkan dengan model *baseline* berikut:

1. Model dasar (*Long-Short Term Memory encoder-decoder*) tanpa joint learning
2. Model tanpa representation layer
3. Model utuh dengan shared branch yang hanya dilatih dengan satu daerah dengan data terbanyak dalam cluster yang sama
4. Model utuh dengan shared branch yang dilatih dengan semua dataset
5. Model utuh yang sepenuhnya shared
6. Model utuh yang sepenuhnya private

Selain dengan model *baseline*, model dibandingkan dengan model lain. Penelitian ini menggunakan model berikut sebagai pembandingan:

1. Model *Autoregressive Integrated Moving Average* (ARIMA) pada variabel SIRD diskrit
2. Model SIRD (*Susceptible-Infected-Recovered-Dead*)

III. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Data jumlah kasus kumulatif harian COVID-19 tiap kabupaten/kota Jawa Timur diperoleh dari *website* JATIM TANGGAP COVID-19 milik Pemerintah Provinsi Jawa Timur [32]. Didapatkan data dari 38 kabupaten/kota dengan data terpendek dari Kabupaten Sampang dengan 313 data (mulai 12 Mei 2020), data terpanjang dari Kota Surabaya, Kota Malang, Kabupaten Magetan, dan Kabupaten Sidoarjo dengan 366 data (mulai 20 Maret 2020), dan rata-rata 350,5 data. Sebagian kabupaten/kota memiliki data yang hilang pada hari tertentu.

Data jumlah tes dan vaksinasi kumulatif harian diperoleh dari *repository* Github Our World In Data (OWID) [34], [35]. Data dari Github OWID yang tersedia adalah data per negara atau negara bagian AS, sehingga data tes dan vaksinasi yang dapat diperoleh adalah data keseluruhan Indonesia (data tes dan vaksinasi Jawa Timur tidak tersedia). Data dari Github OWID diperoleh dengan cara mengunduh data tersebut. Data jumlah tes berawal dari sebelum 20 Maret 2020, sedangkan data jumlah vaksinasi berawal dari 12 Januari 2021. Kedua data ini juga memiliki kekosongan pada hari tertentu.

Rekap data periode pembatasan sosial terbaru tidak ditemukan, sehingga data periode pembatasan sosial dikumpulkan dari situs-situs berita secara manual. Ditemukan bahwa Periode Pembatasan Sosial Berskala Besar (PSBB) seiring waktu digantikan oleh Pemberlakuan Pembatasan Kegiatan Masyarakat (PPKM) dan PPKM mikro, sehingga keduanya juga digunakan. Di antara 38 kabupaten/kota, 6 melakukan PSBB, 29 melakukan PPKM, dan 38 (semuanya) melakukan PPKM mikro.

Tanggal penting yang digunakan adalah tanggal yang mungkin mempengaruhi mobilitas, contohnya libur panjang dan tanggal yang diperkirakan menimbulkan lonjakan jumlah kasus menurut situs-situs berita. Tanggal penting ini diperoleh dari kalender [40] dan situs-situs berita yang menyeringkannya. Ditemukan 14 tanggal libur panjang, yaitu libur lebih dari satu hari, dan 12 di antaranya diberitakan untuk waspada lonjakan COVID-19. Selain libur panjang, digunakan pula periode bulan Ramadhan dan tanggal pilkada yang dilakukan oleh 19 kabupaten/kota. Digunakan atau tidaknya tanggal-tanggal ini ditentukan melalui eksplorasi data.

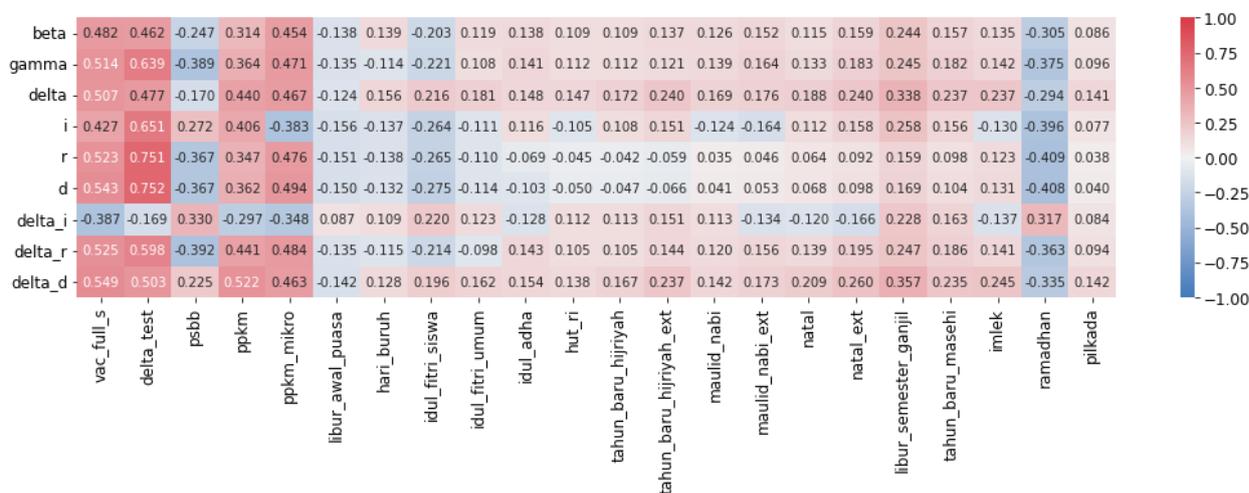
B. Eksplorasi Data

Matriks korelasi yang dihasilkan dicantumkan pada Gambar 3. Dari matriks korelasi tersebut, didapat variabel berikut: Δ_{test} , vac_full_s , $ppkm$, $ppkm_mikro$, $psbb$, $ramadhan$.

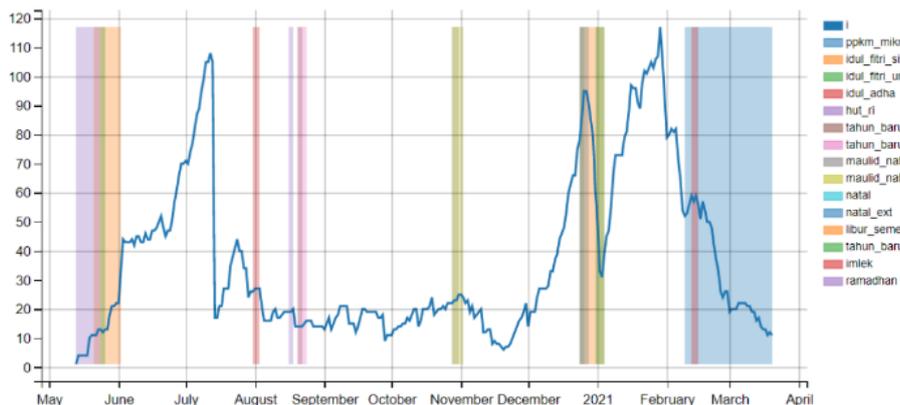
Untuk menyeleksi tanggal penting, dilakukan observasi pada grafik jumlah kasus positif aktif, perubahannya, dan variabel beta dari variabel SIRD (*Susceptible-Infected-Recovered-Dead*) (Gambar 4). Tanggal yang terlihat berpengaruh untuk lebih dari 50% kabupaten/kota digunakan dalam penelitian ini. Hasil observasi dipaparkan pada Tabel 2.

TABEL 2
HASIL OBSERVASI TANGGAL YANG BERPENGARUH TERHADAP POLA JUMLAH KASUS POSITIF AKTIF (I), PERUBAHANNYA (DELTA I), DAN BETA.

Grafik	Tanggal-Tanggal yang Berpengaruh
Jumlah kasus positif aktif (I)	hut_ri, idul_adha, idul_fitri_siswa, idul_fitri_umum, libur_semester_ganjil, Maulid_nabi, Maulid_nabi_ext, natal, natal_ext, pilkada, pppk, pppk_mikro, psbb, tahun_baru_hijriyah, tahun_baru_hijriyah_ext, tahun_baru_masehi
Perubahan jumlah kasus positif aktif (Delta I)	hut_ri, idul_adha, idul_fitri_siswa, idul_fitri_umum, libur_semester_ganjil, Maulid_nabi, Maulid_nabi_ext, natal, natal_ext, pilkada, pppk, pppk_mikro, psbb, tahun_baru_hijriyah, tahun_baru_hijriyah_ext, tahun_baru_masehi
Beta	hari_buruh, idul_adha, idul_fitri_siswa, idul_fitri_umum, imlek, libur_awal_puasa, libur_semester_ganjil, Maulid_nabi, Maulid_nabi_ext, natal, natal_ext, pilkada, pppk, pppk_mikro, psbb, ramadhan, tahun_baru_hijriyah, tahun_baru_hijriyah_ext, tahun_baru_masehi
Gabungan	hari_buruh, hut_ri, idul_adha, idul_fitri_siswa, idul_fitri_umum, imlek, libur_awal_puasa, libur_semester_ganjil, Maulid_nabi, Maulid_nabi_ext, natal, natal_ext, pilkada, pppk, pppk_mikro, psbb, ramadhan, tahun_baru_hijriyah, tahun_baru_hijriyah_ext, tahun_baru_masehi



Gambar 3. Matriks korelasi variabel *eksogen* termasuk tanggal terhadap jumlah kasus positif, sembuh, dan meninggal harian, perubahannya, dan variabel SIRD beta, gamma, dan delta



Gambar 4. Grafik jumlah kasus positif aktif Kab. Sampang dan tanggal-tanggal penting

C. Clustering

Hasil *clustering* untuk tiap kelompok dipaparkan pada Tabel 3. N adalah jumlah *cluster* optimal yang ditentukan berdasarkan nilai siluet terbesar. Beberapa nilai N menghasilkan jumlah *cluster* yang hanya memiliki satu anggota, yang

dalam penelitian ini disebut *single cluster*. *Single cluster* ini tidak dapat digunakan dalam *joint learning* dan dibuang sebagai *outlier* karena berarti anggotanya tidak mirip dengan daerah lain manapun. Pembagian kelompok dilakukan untuk melihat kinerja model untuk tingkat keterbatasan berbeda (selengkapnya telah dijelaskan dalam bab metode, bagian *preprocessing*). Dari tiap *cluster*, kabupaten/kota dengan data paling sedikit dipilih untuk menjadi *target*, yaitu daerah fokus penelitian ini.

TABEL 3
HASIL CLUSTERING

Kelompok	N	Silhouette	Non-Single	Target
90 Hari	4	0.2667	4	Kab. Tuban, Kab. Bojonegoro, Kota Madiun, Kab. Sampang
180 Hari	4	0.5383	3	Kab. Tuban, Kab. Sampang, Kota Pasuruan
366 Hari	4	0.4075	3	Kab. Tuban, Kab. Sampang, Kota Pasuruan
Sebelum Vaksinasi	2	0.4734	2	Kab. Sampang, Kota Pasuruan

D. Evaluasi

Evaluasi model ramalan dilakukan dengan menghitung nilai *Root Mean Square Scaled Error* (RMSSE) rata-rata dari ramalan jumlah total kasus positif, sembuh, dan meninggal. Model dievaluasi untuk tiap daerah target dari tiap kelompok jumlah data. Model diharapkan mencapai nilai RMSSE di bawah satu dan memiliki kinerja lebih baik daripada model perbandingan. Model yang diajukan akan disebut sebagai model utama.

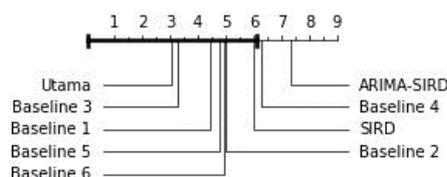
1) Kelompok 90 Hari

Kelompok pertama ini memiliki jumlah data yang paling sedikit, tapi juga memiliki dinamika yang paling sederhana karena masih awal. Model utama berhasil mencapai RMSSE di bawah 1 untuk jumlah kasus positif aktif (i), jumlah kasus sembuh (r), dan jumlah kasus meninggal (d) Kab. Sampang serta jumlah kasus sembuh (r) Kab. Tuban. Sebagian besar model *baseline* memperoleh hasil serupa untuk Kab. Sampang sedangkan model SIRD (*Susceptible-Infected-Recovered-Dead*) tidak pernah mencapai nilai RMSSE di bawah 1. Hasil evaluasi model *Autoregressive Integrated Moving Average* (ARIMA)-SIRD mendapat peringkat terburuk untuk semua kabko (kabupaten/kota) dan label kecuali untuk jumlah kasus sembuh (r) Kota Madiun dan jumlah kasus meninggal (d) Kab. Sampang dimana model tersebut mendapat nilai sempurna; sepertinya karena tren yang cukup stabil. Semua model memperoleh hasil sangat buruk untuk jumlah kasus meninggal (d) Kota Madiun. Hal ini terjadi karena kenaikan yang tiba-tiba di akhir itu sangat sulit diprediksi. Selain itu, nilai RMSSE yang sangat besar juga disebabkan oleh pembagi RMSSE yang bernilai sangat kecil ketika tidak ada perubahan jumlah kasus dalam waktu lama.

Evaluasi dilanjutkan dengan uji statistik untuk menguji signifikansi perbedaan tiap model menggunakan rata-rata rank Hasil uji Adjusted Friedman (FF) menghasilkan $p > 0.05$ untuk semua label dan baru mendapat $p < 0.05$ ketika rank semua label digabungkan menjadi satu (Tabel 4). Perbedaan rank tiap model *baseline* dan perbandingan dengan model utama yang melebihi nilai *critical difference* (CD) Bonferroni-Dunn hanya dua model, yaitu (1) model *baseline 4* dan (2) model ARIMA-SIRD untuk label gabungan (Gambar 5). Keduanya memiliki rata-rata rank yang lebih buruk dari model utama. Ini berarti model utama lebih baik dari model *baseline 4* (tanpa clustering) dan ARIMA-SIRD secara signifikan. Dengan demikian, model utama hanya lebih baik secara signifikan dari model *baseline 4* dan ARIMA-SIRD untuk kelompok 90 hari. Karena model utama tidak berbeda secara signifikan dengan model *baseline* selain *baseline 4*, model utama tidak lebih baik daripada *Long-Short Term Memory* (LSTM) biasa (*baseline 1*), bagian-bagian model utama selain LSTM kurang berguna, dan *joint learning* dengan dataset semua daerah tanpa clustering justru memperburuk model.

TABEL 4
HASIL UJI STATISTIK RATA-RATA RANK KELOMPOK 90 HARI

Stat	I	R	D	Gabungan
X2F	9.6000	12.5333	9.6667	24.2444
FF	1.2857	1.9315	1.2985	3.7166
P	0.2969	0.1016	0.2908	0.0009
CD				3.0455

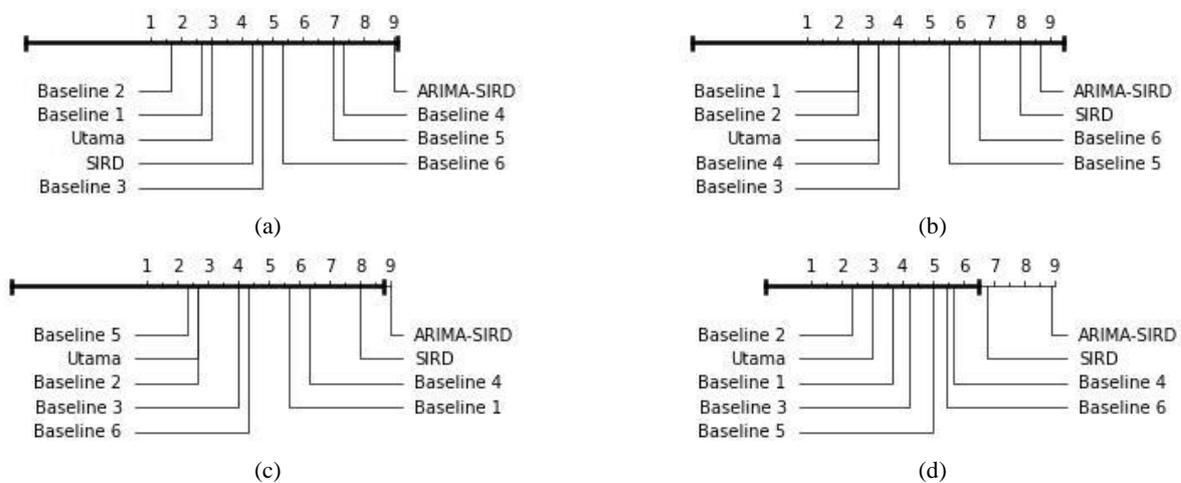


Gambar 5. Diagram Bonferroni-Dunn untuk label gabungan kelompok 90 hari

2) Kelompok 180 Hari

Kelompok kedua ini memiliki jumlah data menengah (dari total data sekitar satu tahun) dengan dinamika yang sudah cukup rumit (ada tren naik dan turun). Model utama tidak berhasil mencapai nilai RMSSE di bawah 1 untuk label manapun. Akan tetapi, model *baseline* 1 (LSTM), 2 (tanpa *representation layer*), dan 4 (tanpa *clustering*) untuk jumlah kasus sembuh (r) Kab. Sampang dan model *baseline* 5 untuk jumlah kasus meninggal (d) Kab. Tuban berhasil mencapai nilai RMSSE di bawah 1, sedangkan model utama mendapat nilai mendekati 1. Grafik untuk label tersebut memiliki tren yang cukup stabil dan sederhana. Model ARIMA-SIRD selalu mendapat nilai terburuk kecuali untuk jumlah kasus sembuh Kab. Sampang.

Evaluasi dilanjutkan dengan uji statistik untuk menguji signifikansi perbedaan tiap model menggunakan rata-rata *rank*. Hasil uji *Adjusted Friedman* (FF) menghasilkan $p < 0.05$ untuk semua label dan gabungan (Tabel 5). Namun, perbedaan *rank* tiap model *baseline* dan pembandingan dengan model utama yang melebihi nilai CD Bonferroni-Dunn hanya 3 model, yaitu (1) model ARIMA-SIRD untuk jumlah kasus meninggal (d) serta (2) model ARIMA-SIRD dan (3) model SIRD untuk label gabungan (Gambar 6). Ketiganya memiliki rata-rata *rank* yang lebih buruk dari model utama. Ini berarti model utama lebih baik dari model ARIMA-SIRD dan SIRD secara signifikan. Dengan demikian, model utama hanya lebih baik secara signifikan dari model ARIMA-SIRD dan SIRD untuk kelompok 180 hari. Karena model utama tidak berbeda secara signifikan dengan model *baseline* manapun, model utama tidak lebih baik daripada LSTM biasa (*baseline* 1) dan bagian-bagian model utama selain LSTM kurang berguna.



Gambar 6. Diagram Bonferroni-Dunn untuk (a) jumlah kasus positif aktif, (b) jumlah kasus sembuh, (c) jumlah kasus meninggal, dan (d) label gabungan untuk kelompok 180 hari

TABEL 5
HASIL UJI STATISTIK RATA-RATA *RANK* UNTUK KELOMPOK 180 HARI

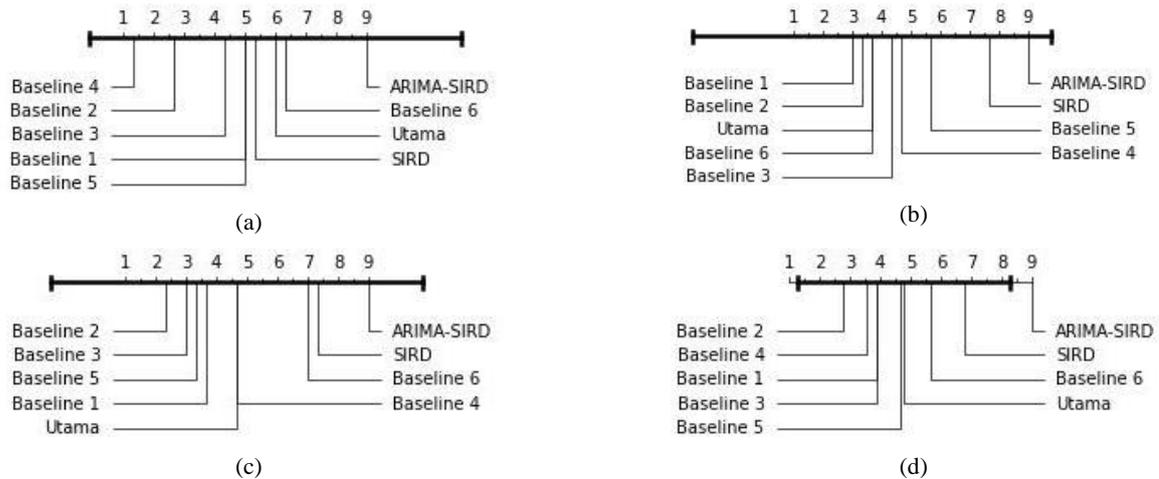
Stat	I	R	D	Gabungan
X2F	18.6667	17.2444	18.6667	38.9037
FF	7.0000	5.1053	7.0000	9.4038
P	0.0005	0.0028	0.0005	0.0000
CD	6.0910	6.0910	6.0910	3.5167

3) Kelompok 366 Hari

Kelompok ketiga ini menggunakan semua data dan memiliki dinamika yang rumit. Model utama hanya berhasil mencapai nilai RMSSE di bawah 1 untuk jumlah kasus sembuh (r) Kab. Sampang. Model ARIMA-SIRD selalu mendapat nilai terburuk. Model-model secara umum mendapat nilai terbaik untuk jumlah kasus sembuh (r) Kab. Tuban dan Kab. Sampang. Kedua grafik tersebut memiliki tren yang cukup stabil dan mirip.

Evaluasi dilanjutkan dengan uji statistik untuk menguji signifikansi perbedaan tiap model menggunakan rata-rata *rank*. Hasil uji *Adjusted Friedman* (FF) menghasilkan $p < 0.05$ untuk semua label dan juga ketika semua label digabungkan menjadi satu (Tabel 6). Namun, perbedaan *rank* tiap model *baseline* dan pembandingan dengan model utama yang melebihi nilai CD Bonferroni-Dunn hanya model ARIMA-SIRD untuk label gabungan (Gambar 7). Model tersebut memiliki rata-rata *rank* yang lebih buruk dari model utama. Ini berarti model utama lebih baik dari model ARIMA-SIRD secara signifikan. Dengan demikian, model utama hanya lebih baik secara signifikan dari model ARIMA-SIRD untuk kelompok 366 hari. Karena model

utama tidak berbeda secara signifikan dengan model *baseline* manapun, model utama tidak lebih baik daripada LSTM biasa (*baseline 1*) dan bagian-bagian model utama selain LSTM kurang berguna.



Gambar 7. Diagram Bonferroni-Dunn untuk (a) jumlah kasus positif aktif, (b) jumlah kasus sembuh, (c) jumlah kasus meninggal, dan (d) label gabungan untuk kelompok 366 hari

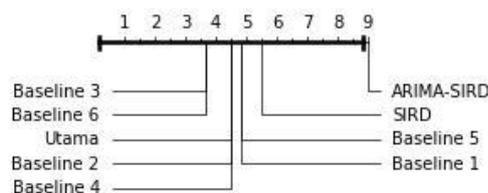
TABEL 6
HASIL UJI STATISTIK RATA-RATA RANK UNTUK KELOMPOK 366 HARI

Stat	I	R	D	Gabungan
X2F	15.2889	13.7778	16.5333	35.1111
FF	3.5102	2.6957	4.4286	7.6145
P	0.0156	0.0435	0.0055	0.0000
CD	6.0910	6.0910	6.0910	3.5167

4) Kelompok Sebelum Vaksinasi

Kelompok keempat ini adalah kelompok dengan pembatasan jumlah data hingga sebelum ada vaksinasi dosis penuh, yaitu sampai 21 Januari 2020. Kelompok ini memiliki jumlah data cukup banyak (sekitar 10 bulan) dengan dinamika yang rumit. Tidak ada model yang berhasil mencapai nilai RMSSE di bawah 1. Model-model secara umum mendapat nilai terbaik untuk jumlah kasus sembuh (r) dan jumlah kasus meninggal (d) Kota Pasuruan. Model ARIMA-SIRD selalu mendapat nilai terburuk. Semua model memperoleh hasil sangat buruk untuk jumlah kasus meninggal (d) Kab. Sampang. Nilai RMSSE yang sangat besar untuk jumlah kasus meninggal tersebut mungkin disebabkan oleh pembagi RMSSE yang bernilai sangat kecil ketika tidak ada perubahan jumlah kasus dalam waktu lama (grafik rata/flat). Meskipun begitu, nilai yang sangat buruk ini tidak terjadi pada jumlah kasus meninggal (d) Kab. Sampang pada kelompok 90 hari, 180 hari, dan 366 hari.

Evaluasi dilanjutkan dengan uji statistik untuk menguji signifikansi perbedaan tiap model menggunakan rata-rata *rank*. Hasil uji *Adjusted Friedman* (FF) menghasilkan $p > 0.05$ untuk semua label dan hanya mendapat $p < 0.05$ ketika semua label digabungkan (Tabel 7). Perbedaan *rank* tiap model *baseline* dan pembandingan dengan model utama yang melebihi nilai CD Bonferroni-Dunn hanya model ARIMA-SIRD untuk label gabungan (Gambar 8). Model tersebut memiliki rata-rata *rank* yang lebih buruk dari model utama. Ini berarti model utama lebih baik dari model ARIMA-SIRD secara signifikan. Dengan demikian, model utama hanya lebih baik secara signifikan dari model ARIMA-SIRD untuk kelompok sebelum vaksinasi. Karena model utama tidak berbeda secara signifikan dengan model *baseline* manapun, model utama tidak lebih baik daripada LSTM biasa (*baseline 1*) dan bagian-bagian model utama selain LSTM kurang berguna.



Gambar 8. Diagram Bonferroni-Dunn untuk label gabungan kelompok sebelum vaksinasi

TABEL 7
HASIL UJI STATISTIK RATA-RATA RANK UNTUK KELOMPOK SEBELUM VAKSINASI

Stat	I	R	D	Gabungan
X2F	10.2667	11.6000	9.4667	16.4889
FF	1.7907	2.6364	1.4490	2.6164
P	0.2138	0.0960	0.3061	0.0210
CD				4.3070

5) Pembahasan

Evaluasi masing-masing kelompok telah dibahas. Bagian ini berisi kesimpulan dari keempat kelompok. Untuk setiap jenis model, sekitar setengah dari model yang dibuat memiliki *best epoch* (jumlah *training step* dimana model mendapat *validation loss* terkecil) yang kecil. Sebagian besar dari model-model itu berasal dari kelompok data 90 hari yang memiliki jumlah data paling sedikit tapi juga pola data paling sederhana, sehingga ini mungkin disebabkan oleh data yang terlalu sedikit atau pola yang terlalu mudah. Meskipun begitu, jumlah model yang seperti itu juga tidak sedikit untuk kelompok lainnya, sehingga mungkin juga model tidak cocok atau pola data terlalu sulit.

Attribution pada tiap blok/*layer* model menggunakan Captum [59] dengan metode *Saliency* menunjukkan bahwa bagian model yang penting untuk model adalah bagian *head*, yaitu LSTM. Meskipun begitu, bobot bagian *shared* juga cukup besar, sehingga seharusnya model mempelajari banyak dari *joint learning*.

Attribution pada *input feature* menggunakan Captum [59] dengan metode *Saliency* menunjukkan bahwa jumlah tes harian (*delta_test*) dan jumlah orang yang divaksinasi penuh (*vac_full_s*) tidak penting bagi model. Dari hasil *attribution*, ditemukan bahwa tanggal penting memiliki bobot yang tidak kalah dari variabel endogen (*beta*, *gamma*, *delta*) dan beberapa tanggal justru lebih besar dari variabel endogen, yang berarti dianggap sangat penting oleh model.

Salah satu sasaran model adalah untuk mencapai nilai RMSSE dibawah 1. Rekap hasil evaluasi tiap model untuk tiap kelompok dan tiap label yang berhasil mencapai nilai RMSSE di bawah 1 dipaparkan secara berurutan pada Tabel 8 dan Tabel 9. Model yang paling banyak mencapai nilai evaluasi di bawah 1 adalah model *baseline 5*, diikuti oleh model utama dan model *baseline 2* dan *3*. Model lebih banyak mencapai RMSSE di bawah 1 untuk kelompok data 90 hari dan tidak pernah mencapainya untuk kelompok data sebelum vaksinasi. Sedangkan dari sisi label, model lebih banyak mencapai RMSSE di bawah 1 untuk jumlah kasus sembuh. Meskipun demikian, jumlah yang berhasil sangatlah sedikit dari keseluruhan label yang diramalkan. Tidak ada model yang berhasil mencapai sasaran ini untuk lebih dari setengah label yang diramalkan. Nilai RMSSE semakin besar seiring bertambahnya data, kecuali untuk kelompok data sebelum vaksinasi yang menghasilkan nilai RMSSE paling besar meskipun datanya kurang dari 366 hari. Ini bisa disebabkan oleh pola data yang semakin rumit seiring berjalannya pandemi dan diterapkannya kebijakan-kebijakan untuk mengatasi pandemi, tapi ini tidak dapat menjelaskan besarnya nilai RMSSE pada kelompok data sebelum vaksinasi.

TABEL 8
PERSENTASE PENCAPAIAN RMSSE DI BAWAH 1 UNTUK TIAP KELOMPOK

Model	Kelompok Pembatasan Data				Total
	90 Hari	180 Hari	366 Hari	Sebelum Vaksinasi	
Utama	33.33%	0.00%	11.11%	0.00%	13.89%
Baseline 1	16.67%	11.11%	11.11%	0.00%	11.11%
Baseline 2	25.00%	11.11%	11.11%	0.00%	13.89%
Baseline 3	41.67%	0.00%	0.00%	0.00%	13.89%
Baseline 4	16.67%	0.00%	11.11%	0.00%	8.33%
Baseline 5	25.00%	22.22%	11.11%	0.00%	16.67%
Baseline 6	25.00%	0.00%	0.00%	0.00%	8.33%
ARIMA-SIRD	8.33%	0.00%	0.00%	0.00%	2.78%
SIRD	0.00%	0.00%	0.00%	0.00%	0.00%
Total	21.30%	4.94%	6.17%	0.00%	9.88%

Semua model memperoleh nilai yang sangat buruk untuk jumlah kasus meninggal (*d*) Kota Madiun pada kelompok 90 hari dan Kab. Sampang pada kelompok sebelum vaksinasi. Pada Kota Madiun kelompok 90 hari, grafik sangat rata dan ada lonjakan di ujung yang menjadi bagian dari *test set*. Model hanya dapat mempelajari data yang rata sedangkan *test set* memiliki lonjakan yang sangat mendadak, sehingga memang sangat sulit diprediksi. Selain itu, nilai RMSSE yang sangat besar juga disebabkan oleh pembagi RMSSE, yaitu *Mean Squared Error* (MSE) peramalan naif satu langkah, yang bernilai sangat kecil ketika tidak ada perubahan jumlah kasus dalam waktu lama (grafik rata/*flat*). Pada Kab. Sampang kelompok sebelum vaksinasi, tren grafik sudah terlihat secara kasat mata, tapi masih memiliki banyak bagian yang rata. Meskipun begitu, jumlah kasus meninggal (*d*) Kab. Sampang pada kelompok 90 hari, 180 hari, dan 366 hari tidak memiliki masalah ini.

Untuk penelitian selanjutnya, mungkin sebaiknya data *training* dihaluskan terlebih dahulu dan digunakan alternatif RMSSE yang tidak terlalu terdampak oleh data yang rata.

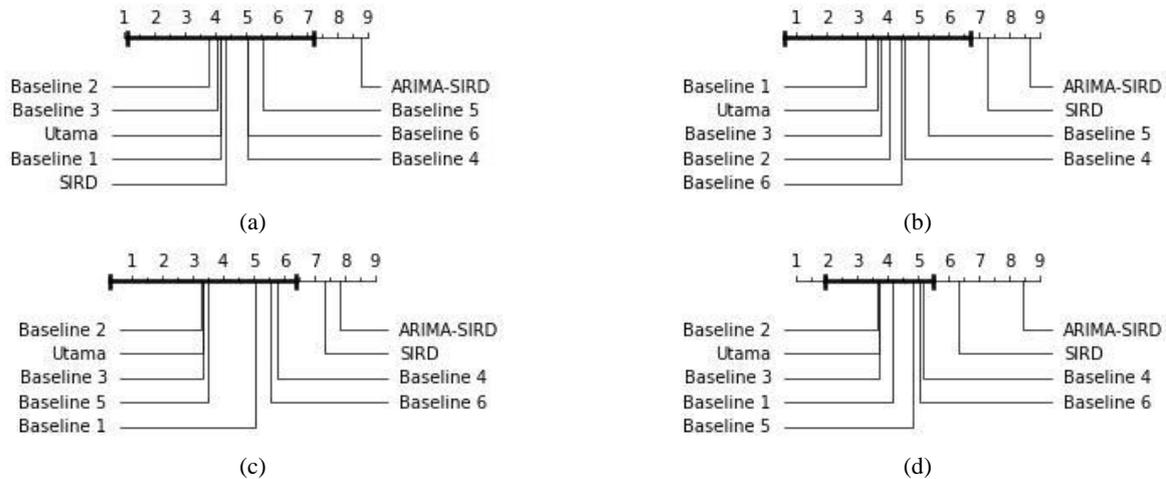
TABEL 9
PERSENTASE PENCAPAIAN RMSSE DI BAWAH 1 UNTUK TIAP LABEL

Model	Jumlah Kasus			Total
	Positif Aktif (I)	Sembuh (R)	Meninggal (D)	
Utama	8.33%	25.00%	8.33%	13.89%
Baseline 1	0.00%	25.00%	8.33%	11.11%
Baseline 2	8.33%	25.00%	8.33%	13.89%
Baseline 3	8.33%	25.00%	8.33%	13.89%
Baseline 4	8.33%	16.67%	0.00%	8.33%
Baseline 5	8.33%	25.00%	16.67%	16.67%
Baseline 6	8.33%	8.33%	8.33%	8.33%
ARIMA-SIRD	0.00%	0.00%	8.33%	2.78%
SIRD	0.00%	0.00%	0.00%	0.00%
Total	5.56%	16.67%	7.41%	9.88%

Evaluasi dilanjutkan dengan uji statistik untuk menguji signifikansi perbedaan tiap model menggunakan rata-rata *rank*. Sebuah model dinilai berbeda secara signifikan dari model utama jika nilai *p* statistik *Adjusted Friedman* di bawah 0.05 dan perbedaan rata-rata *rank*-nya di lebih besar daripada CD Bonferroni-Dunn. Hanya sedikit model yang dinilai berbeda secara signifikan dan semuanya lebih buruk dari model utama. Dari evaluasi per kelompok, ditarik kesimpulan bahwa:

1. Model utama tidak lebih baik daripada model LSTM *encoder-decoder* biasa
2. Model utama lebih baik daripada model ARIMA-SIRD
3. Model utama mungkin lebih baik dari model SIRD
4. *Joint learning* dengan model semua daerah sekaligus tanpa *clustering* dapat memperburuk model

Ketika semua kelompok digabungkan dan dievaluasi, model ARIMA-SIRD dinilai lebih buruk secara signifikan untuk tiap label dan label gabungan, sedangkan model SIRD dinilai lebih buruk secara signifikan untuk label gabungan dan semua label kecuali jumlah kasus positif aktif (i). Model *baseline 4* tidak dinilai berbeda signifikan dalam uji statistik kelompok gabungan. Hasil uji statistik dan diagram Bonferroni-Dunn secara berurutan dipaparkan pada Tabel 10 dan Gambar 9.



Gambar 9. Diagram Bonferroni-Dunn untuk (a) jumlah kasus positif aktif, (b) jumlah kasus sembuh, (c) jumlah kasus meninggal, dan (d) label gabungan untuk kelompok gabungan

TABEL 10
HASIL UJI STATISTIK RATA-RATA *RANK* UNTUK KELOMPOK GABUNGAN

Stat	I	R	D	Gabungan
X2F	29.8444	36.7542	40.4000	91.6815
FF	4.9624	6.8240	7.9928	16.3451
P	0.0000	0.0000	0.0000	0.0000
CD	3.0455	3.0455	3.0455	1.7583

IV. SIMPULAN

Joint learning yang dilakukan dalam penelitian ini tidak berhasil meningkatkan kinerja model dalam penelitian ini. Kinerja model yang diajukan lebih baik daripada model SIRD dan *Autoregressive Integrated Moving Average* (ARIMA)-SIRD tapi tidak lebih baik daripada model *Long-Short Term Memory* (LSTM) *encoder-decoder* biasa dan jarang mencapai nilai RMSSE di bawah 1. Selain itu, ditemukan bahwa melakukan *joint learning* dengan data semua daerah justru memperburuk model. Ditemukan juga bahwa RMSSE terlalu sensitif untuk *time series* yang jarang berubah karena menggunakan pembagian dengan *Mean Squared Error* (MSE) peramalan naif satu langkah dalam sampel.

Penelitian selanjutnya dapat dilakukan untuk mencari alternatif RMSSE dan *Mean Absolute Scaled Error* (MASE), yang memiliki prinsip yang sama, yang tidak rentan terhadap *time series* yang jarang berubah. Selain itu, *joint learning* mungkin tidak berhasil karena perlakuan yang sama untuk semua model *baseline*, yaitu transformasi ke variabel SIRD dan *clustering* menggunakan *DTW* (*Dynamic Time Warping*) *Barycenter Averaging* (DBA) pada variabel SIRD diskrit. Penelitian selanjutnya dapat dilakukan untuk mencari dan membandingkan alternatif kedua metode tersebut.

DAFTAR PUSTAKA

- [1] C. Sohrabi *et al.*, "World Health Organization declares global emergency: A review of the 2019 novel coronavirus (COVID-19)," *Int. J. Surg.*, vol. 76, no. 4, pp. 71–76, 2020, doi: 10.1016/j.ijsu.2020.02.034.
- [2] T. Singhal, "A Review of Coronavirus Disease-2019 (COVID-19)," *Indian J. Pediatr.*, vol. 87, no. 3, pp. 281–286, 2020, doi: 10.1007/s12098-020-03263-6.
- [3] Y. C. Wu, C. S. Chen, dan Y. J. Chan, "The outbreak of COVID-19: An overview," *J. Chinese Med. Assoc.*, vol. 83, no. 3, pp. 217–220, 2020, doi: 10.1097/JCMA.0000000000000270.
- [4] M. Park, A. R. Cook, J. T. Lim, Y. Sun, dan B. L. Dickens, "A Systematic Review of COVID-19 Epidemiology Based on Current Evidence," *J. Clin. Med.*, vol. 9, no. 4, pp. 967, 2020, doi: 10.3390/jcm9040967.
- [5] H. Harapan *et al.*, "Coronavirus disease 2019 (COVID-19): A literature review," *J. Infect. Public Health*, vol. 13, no. 5, pp. 667–673, 2020, doi: 10.1016/j.jiph.2020.03.019.
- [6] V. F. Mukaromah. (2020) Lewati Singapura, Kasus Covid-19 di Indonesia Kini Terbanyak di ASEAN. [Online]. Tersedia: <https://www.kompas.com/tren/read/2020/06/17/175103265/lewati-singapura-kasus-covid-19-di-indonesia-kini-terbanyak-di-asean?page=all>
- [7] Komite Penanganan COVID-19 dan Pemulihan Ekonomi Nasional. (2021) Peta Sebaran COVID-19 | Covid19.go.id. [Online]. Tersedia: <https://covid19.go.id/peta-sebaran-covid19>
- [8] Widyawati. (2021) Pelaksanaan Vaksinasi COVID-19 di Indonesia Membutuhkan Waktu 15 Bulan. [Online]. Tersedia: <https://sehatnegeriku.kemkes.go.id/baca/rilis-media/20210103/2536122/pelaksanaan-vaksinasi-covid-19-indonesia-membutuhkan-waktu-15-bulan/>
- [9] Komite Penanganan COVID-19 dan Pemulihan Ekonomi Nasional. (2021) Vaksin COVID-19 GRATIS, Tanpa syarat! [Online]. Tersedia: <https://covid19.go.id/edukasi/masyarakat-umum/vaksin-covid-19-gratis-tanpa-syarat>
- [10] H. P. Sari. (2021) Tak Jadi April, Vaksinasi Covid-19 Tahap Ketiga Dimulai Juni 2021. [Online]. Tersedia: <https://nasional.kompas.com/read/2021/04/01/08062301/tak-jadi-april-vaksinasi-covid-19-tahap-ketiga-dimulai-juni-2021>
- [11] J. Ashton, "COVID-19 and the anti-vaxxers," *J. R. Soc. Med.*, vol. 114, no. 1, pp. 42–43, 2021, doi: 10.1177/0141076820986065.
- [12] O. J. Wouters *et al.*, "Challenges in ensuring global access to COVID-19 vaccines: production, affordability, allocation, and deployment," *Lancet*, vol. 397, no. 10278, pp. 1023–1034, 2021, doi: 10.1016/S0140-6736(21)00306-8.
- [13] R. Jhaveri, "The COVID-19 mRNA Vaccines and the Pandemic: Do They Represent the Beginning of the End or the End of the Beginning?," *Clin. Ther.*, vol. 43, no. 3, pp. 549–556, 2021, doi: 10.1016/j.clinthera.2021.01.014.
- [14] S. Sirekbasan, "The biggest challenge in the fight against the COVID-19 pandemic: Vaccine hesitancy," *Ukr. J. Nephrol. Dial.*, vol. 69, no. 1, pp. 3–5, 2021, doi: 10.31450/ukrjnd.1(69).2021.01.
- [15] V. Thakur *et al.*, "Waves and variants of SARS-CoV-2: understanding the causes and effect of the COVID-19 catastrophe," *Infection*, vol. 49, no. 6, 2021, doi: 10.1007/s15010-021-01734-2.
- [16] A. Fontanet, B. Autran, B. Lina, M. P. Kieny, S. S. A. Karim, dan D. Sridhar, "SARS-CoV-2 variants and ending the COVID-19 pandemic," *Lancet*, vol. 397, no. 10278, pp. 952–954, 2021, doi: 10.1016/S0140-6736(21)00370-6.
- [17] R. Djalante *et al.*, "Review and analysis of current responses to COVID-19 in Indonesia: Period of January to March 2020," *Prog. Disaster Sci.*, vol. 6, pp. 100091, 2020, doi: 10.1016/j.pdisas.2020.100091.
- [18] I. Rudan, "A cascade of causes that led to the COVID-19 tragedy in Italy and in other European union countries," *J. Glob. Health*, vol. 10, no. 1, 2020, doi: 10.7189/JOGH.10.010335.
- [19] F. Petropoulos dan S. Makridakis, "Forecasting the novel coronavirus COVID-19," *PLoS One*, vol. 15, no. 3, pp. 1–8, 2020, doi: 10.1371/journal.pone.0231236.
- [20] J. Devaraj *et al.*, "Forecasting of COVID-19 cases using deep learning models: Is it reliable and practically significant?," *Results Phys.*, vol. 21, no. 2, pp. 1–25, 2021, doi: 10.1016/j.rinp.2021.103817.
- [21] I. Goodfellow, Y. Bengio, dan A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Tersedia: <https://www.deeplearningbook.org/>
- [22] A. H. Elsheikh *et al.*, "Deep learning-based forecasting model for COVID-19 outbreak in Saudi Arabia," *Process Saf. Environ. Prot.*, vol. 149, pp. 223–233, 2021, doi: 10.1016/j.psep.2020.10.048.
- [23] S. J. Pan dan Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2010, doi: 10.1109/TKDE.2009.191.
- [24] A. R. Ajiboye, R. Abdullah-Arshah, H. Qin, dan H. Isah-Kebbe, "Evaluating the Effect of Dataset Size on Predictive Model using Supervised Learning Technique," *Int. J. Softw. Eng. Comput. Sci.*, vol. 1, pp. 75–84, 2015, doi: 10.15282/ijsecs.1.2015.6.0006.
- [25] Y. Li *et al.*, "ALeRT-COVID: Attentive Lockdown-awaRe Transfer Learning for Predicting COVID-19 Pandemics in Different Countries," *J. Healthc. Informatics Res.*, pp. 98–113, 2021, doi: 10.1007/s41666-020-00088-y.
- [26] Y. Gautam, "Transfer Learning for COVID-19 cases and deaths forecast using LSTM network," *ISA Trans.*, 2021, doi: 10.1016/j.isatra.2020.12.057.
- [27] Z. Zhang, X. Huang, Q. Huang, X. Zhang, dan Y. Li, "Joint learning of neural networks via iterative reweighted least squares," in *Proceedings*

- of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2019, pp. 18–26.
- [28] S. Chatterjee, A. Sarkar, S. Chatterjee, M. Karmakar, dan R. Paul, “Studying the progress of COVID-19 outbreak in India using SIRD model,” *Indian J. Phys.*, vol. 95, no. 6, pp. 1941–1957, 2021, doi: 10.1007/s12648-020-01766-8.
- [29] P. Nadler, R. Arcucci, dan Y. Guo, “A Neural SIR Model for Global Forecasting,” in *Proceedings of Machine Learning Research*, 2020, vol. 136, pp. 254–266. [Online]. Tersedia: <http://proceedings.mlr.press/v136/nadler20a.html>
- [30] F. Petitjean, A. Ketterlin, dan P. Gançarski, “A global averaging method for dynamic time warping, with applications to clustering,” *Pattern Recognit.*, vol. 44, no. 3, pp. 678–693, 2011, doi: 10.1016/j.patcog.2010.09.013.
- [31] R. J. Hyndman dan G. Athanasopoulos, *Forecasting: principles and practice*, 2 ed. Melbourne, Australia: OTexts, 2018. [Online]. Tersedia: <https://otexts.com/fpp2/>
- [32] Pemerintah Provinsi Jawa Timur. (2021) JATIM TANGGAP COVID-19. [Online]. Tersedia: <http://infocovid19.jatimprov.go.id/>
- [33] Badan Pusat Statistik Provinsi Jawa Timur. (2020) Jumlah Penduduk Provinsi Jawa Timur (Jiwa), 2018–2020. [Online]. Tersedia: <https://jatim.bps.go.id/indicator/12/375/1/jumlah-penduduk-provinsi-jawa-timur.html>
- [34] J. Hasell *et al.*, “A cross-country database of COVID-19 testing,” *Sci. Data*, vol. 7, no. 1, pp. 1–7, 2020, doi: 10.1038/s41597-020-00688-8.
- [35] E. Mathieu *et al.*, “A global database of COVID-19 vaccinations,” *Nat. Hum. Behav.*, vol. 5, no. 7, pp. 947–953, 2021, doi: 10.1038/s41562-021-01122-8.
- [36] H. Lau *et al.*, “The positive impact of lockdown in Wuhan on containing the COVID-19 outbreak in China,” *J. Travel Med.*, vol. 27, no. 3, pp. 1–7, 2020, doi: 10.1093/jtm/taaa037.
- [37] E. Mahase, “Covid-19: How does local lockdown work, and is it effective?,” *BMJ*, vol. 370, no. 8253, pp. 1–2, 2020, doi: 10.1136/bmj.m2679.
- [38] N. Soares *et al.*, “SIRNet: Understanding Social Distancing Measures with Hybrid Neural Network Model for COVID-19 Infectious Spread,” *arXiv*, 2020, [Online]. Tersedia: <http://arxiv.org/abs/2004.10376>
- [39] M. Yechezkel, A. Weiss, I. Rejwan, E. Shahmoon, S. Ben-Gal, dan D. Yamin, “Human mobility and poverty as key drivers of COVID-19 transmission and control,” *BMC Public Health*, pp. 1–13, 2021, doi: 10.1101/2020.06.04.20112417.
- [40] Dinas Pendidikan Provinsi Jawa Timur. (2020) Kalender pendidikan. [Online]. Tersedia: <https://dindik.jatimprov.go.id/pages/35/kalender-pendidikan>
- [41] H. Akoglu, “User’s guide to correlation coefficients,” *Turkish J. Emerg. Med.*, vol. 18, no. 3, pp. 91–93, 2018, doi: 10.1016/j.tjem.2018.08.001.
- [42] A. Paszke *et al.*, “PyTorch: An imperative style, high-performance deep learning library,” *Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, [Online]. Tersedia: <https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>
- [43] M. Goic, M. S. Bozanic-Leal, M. Badal, dan L. J. Basso, “COVID-19: Short-term forecast of ICU beds in times of crisis,” *PLoS One*, vol. 16, no. 1, pp. 1–24, 2021, doi: 10.1371/journal.pone.0245272.
- [44] P. Wang, X. Zheng, G. Ai, D. Liu, dan B. Zhu, “Time series prediction for the epidemic trends of COVID-19 using the improved LSTM deep learning method: Case studies in Russia, Peru and Iran,” *Chaos, Solitons and Fractals*, vol. 140, pp. 1–8, 2020, doi: 10.1016/j.chaos.2020.110214.
- [45] B. Epstein, R. Meir, dan T. Michaeli, “Joint auto-encoders: A flexible multi-task learning framework,” *arXiv*, pp. 1–11, 2017, doi: 10.48550/arXiv.1705.10494.
- [46] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, dan D. J. Inman, “1D convolutional neural networks and applications: A survey,” *Mech. Syst. Signal Process.*, vol. 151, pp. 107398, 2021, doi: 10.1016/j.ymsp.2020.107398.
- [47] M. G. Ragab *et al.*, “A novel one-dimensional cnn with exponential adaptive gradients for air pollution index prediction,” *Sustain.*, vol. 12, no. 23, pp. 1–22, 2020, doi: 10.3390/su122310090.
- [48] A. A. M. Al-Saffar, H. Tao, dan M. A. Talab, “Review of deep convolution neural network in image classification,” *2017 Int. Conf. Radar, Antenna, Microwave, Electron. Telecommun.*, pp. 26–31, 2017, doi: 10.1109/ICRAMET.2017.8253139.
- [49] G. Lai, W. C. Chang, Y. Yang, dan H. Liu, “Modeling long- and short-term temporal patterns with deep neural networks,” in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 2018, pp. 95–104. doi: 10.1145/3209978.3210006.
- [50] K. He, X. Zhang, S. Ren, dan J. Sun, “Deep Residual Learning for Image Recognition Kaiming,” *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, pp. 770–778, 2016, doi: 10.1002/chin.200650130.
- [51] H. Hapke dan C. Nelson, *Building Machine Learning Pipelines*. Sebastopol, CA: O’Reilly Media, Inc., 2020.
- [52] T. Hastie, R. Tibshirani, dan J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, 2 ed. Springer-Verlag, New York: Springer Science & Business Media, 2009.
- [53] R. Tavenard *et al.*, “Tsllearn, a Machine Learning Toolkit for Time Series Data,” *J. Mach. Learn. Res.*, vol. 21, no. 118, pp. 1–6, 2020.
- [54] T. M. Kodinariya dan P. R. Makwana, “Review on determining number of Cluster in K-Means Clustering,” *Int. J. Adv. Res. Comput. Sci. Manag. Stud.*, vol. 1, no. 6, pp. 90–95, 2013.
- [55] S. Makridakis *et al.*, “The M5 uncertainty competition: Results, findings and conclusions,” *Int. J. Forecast.*, pp. 1–21, 2021, doi: 10.1016/j.ijforecast.2021.10.009.
- [56] R. J. Hyndman dan A. B. Koehler, “Another look at measures of forecast accuracy,” *Int. J. Forecast.*, vol. 22, no. 4, pp. 679–688, 2006, doi: 10.1016/j.ijforecast.2006.03.001.
- [57] T. Akiba, S. Sano, T. Yanase, T. Ohta, dan M. Koyama, “Optuna: A next-generation hyperparameter optimization framework,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2623–2631. doi: 10.1145/3292500.3330701.
- [58] J. Demšar, “Statistical comparisons of classifiers over multiple data sets,” *J. Mach. Learn. Res.*, vol. 7, no. 1, pp. 1–30, 2006.
- [59] N. Kokhlikyan *et al.*, “Captum: A unified and generic model interpretability library for PyTorch,” *arXiv*, pp. 1–11, 2020, [Online]. Tersedia: <http://arxiv.org/abs/2009.07896>