

## BAB II

### LANDASAN TEORI

#### 2.1 Penelitian Terdahulu

Penelitian terdahulu merupakan sub bab landasan untuk dilakukannya perbandingan sebagai acuan dengan penelitian yang dilakukan peneliti sekarang. Acuan tersebut berguna untuk mencegah adanya plagiasi dan anggapan terhadap kesamaan materi/topik yang diteliti. Kajian pendahuluan dari penelitian terdahulu dan yang sekarang ini akan dirangkum dalam bentuk tabel sedemikian rupa sebagai berikut:

Tabel 2.1 Tabel Perbandingan Penelitian Terdahulu

No	Peneliti	Judul Artikel, Jurnal dan Tahun	Metode Penelitian	Hasil Penelitian
1	S. A. Sanjaya dan M. I. Fianty	<i>Time-based Geospatial Analysis of Night-Time Light Data and Citizen Movement Restriction During Covid-19 Period</i> , G-Tech J. Teknol. Terap., vol. 7, no. 2, 2023 [14]	<i>Time-based Geospatial Analysis</i>	Intensitas cahaya menurun antara 1% hingga 16% saat COVID-19 memasuki Indonesia, meskipun terjadi penurunan mobilitas masyarakat yang ekstrem, hampir mencapai -70% berdasarkan data Google. Namun, pembatasan kegiatan masyarakat tidak secara langsung berpengaruh besar terhadap perubahan intensitas cahaya, yang juga bisa dipengaruhi oleh faktor cuaca seperti awan, hujan, kabut, atau banjir yang dapat mendistorsi citra satelit.
2	A. Nilsen	Perbandingan Model RNN, Model LSTM, dan Model GRU dalam Memprediksi Harga Saham-Saham LQ45, J. Stat. dan Apl., vol. 6, no. 1, 2022 [20]	<i>Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU)</i>	Model GRU lebih unggul dalam memprediksi harga saham-saham yang terdaftar di indeks LQ45, dengan nilai evaluasi (RMSE, MSE, dan MAE) yang paling rendah dibandingkan dengan model RNN dan LSTM. Saham TKIM (PT Pabrik Kertas Tjiwi Kimia Tbk.) tercatat memiliki hasil tertinggi pada model RNN yang tidak menggunakan <i>gate</i> , sementara PT Gudang Garam Tbk (GGRM) memiliki galat tertinggi untuk model LSTM dan GRU yang menggunakan <i>gate</i> .

No	Peneliti	Judul Artikel, Jurnal dan Tahun	Metode Penelitian	Hasil Penelitian
				Perbandingan antara ketiga model menunjukkan bahwa GRU memiliki akurasi yang lebih tinggi sebesar 52% dari model RNN dan 47.53% dari model LSTM, menegaskan keefektifan model GRU dalam prediksi harga saham di indeks LQ45.
3	W. Huacani, N. P. Meza, F. Aguirre, D. D. Sanchez, dan E. N. Luque	<i>Analysis of Deforested Area Using Google Earth Engine in The Period 2001-2020 In the Apurimac Region</i> , J. Internet E-bus. Stud., 2022 [16]	<i>Decision Tree</i>	Selama periode 2001-2020 di wilayah Apurimac, terjadi kehilangan tutupan hutan primer yang mengalami deforestasi seluas 3958,231 hektar. Meskipun terdapat fluktuasi tahunan yang signifikan dalam laju deforestasi, analisis statistik menunjukkan bahwa tidak ada hubungan signifikan antara deforestasi dan periode waktu dengan nilai $r^2$ sangat rendah (0.001) dan nilai p bilateral yang lebih besar dari 0.05.
4	D. A. Kristiyanti dan Sri Hardani	<i>Sentiment Analysis of Public Acceptance of Covid-19 Vaccines Types in Indonesia using Naïve Bayes, Support Vector Machine, and Long Short-Term Memory (LSTM)</i> , J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 7, no. 3, 2023 [21]	<i>Naïve Bayes, Support Vector Machine, dan Long Short-Term Memory (LSTM)</i>	Penelitian tentang analisis sentimen terhadap vaksin Covid-19 di Indonesia dengan menggunakan algoritma SVM, Naïve Bayes, dan LSTM menunjukkan bahwa SVM memberikan hasil terbaik dengan akurasi 84,89%, presisi 85,00%, <i>recall</i> 100,00%, dan skor F-1 92,00%. Dibandingkan dengan Naïve Bayes dan LSTM, SVM lebih unggul. Analisis data Twitter mengungkapkan bahwa vaksin Sinovac menerima respons positif dan negatif terbanyak. Respons positif terkait dengan penggunaannya yang luas dan efek samping yang ringan. Namun, Sinovac juga dikritik karena dianggap kurang efektif, mendapat keraguan dari beberapa negara, laporan kematian pasca vaksinasi, dan timbulnya gejala pasca vaksinasi.
5	Ball, J. G. C., Petrova, K., Coomes, D. A., dan	<i>Using deep convolutional neural networks to forecast spatial patterns of Amazonian</i>	2D CNNs, 3D CNNs, <i>Convolutional Long Short-Term Memory (ConvLSTM)</i> ,	Hasil <i>network</i> dapat memprediksi lokasi kehilangan hutan di masa depan dengan tingkat akurasi tinggi ( $F1=0,58-0,71$ ). Model terbaik, 3D CNN, memiliki akurasi tertinggi ( $F1=0,71$ ) saat

No	Peneliti	Judul Artikel, Jurnal dan Tahun	Metode Penelitian	Hasil Penelitian
	Flaxman, S.	<i>deforestation, Methods in Ecology and Evolution</i> , 2022 [15]	<i>Recurrent Neural Networks (RNNs)</i>	divalidasi pada kehilangan hutan tahun 2020 ( <i>training data</i> 2014–2019). Interpretasi visual ramalan menunjukkan bahwa CNN dapat otomatis membedakan pemicu kehilangan hutan, seperti jalur akses baru, dengan risiko tinggi, sementara kehilangan alam terkonsentrasi tidak memperoleh risiko tinggi. CNN dapat memanfaatkan data <i>time-series</i> terbatas untuk memprediksi pola deforestasi masa depan, menjadi langkah penting dalam memanfaatkan data penginderaan jauh satelit untuk mengatasi deforestasi <i>global</i> . Model ini dapat diterapkan dengan mudah di berbagai lokasi hutan tropis, membantu pemerintah dan organisasi konservasi dalam pencegahan deforestasi dan perencanaan kawasan lindung.
6	M. Singh dan S. Yan	<i>Spatial–temporal variations in deforestation hotspots in Sumatra and Kalimantan from 2001–2018</i> , <i>Ecol. Evol.</i> , vol. 11, pp. 7302–7314, 2021 [4]	<i>Emerging Hotspot Analysis (EHA)</i>	<p>Dari hasil analisis EHA tentang tren deforestasi di Sumatra dan Kalimantan dari 2001 hingga 2018, ditemukan bahwa <i>hotspot</i> deforestasi banyak terjadi di luar kawasan lindung, khususnya di Riau, Jambi, dan Sumatra Selatan untuk Sumatra, serta di area pesisir timur dan selatan Kalimantan.</p> <p>Di Sumatra, terdeteksi tujuh <i>hotspot</i> deforestasi (<i>New, Consecutive, Intensifying, Persistent, Diminishing, Sporadic, dan Oscillating</i>) di bagian tengah dan selatan, terutama di provinsi Riau, Jambi, dan Sumatra Selatan. Sementara di Kalimantan, hanya tiga <i>hotspot</i> deforestasi (<i>Consecutive, Sporadic, dan Oscillating</i>) yang teridentifikasi, terutama di daerah pesisir timur dan selatan, serta beberapa di bagian barat. Empat kategori <i>hotspot</i> hanya muncul di Sumatra yaitu <i>New, Intensifying, Persistent, dan Diminishing</i>. <i>Hotspot</i> baru (<i>New</i>) teramati di beberapa provinsi dari utara (misalnya Aceh) hingga selatan</p>

No	Peneliti	Judul Artikel, Jurnal dan Tahun	Metode Penelitian	Hasil Penelitian
				(misalnya Sumatra Selatan). Meski terdapat perbedaan kategori <i>hotspot</i> di kedua pulau, <i>hotspot</i> yang <i>Oscillating</i> mendominasi di kedua pulau tersebut. Faktor topografi dan antropogenik seperti ketinggian rendah dan keberadaan jalan besar mempengaruhi distribusi <i>hotspot</i> ini. Hasil ini menekankan pentingnya mengutamakan area <i>hotspot</i> deforestasi sebagai target konservasi untuk melindungi keanekaragaman hayati dan meningkatkan layanan ekosistem dalam jangka panjang.
7	I. W. Mustika, H. N. Adi, dan F. Najib	<i>Comparison of Keras Optimizers for Earthquake Signal Classification Based on Deep Neural Networks</i> , ICOIACT 2021 - 4th International Conference on Information and Communications Technology: The Role of AI in Health and Social Revolution in Turbulence Era, 2021 [17]	<i>Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM)</i> , dengan <i>optimizers</i> Adadelta, Adagrad, Adam, Adamax, Nadam, RMSprop, dan SGD	Di antara <i>optimizers</i> yang diteliti, yaitu Adadelta, Adagrad, Adam, Adamax, Nadam, RMSprop, dan SGD, RMSprop menunjukkan performa superior dalam hal akurasi. Hasil penelitian menunjukkan bahwa RMSprop memberikan akurasi terbaik yaitu 99.2% dengan <i>loss</i> 0.102 untuk model 1 dan 99.0% dengan <i>loss</i> 0.107 untuk model 2, menjadikannya <i>optimizer</i> yang paling cocok untuk aplikasi yang menggabungkan CNN dan LSTM dalam klasifikasi gempa.
8	S. Zahara, Sugianto, dan M. B. Ilmiddaviq	<i>Consumer price index prediction using Long Short Term Memory (LSTM) based cloud computing</i> , Journal of Physics: Conference Series, 2020 [18]	<i>Cloud Computing Long Short-Term Memory (LSTM)</i>	Dalam penelitian ini, model prediksi Indeks Harga Konsumen atau <i>Consumer Price Index (CPI)</i> bulanan menggunakan metode dalam <i>Long Short-Term Memory (LSTM)</i> berjalan dengan <i>input</i> parameter non-linier berupa harga pangan pokok harian di Surabaya. Beberapa algoritma dan <i>optimizers</i> diterapkan untuk mendapatkan akurasi terbaik dalam metode LSTM. Hasilnya menunjukkan bahwa Nadam (Nesterov Adam) memberikan performa terbaik dengan nilai RMSE sebesar 4.088.
9	C. Jöriges, C. Berkenbri	<i>Prediction and reconstruction of ocean wave</i>	<i>Long Short-Term Memory (LSTM)</i>	Dalam penelitian ini, model LSTM yang mengintegrasikan data batimetri telah berhasil

No	Peneliti	Judul Artikel, Jurnal dan Tahun	Metode Penelitian	Hasil Penelitian
	nk, dan B. Stumpe	<i>heights based on bathymetric data using LSTM neural networks</i> , Ocean Eng., vol. 232, 2021 [19]		mengembangkan kemampuan prediksi tinggi gelombang signifikan SWH di pantai dengan peningkatan akurasi yang signifikan. Model ini mencatat peningkatan RMSE sebesar 16.7% untuk rekonstruksi, dan antara 7.4–11.7% untuk prediksi jangka pendek, serta 8.8–9.1% untuk prediksi jangka panjang. Struktur LSTM paralel (P-LSTM) dengan RMSE sebesar 0.069 m, diidentifikasi sebagai metode terbaik, menunjukkan efektivitas yang lebih tinggi dibandingkan dengan jaringan <i>feed-forward</i> dalam dan teknik pembelajaran mesin lainnya dalam prediksi SWH.
10	A. Fadli, A. Kosugo, K. Ichii, dan R. Ramli	<i>Satellite-based monitoring of forest cover change in indonesia using google earth engine from 2000 to 2016</i> , J. Phys. Conf. Ser., vol. 1317, p. 12046, 2019, doi: 10.1088/1742-6596/1317/1/012046 [7]	Google Earth Engine (Javascript API)	Dari hasil analisis, tutupan hutan Indonesia pada tahun 2000 diperkirakan sekitar 338.000 km <sup>2</sup> . Hasil analisis perubahan temporal tutupan hutan menunjukkan adanya variasi antar tahun dalam kehilangan tutupan hutan dari tahun 2000 hingga 2016. Puncak kehilangan terbesar terjadi pada tahun 2011, yang disebabkan oleh aktivitas manusia yang mengubah lahan hutan menjadi lahan untuk pembangunan. Selain itu, perubahan tutupan hutan yang signifikan juga terdeteksi pada tahun 2015.
11	R. Hesnananda	Algoritma Klasifikasi Bibit Terbaik untuk Tanaman Keladi Tikus. Penerbit NEM, 2021. [Online]. Available: <a href="https://books.google.co.id/books?id=e2pbEAAAQBAJ">https://books.google.co.id/books?id=e2pbEAAAQBAJ</a> [23]	Perbandingan metode penelitian CRISP-DM, KDD, dan SEMMA	Berdasarkan analisis, KDD dan SEMMA dinyatakan ekuivalen dalam berbagai tahapannya, dengan SEMMA sangat mirip dengan lima tahapan dalam KDD. Sementara itu, CRISP-DM memiliki beberapa tahap tambahan seperti <i>Business Understanding</i> dan <i>Deployment</i> yang tidak secara eksplisit ada dalam KDD. CRISP-DM lebih komprehensif dan terdokumentasi dengan baik dibandingkan SEMMA, dengan setiap fase terstruktur dan jelas sehingga lebih mudah diaplikasikan. CRISP-DM

No	Peneliti	Judul Artikel, Jurnal dan Tahun	Metode Penelitian	Hasil Penelitian
				menyediakan standar proses baku untuk data mining yang dapat diterapkan dalam berbagai strategi pemecahan masalah di bisnis maupun penelitian, membuatnya lebih mudah diterapkan bahkan bagi pemula.

Tabel 2.1 merupakan tabel penelitian terdahulu dengan topik serupa sebagai acuan dengan permasalahan, area fokus, metode, dan analisis yang berbeda-beda sesuai dengan tujuannya dibentuk penelitian. Penelitian memfokuskan pada penghubungan dan perbandingan antar data intensitas cahaya dan deforestasi di hutan Kalimantan untuk merumuskan suatu kesimpulan beserta prediksinya di masa depan. Karakteristik dan gambaran awal *dataset Night-Time Light* didapatkan dari penelitian terdahulu, yang menjelaskan analisis *time-series* menggunakan data terkait [14]. Pengolahan dan analisis dengan *dataset Global Forest Change* ditunjukkan oleh penelitian terdahulu, yang memberikan landasan awal penelitian dengan mengukur seberapa besar dinamika perubahan hutan, khususnya hutan di Pulau Kalimantan [4] [7] [15] [16].

Berdasarkan dari 11 penelitian terdahulu, disimpulkan bahwa terdapat beberapa penelitian terdahulu yang menggunakan *dataset* yang sama dalam tahapan pengolahan dan pengambilan data dari Google Earth Engine [7] [14] [16]. Keterhubungan ini menunjukkan pentingnya *dataset Global Forest Change* dan *Night-Time Light* sebagai referensi krusial. Google Earth Engine menyediakan fitur seperti *code editor* dan *image collection* yang dapat diakses dan diolah dengan metode *Exploratory Data Analysis* (EDA). Penelitian ini mengadopsi pendekatan pengestrakan data dari *image collection* Google Earth untuk menganalisis dinamika perubahan hutan dan distribusi intensitas cahaya. Penelitian ini membedakan diri dengan mengonversi citra satelit hutan menjadi data numerik yang kemudian diolah menggunakan Python, memungkinkan analisis yang lebih terperinci. Hal ini memberikan wawasan yang lebih mendalam tentang perubahan hutan dan memungkinkan penggunaan data yang lebih efisien.

Perbandingan metode, *framework*, *optimizers*, dan evaluasi yang menjadi acuan dalam metodologi penelitian ini telah dijelaskan oleh penelitian terdahulu [17] [18] [19] [20] [21]. Keterhubungan antar artikel ini menunjukkan bahwa *model Long Short-Term Memory* (LSTM) memiliki kemampuan yang efektif sebagai turunan dari *Recurrent Neural Network* untuk melakukan analisis prediksi secara *time-series* [20]. Beberapa penelitian terdahulu menggunakan berbagai metode evaluasi, termasuk RMSE, MAE, dan MSE, namun penelitian ini memfokuskan pada RMSE karena penelitian sebelumnya mendukung penggunaannya [18] [19]. RMSE memberikan penekanan lebih besar pada kesalahan yang besar dan memberikan estimasi yang lebih akurat tentang variabilitas dalam kumpulan data.

Dalam penelitian ini, perbandingan *optimizer* dilakukan berdasarkan model data, mengadaptasi dari penelitian terdahulu yang menggunakan berbagai *optimizer* untuk tuning algoritma LSTM dan mengevaluasi model dengan RMSE [17] [18]. Keterhubungan antar artikel menunjukkan bahwa penggunaan *optimizer* seperti Adadelta, Adagrad, Adam, Adamax, Nadam, RMSprop, dan SGD telah menjadi praktik umum dalam penelitian sebelumnya [17] [18]. Penelitian ini mengadopsi metode dari penelitian terdahulu dengan menekankan pada penggunaan RMSE sebagai metrik evaluasi utama. Pembedanya terletak pada perbandingan *optimizer* yang lebih luas, dengan fokus pada penyesuaian parameter seperti Adadelta, Adagrad, Adam, Adamax, Nadam, RMSprop, dan SGD dalam model LSTM. Pendekatan ini memungkinkan penelitian untuk mengeksplorasi dan mengerucutkan kombinasi *optimizers* yang lebih optimal guna meningkatkan akurasi prediksi. *Optimizer* yang akan diuji meliputi Adam, RMSprop, SGD, Adamax, dan Adadelta, untuk mengeksplorasi kombinasi yang paling efektif dalam meningkatkan akurasi prediksi model.

## **2.2 Teori yang digunakan**

### **2.2.1 Tutupan Hutan**

Tutupan lahan adalah konsep yang merujuk pada distribusi dan karakteristik penggunaan suatu wilayah tertentu dalam hal tutupan fisiknya, seperti hutan, lahan pertanian, pemukiman, air, dan area terbuka lainnya.

Analisis tutupan lahan menjadi krusial dalam pemahaman dinamika ekosistem dan dampak intervensi manusia terhadap lingkungan [24]. Tutupan hutan merupakan *subset* penting dari tutupan lahan, fokus pada distribusi dan karakteristik hutan di suatu wilayah. Keterkaitan antara tutupan hutan dan tutupan lahan menyediakan pemahaman mendalam tentang dampak intervensi manusia terhadap ekosistem hutan dan lingkungan lebih luas. Program distribusi hutan harus didasari dari pengalaman-pengalaman sebelumnya, menggabungkan pengetahuan ekologi, meningkatkan teknik regenerasi, mengatasi hambatan, dan menyesuaikan diri dengan tujuan di masa depan agar fungsi ekosistem tetap terjaga untuk generasi mendatang [25].

### **2.2.2 Deforestasi**

Deforestasi adalah proses pengurangan luas hutan yang dapat mencakup penebangan pohon secara besar-besaran, perubahan penggunaan lahan, atau aktivitas manusia lainnya yang menyebabkan hilangnya tutupan hutan [26]. memiliki dampak yang dapat berakibat fatal akibat hilangnya struktur dan fungsi ekosistem hutan itu sendiri. Hal ini mencakup kerugian aspek-aspek yang menjadi atribut dari tutupan lahan, membawa konsekuensi signifikan terhadap integritas dan kinerja alami hutan. Kelestarian hutan terancam oleh konversi lahan menjadi area non-hutan, terutama melalui usaha pertanian dan perkebunan yang berkembang seiring dengan peningkatan populasi. Penggunaan lahan untuk kegiatan ekonomi telah menyebabkan gangguan terhadap keamanan hutan, termasuk pelanggaran batas, penebangan liar, kebakaran hutan dan lahan, serta perdagangan ilegal flora dan fauna [27].

### **2.2.3 Global Forest Change**

*Global Forest Change* (GFC) merupakan *dataset* yang menyediakan informasi mengenai perubahan tutupan hutan di seluruh dunia. Dikembangkan oleh Global Forest Watch (GFW), *dataset* ini berdasarkan pengamatan citra satelit dan menyediakan pemahaman mendalam tentang



deforestasi, pertumbuhan hutan, dan dinamika perubahan lahan [28]. *Dataset* dari Global Forest Watch (GFW) sebagai produk *remote sensing* interaktif yang digunakan oleh peneliti, pelestari lingkungan, masyarakat lokal, dan 2 juta pengguna lainnya untuk menganalisis perubahan tutupan hutan [29].

GFC memberikan lapisan temporal pada perubahan lahan tersebut, memungkinkan analisis historis untuk mengevaluasi dampak aktivitas manusia terhadap ekosistem hutan. Berbagai macam penelitian dapat dikembangkan berdasarkan urutan kegiatan mengidentifikasi lokasi perubahan hutan, memantau tren seiring waktu, dan memberikan kontribusi dalam pengembangan strategi pelestarian hutan [30]. GFC menjadi instrumen penting dalam penelitian lingkungan dan keberlanjutan, memberikan landasan data yang kritis untuk analisis transformasi hutan di berbagai wilayah.

#### **2.2.4 Night-Time Light: VIIRS**

*Night-Time Light* (NTL) menggunakan sensor *Visible Infrared Imaging Radiometer Suite* (VIIRS) untuk mengukur radiasi cahaya yang dipancarkan oleh permukaan bumi pada malam hari yang mencakup rentang panjang gelombang spektral mulai dari 0.41 hingga 12.2  $\mu\text{m}$  dan wilayah cakupan mencapai 3000 km [31]. VIIRS adalah instrumen yang terpasang pada satelit suhu yang diluncurkan oleh *National Aeronautics and Space Administration* (NASA) dan *National Oceanic and Atmospheric Administration* (NOAA). VIIRS memberikan pemahaman mendalam tentang aktivitas manusia, distribusi populasi, dan perubahan perkotaan yang dapat digunakan dalam berbagai bidang, termasuk analisis perkotaan, pemantauan pertumbuhan populasi, dan evaluasi dampak kegiatan manusia terhadap lingkungan malam.

#### **2.2.5 Time-Series Analysis**

*Time-Series Analysis* merupakan pendekatan statistika yang digunakan untuk memahami dan menganalisis pola-pola yang terdapat dalam data sepanjang rentang waktu yang berurutan. Data *time-series* umumnya terdiri dari pengamatan yang diukur atau diamati secara berurutan pada interval waktu yang tetap, seperti contohnya pada transformasi hutan di Kalimantan. Keunggulan utama analisis ini adalah kemampuannya untuk menangkap dinamika perubahan seiring waktu, memberikan wawasan yang berharga untuk pengambilan keputusan dan perencanaan strategis.

*Time-Series Analysis* dilakukan dengan tujuan mendapatkan pola yang terkandung dalam data menggunakan data historis, yang nantinya akan diterapkan untuk meramalkan nilai di masa depan. Oleh karena itu, dalam pemilihan metode deret waktu yang tepat, perlu mempertimbangkan karakteristik pola yang ada dalam data. Hal ini bertujuan agar metode yang digunakan dapat diuji dan disesuaikan dengan pola khusus yang teridentifikasi dalam dataset tersebut [32]. Dalam *time-series*, penting untuk memastikan bahwa semua data input sudah bersifat stasioner. Verifikasi stasioneritas data dapat dilakukan melalui analisis *Autocorrelation Function* (ACF) dan *Partial Autocorrelation Function* (PACF).

Koefisien korelasi sederhana antara  $y_t$  dengan  $y_{t-1}$  didasari oleh rumus [32] :

$$r_k = \frac{\sum_t^n (Y_t - \bar{y})(Y_{t-1} - \bar{y})}{\sum_t^n (Y_{t-1} - \bar{y})^2}$$

Rumus 2.1 Rumus Koefisien Korelasi *Time-Series*

Keterangan rumus 2.1:

$r_k$  = koefisien korelasi lag  $k$

$k = 0, 1, 2, \dots, k$

$n$  = jumlah data

$Y_t$  = nilai stasioner

$\bar{y}$  = rata-rata perubah  $Y$  secara acak

Autokorelasi parsial didasari rumus berikut [32]:

$$\rho_k = \phi_1\rho_{k-1} + \phi_2\rho_{k-2} + \dots + \phi_p\rho_{k-p}$$

Rumus 2.2 Rumus Autokorelasi Parsial *Time-Series*

Keterangan rumus 2.2:

$\rho$  = nilai fungsi ACF

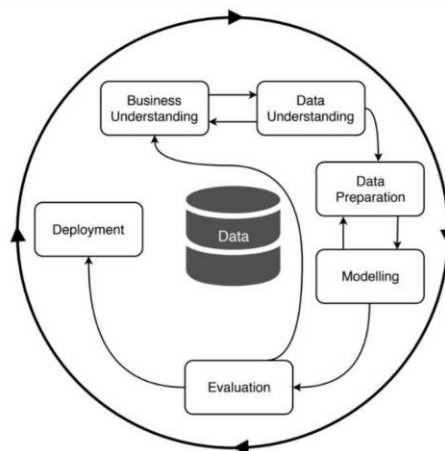
$\phi$  = nilai fungsi PCAF

$k$  = *time lag*

## 2.3 Framework/Algoritma yang digunakan

### 2.3.1 Cross-Industry Standard Process Model for Data Mining (CRISP-DM)

CRISP-DM (*Cross-Industry Standard Process for Data Mining*) adalah sebuah pendekatan standar yang digunakan untuk mengatasi permasalahan dalam proses *data mining*. CRISP-DM merupakan metode penelitian dengan tahapan *data mining* dari pemahaman awal terhadap kebutuhan bisnis hingga implementasi solusi yang dihasilkan. Metode ini memberikan kerangka kerja yang terstruktur untuk mengelola proyek *data mining*, sehingga memungkinkan pengerjaan analisis data dapat berjalan efisien dan efektif [23].



Gambar 2.1 Kerangka CRISP-DM [33]

Pendekatan ini terdiri dari enam tahapan utama yang harus dilalui dalam sebuah proyek *data mining* seperti pada gambar 2.1, yaitu *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation*, dan *Deployment* [33].

1. *Business Understanding*

Tahap ini melibatkan pemahaman terhadap tujuan bisnis dan kondisi bisnis yang relevan dengan permasalahan yang akan diselesaikan melalui *data mining*.

2. *Data Understanding*

Tahap ini melibatkan persiapan data awal dan identifikasi kualitas data, serta deskripsi fitur-fitur yang terdapat dalam data.

3. *Data Preparation*

Tahap ini melibatkan persiapan data yang meliputi identifikasi data, pemilihan data, pembersihan, dan transformasi data untuk mempersiapkannya untuk tahap *modeling*.

4. *Modeling*

Tahap ini melibatkan penerapan algoritma untuk menghasilkan pola data yang sesuai dengan data penelitian.

5. *Evaluation*

Tahap ini melibatkan evaluasi terhadap model yang telah dibuat untuk menentukan apakah model tersebut sesuai untuk proses *data mining* tertentu.

6. *Deployment*

Tahap ini melibatkan penggunaan model yang dihasilkan dalam proses data mining. Tahap ini mencakup perencanaan, pemantauan, pembuatan laporan akhir, dan tinjauan proyek.

### 2.3.2 Knowledge Discovery in Databases (KDD)

*Knowledge Discovery in Databases* (KDD) adalah proses iteratif dan interaktif yang tidak hanya berfokus pada analisis data tetapi juga pada

interpretasi dan pemahaman pola yang ditemukan. KDD telah menjadi alat yang sangat diperlukan dalam mengekstraksi informasi berharga dari kumpulan data yang besar dan kompleks [34]. Proses KDD melibatkan beberapa tahapan, termasuk *Data Selection*, *Data Preprocessing*, *Data Transformation*, *Data Mining*, dan *Interpretation/Evaluation* [35].

Berikut ini merupakan masing-masing penjelasan tahapan dari metode KDD [23]:

1. *Data Selection*

Pada tahap ini, data yang relevan dipilih dari *database* yang besar untuk dianalisis lebih lanjut. Pemilihan data ini bertujuan untuk fokus pada *subset* data yang berhubungan dengan masalah tertentu dan dapat meningkatkan efisiensi proses KDD.

2. *Data Preprocessing*

*Data Preprocessing* bertujuan untuk meningkatkan kualitas data dengan menangani masalah seperti data yang hilang, data yang tidak konsisten, dan *outlier*. Teknik pembersihan data seperti imputasi, normalisasi, dan standardisasi sering digunakan pada tahap ini untuk memastikan bahwa data siap untuk dianalisis.

3. *Data Transformation*

Transformasi data melibatkan perubahan data ke dalam format yang lebih sesuai untuk *data mining*. Teknik transformasi data seperti reduksi dimensi, agregasi, dan diskritisasi dapat digunakan untuk menyederhanakan data untuk pengelolaan saat analisis.

4. *Data Mining*

Inti dari proses KDD dengan pengolahan berbagai algoritma dan teknik diterapkan untuk mengidentifikasi pola, tren, dan hubungan tersembunyi dalam data.

5. *Interpretation and Evaluation*

Pola dan pengetahuan yang ditemukan pada tahap *data mining* harus ditafsirkan dan dievaluasi untuk memastikan validitas, kebaruan, dan kegunaannya. Tahap ini melibatkan analisis statistik, visualisasi

data, dan penilaian ahli untuk memahami implikasi dari temuan dan menentukan tindakan selanjutnya.

### 2.3.3 Sample, Explore, Modify, Model, and Assessment (SEMMA)

SEMMA adalah akronim dari *Sample*, *Explore*, *Modify*, *Model*, dan *Assess*, yang merupakan metodologi dalam proses *data mining* yang dikembangkan oleh SAS Institute [23]. SEMMA dirancang untuk memberikan pendekatan yang terstruktur dan efisien dalam menganalisis data untuk mendapatkan wawasan dan membuat prediksi yang akurat [36].

Berikut ini merupakan masing-masing penjelasan tahapan dari metode SEMMA [23] [37]:

1. *Sample*

Tahap pemilihan sampel yang representatif dari *dataset* yang lebih besar yang cukup besar untuk mencerminkan karakteristik populasi namun cukup kecil untuk dikelola dengan efisien.

2. *Explore*

Tahap penjelajahan data sampel untuk mengidentifikasi pola, tren, dan hubungan menggunakan visualisasi data dan analisis statistik deskriptif.

3. *Modify*

Tahap modifikasi ini dapat meningkatkan kualitas data untuk pemodelan dengan transformasi variabel, pembuatan variabel baru, atau penanganan *outlier*.

4. *Model*

Tahap penerapan berbagai teknik pemodelan untuk membangun model prediktif, memilih metode yang tepat berdasarkan jenis masalah dan karakteristik data.

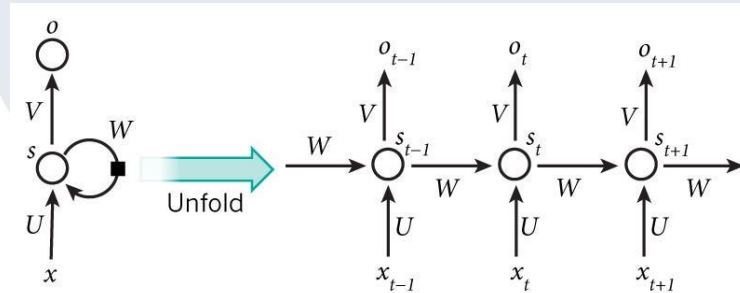
5. *Assessment*

Tahap penilaian kinerja model yang dibangun menggunakan metrik seperti akurasi, presisi, *recall*, dan F1 score untuk memastikan bahwa

model bekerja dengan baik dan memiliki kemampuan generalisasi yang baik terhadap data baru.

### 2.3.4 Recurrent Neural Network (RNN)

*Recurrent Neural Network* (RNN) adalah jenis turunan arsitektur jaringan saraf berulang dari *Artificial Neural Network* (ANN). RNN dirancang khusus untuk memproses data berurutan, seperti teks atau kalimat. Kelebihan utama RNN adalah kemampuannya untuk menyimpan informasi yang dapat digunakan untuk memahami konteks dari data sebelumnya dalam proses pembelajarannya [38].



Gambar 2.2 Proses Pemodelan *Recurrent Neural Network*

Proses model ini bersifat berulang secara *forward* dan *backward* dengan *input* yang langsung diproses di *hidden layer*. Pada tahapan *hidden layer* ini dilakukan perhitungan, memperbarui, dan menyimpan terhadap bobot RNN sebelumnya secara berurutan seperti pada gambar 2.2 dengan keterangan sebagai berikut [39]:

- a.  $X_t$  sebagai *input*
- b.  $S_t$  sebagai *hidden state*
- c.  $O_t$  sebagai *output*
- d.  $t$  sebagai langkah *time state*

Dalam melakukan perhitungan fungsi *hidden state*, umumnya digunakan fungsi aktivasi seperti *Rectified Linear Unit* (ReLU) atau *hyperbolic tangent*. Rumus 2.3 adalah rumus untuk persamaan perhitungan pada *hidden state* pada waktu  $t$  dalam RNN sebagai berikut [38]:

$$h_t = f(W_{hh} \cdot h_{t-1} + W_{xh} \cdot x_t + b_h)$$

Rumus 4.3 Rumus *Hidden State* RNN

Selain model jaringan saraf RNN, terdapat model berbasis *Convolutional Neural Network* (CNN) sebagai salah satu turunan dari ANN. CNN menawarkan akurasi tinggi dan kecepatan pelatihan yang cepat dalam analisis data berbentuk gambar (*image recognition*). Namun, penggunaan model RNN lebih mudah untuk menangkap data bersifat urutan (*sequence*), karena memiliki kemampuan prediksi yang lebih tinggi dan memerlukan waktu pelatihan yang jangka panjang [40].

### 2.3.5 Long Short-Term Memory (LSTM)

*Long Short-Term Memory* (LSTM) adalah metode turunan dan pengembangan dari *Recurrent Neural Network* (RNN). LSTM memodifikasi dan mengembangkan hasil analisis data urutan/*sequence modelling*, seperti teks atau kalimat yang lebih stabil dan mencegah *vanishing gradient* yang dialami pada model RNN [41]. LSTM mampu melakukan model data dengan *long-range dependencies* yang tidak dapat dioptimalisasi akurasi oleh RNN. LSTM memiliki desain arsitektur yang mirip dengan RNN, tetapi LSTM mampu belajar menghubungkan informasi dari *input per gate* sebelumnya dengan lebih baik. Hal ini berbeda dengan RNN yang cenderung kehilangan informasi yang ada dalam memori lama karena menimbun memori baru. LSTM mengatasi masalah ini dengan menggunakan *memory cells* dan *gate units* yang memungkinkannya untuk mengatur memori pada setiap *input* [42].

LSTM memiliki tiga *gate*, yaitu *input gate* (untuk mengatur nilai yang diperbarui), *forget gate* (untuk mengatur informasi yang dilupakan), dan *output gate* (untuk mengatur bagian yang akan dihasilkan) [38]. *Forget gate* adalah bagian pertama dari LSTM yang memiliki peran untuk menentukan informasi dari *cell state* sebelumnya ( $C_{t-1}$ ) yang akan disimpan dalam *cell state* pada waktu  $t$  ( $C_t$ ). *Forget gate* menghasilkan nilai antara 0 hingga 1



menggunakan fungsi sigmoid ( $\sigma(\cdot)$ ). Berikut rumus persamaan untuk *forget gate* pada rumus 4.4 adalah [38]:

$$f_t = \sigma(W_{hf} \cdot h_{t-1} + W_{xf} \cdot x_t + b_f)$$

Rumus 4.4 Rumus Persamaan *Forget Gate*

Bagian kedua dari LSTM terdiri dari *input gate* ( $i_t$ ) dan *candidate state* ( $\tilde{c}_t$ ). *Input gate* digunakan untuk menentukan nilai mana yang akan diperbarui, sementara *candidate state* bertanggung jawab untuk mengumpulkan dan memperbarui informasi baru dari data. *Candidate state* mirip dengan *cell state* ( $h_t$ ) pada RNN, tetapi berbeda dalam cara pembaruan. *Cell state* pada RNN meneruskan informasi secara langsung, sementara *candidate state* pada LSTM digunakan untuk memperbarui *cell state* ( $C_t$ ). Berikut merupakan rumus persamaan dari *input gate* dan *candidate state* pada rumus 4.5 dan rumus 4.6 adalah [38]:

$$\tilde{c}_t = \tanh(W_{hc} \cdot h_{t-1} + W_{xc} \cdot x_t + b_c)$$

Rumus 4.5 Rumus Persamaan *Input Gate*

$$i_t = \sigma(W_{hi} \cdot h_{t-1} + W_{xi} \cdot x_t + b_i)$$

Rumus 4.6 Rumus Persamaan *Candidate State*

Bagian keempat dari LSTM adalah *cell state* ( $C_t$ ), yang berfungsi menggabungkan hasil perhitungan dari *input gate*, *candidate state*, dan *forget gate*. *Cell state* dapat dirumuskan dengan persamaan pada rumus 4.7 sebagai berikut [38]:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{c}_t$$

Rumus 4.7 Rumus Persamaan *Cell State*

Bagian kelima dari LSTM terdiri dari *state* ( $h_t$ ) dan *output gate* ( $o_t$ ). *State* ( $h_t$ ) berperan dalam mengirim informasi ke *cell* berikutnya dan menggunakan fungsi aktivasi *hyperbolic tangent*. *Output gate* mengatur seberapa banyak informasi yang akan diteruskan ke *cell* berikutnya. Berikut

rumus persamaan dari *state* ( $h_t$ ) dan *output gate* pada rumus 4.8 dan rumus 4.9 adalah [38]:

$$h_t = o_t \odot \tanh(C_t)$$

Rumus 4.8 Rumus Persamaan *State*  $h_t$

$$O_t = \sigma(W_{ho} \cdot h_{t-1} + W_{xo} \cdot x_t + b_o)$$

Rumus 4.9 Rumus Persamaan *Output Gate*

### 2.3.6 Root Mean Square Error (RMSE)

*Root Mean Square Error* (RMSE) adalah metrik evaluasi yang umum digunakan dalam pemodelan data untuk mengukur sejauh mana perbedaan antara nilai yang diprediksi oleh model dengan nilai yang sebenarnya dari data yang diamati. RMSE memberikan gambaran tentang tingkat ketepatan model dalam menyesuaikan data. Nilai RMSE yang lebih rendah menunjukkan performa model yang lebih baik. Pemahaman yang mendalam tentang RMSE membantu peneliti dan praktisi dalam mengukur kualitas prediksi model dan membuat perbaikan yang diperlukan untuk meningkatkan kinerja model dalam memodelkan data dengan akurat.

Cara menilai dari evaluasi model prediksi dengan menggunakan RMSE adalah dengan melihat nilai RMSE yang semakin mendekati atau bahkan menyentuh nilai 0, bahwa model memiliki hasil prediksi yang baik. Sebaliknya, nilai RMSE yang semakin tinggi dari nilai 0 memiliki arti bahwa model prediksi kurang baik. Berikut persamaan rumus dari *Root Mean Square Error* [43] pada rumus 4.10:

$$RMSE = \sqrt{\frac{\sum (y_t - \hat{y}_t)^2}{n}}$$

Rumus 4.10 Rumus RMSE

Keterangan:

N = jumlah total sampel

$y_t$  = nilai aktual dari indeks

$\hat{y}_t$  = nilai prediksi dari indeks

## 2.4 Tools yang digunakan

### 2.4.1 Google Earth Engine

Google Earth Engine (GEE) adalah *platform* pengolahan data geospasial yang dikembangkan oleh Google untuk mendukung analisis dan pemodelan berbasis citra satelit. GEE merupakan teknologi paralel massal yang dirancang untuk pengolahan data geospasial berperforma tinggi dan memiliki salinan lengkap dari katalog Landsat, Sentinel, MODIS, dan lain sebagainya, salah satunya untuk mendeteksi perubahan tutupan tanah [16]. GEE memungkinkan pengguna untuk melakukan analisis spasial, ekstraksi informasi, dan pemodelan tanpa perlu mengunduh atau memproses data secara lokal. Google Earth Engine memungkinkan analisis geospasial skala planet untuk isu-isu sosial seperti deforestasi, kekeringan, dan perubahan iklim, memberikan manfaat baik bagi ilmuwan pemantauan jarak jauh tradisional maupun mereka yang memiliki keterbatasan kapasitas teknis [44]. *Platform* ini juga memberikan kemampuan untuk membuat peta interaktif, visualisasi temporal, dan pemantauan perubahan lahan. Kemampuan untuk mengintegrasikan alat pemrograman seperti JavaScript dan Python, Google Earth Engine menjadi alat yang sangat efektif untuk penelitian di berbagai bidang, termasuk pemantauan lingkungan, perubahan iklim, dan keberlanjutan sumber daya alam.

### 2.4.2 Python

Python sebagai bahasa pemrograman tingkat tinggi dalam pengembangannya beragam secara objektif dan dapat dilakukan dalam berbagai bidang, contohnya dalam pengolahan data. Python memiliki bentuk yang jelas dan penamaan yang sederhana sehingga mudah dipahami. Bahasa Python dalam pengolahan data, umumnya digunakan untuk analisis

statistik, pengembangan model *machine learning*, *data cleansing*, atau apa pun yang berkaitan dengan pekerjaan di bidang *data science* [45].

Python memiliki banyak *library* berbagai bidang yang dapat digunakan untuk mempermudah penggunaan fungsi dalam pemrograman Python, seperti Scikit-Learn (untuk pemodelan AI dan ML), TensorFlow (untuk aplikasi *computer vision*), dan OpenCV Python (untuk implementasi *deep learning*) [46].

### 2.4.3 Anaconda

Anaconda sebagai *packet* distribusi dari Python untuk kebutuhan *continuum analytics*, yakni distribusi untuk keperluan pemrograman pekerjaan di bidang *data science*, matematika, hingga teknik tertentu di dalam satu layanan *platform* yang sama dan mudah digunakan [47]. Anaconda menjadi landasan dari layanan Jupyter, khususnya aplikasi Jupyter Notebook yang tersedia dalam paket distribusi dari Anaconda untuk bahasa pemrograman Python.

### 2.4.4 Jupyter Notebook

Jupyter adalah sebuah layanan aplikasi dari perusahaan Jupyter, yakni sebuah organisasi *non-profit* yang mendukung pengembangan *software* interaktif untuk berbagai bahasa pemrograman. Notebook menjadi salah bentuk representasi *software* interaktif dari Jupyter sendiri yang memungkinkan pengembang aplikasi untuk membuat hingga berbagi dokumen interaktif, seperti proyek *coding*, *live*, *visualization*, dan teks naratif. Penggunaan Jupyter Notebook yang bisa mengkombinasikan suatu *workspace* dengan kode-kode dari dokumen yang berbeda ke dalam suatu wadah proyek yang sama sehingga dapat ditampilkan seluruh *tools* dan menunya, seperti *field* untuk *coding*, tampilan hasil *output*, dan visualisasi dari pengerjaan dalam *workspace* [48].

#### 2.4.5 Tableau

Tableau adalah alat visualisasi data yang membantu peneliti dengan mudah membuat visualisasi yang menarik untuk meningkatkan pemahaman terhadap data dan analisis mereka [49]. Dengan menggunakan berbagai jenis grafik dan visualisasi interaktif, Tableau memungkinkan pembuat keputusan untuk mengeksplorasi data dengan lebih mendalam, mengidentifikasi pola, dan membuat keputusan berdasarkan wawasan yang diberikan oleh visualisasi, seperti *chart*, *heatmap*, *plot*, dan lain sebagainya. Teori visualisasi data mendukung penggunaan alat, seperti Tableau untuk meningkatkan efektivitas dalam menyampaikan informasi dan menyederhanakan kompleksitas data.

