

Comparison of LSTM Model Performance with Classical Regression in Predicting Gaming Laptop Prices in Indonesia

Agus Dewantoro ^{1*}, Theopilus Bayu Sasongko ^{2*}

*Informatika, Universitas Amikom Yogyakarta

agusdewantoro@students.amikom.ac.id ¹, theopilus.27@amikom.ac.id ²

Article Info

Article history:

Received 2024-07-05

Revised 2024-07-08

Accepted 2024-07-11

Keyword:

Classical Regression,
Gaming Laptops,
LSTM,
Price Prediction,
Regression.

ABSTRACT

The demand for gaming laptops has surged in the digital era, appealing to both professional gamers and the general public. Gaming laptops come equipped with advanced features such as powerful graphics, fast processors, and sleek designs, offering a portable solution for gaming enthusiasts. However, the price of gaming laptops varies due to factors like brand, hardware specifications, screen size, and additional features. Accurately predicting these prices can help consumers make informed purchasing decisions and assist manufacturers in setting competitive prices. This research proposes the use of the Long Short-Term Memory (LSTM) algorithm to predict gaming laptop prices, comparing its performance with classic regression algorithms such as Linear Regression and Multi-layer Perceptron. Utilizing a comprehensive dataset of gaming laptop prices and specifications in Indonesia, this study employs robust pre-processing and model optimization techniques. The results show that the LSTM model achieves a Root Mean Squared Error (RMSE) of 0.09011, a Mean Squared Error (MSE) of 0.00812, and an R² Score of 0.90016. In comparison, the Linear Regression model has an RMSE of 0.09075, an MSE of 0.00823, and an R² Score of 0.89873, while the Multi-layer Perceptron model has an RMSE of 0.09891, an MSE of 0.00978, and an R² Score of 0.87971. These results indicate that the Long Short-Term Memory algorithm outperforms other classic regression algorithms in this case. This study highlights the potential of LSTM in developing a robust price prediction model for gaming laptops, particularly in the Indonesian market, providing valuable insights for both consumers and manufacturers.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Pasar laptop gaming di Indonesia mengalami pertumbuhan yang belum pernah terjadi sebelumnya, didorong oleh meningkatnya popularitas gaming di kalangan konsumen [1]. Dengan lonjakan permintaan ini, muncul kebutuhan untuk model prediksi harga yang akurat guna membantu konsumen membuat keputusan pembelian yang tepat dan pengecer mengoptimalkan strategi harga mereka. Sebagai perangkat yang sangat canggih, laptop gaming menawarkan kombinasi unik dari grafis yang kuat, prosesor cepat, dan desain yang ramping, menjadikannya solusi portabel yang sangat diminati oleh para penggemar game. Namun, harga laptop gaming bervariasi karena berbagai faktor seperti merek, spesifikasi

hardware, ukuran layar, dan fitur tambahan seperti sistem pendingin yang efisien dan keyboard mekanis. Selain itu, faktor eksternal seperti fluktuasi nilai tukar mata uang, biaya produksi, dan kebijakan impor juga mempengaruhi harga di pasar lokal. Prediksi harga laptop gaming penting karena membantu konsumen dan produsen. Konsumen bisa mendapatkan panduan tentang kapan waktu terbaik untuk membeli laptop gaming dengan harga terbaik, sementara produsen dapat mengoptimalkan strategi penetapan harga untuk tetap kompetitif di pasar. Dengan permintaan yang terus meningkat dan variasi harga yang besar, model prediksi harga yang akurat dapat membantu dalam perencanaan anggaran dan strategi pemasaran. Dengan demikian, menentukan harga dengan optimal menjadi tantangan yang

kompleks bagi produsen dan konsumen. Produsen harus mempertimbangkan keseimbangan antara menawarkan teknologi terbaru dan menjaga harga tetap kompetitif, sementara konsumen mencari nilai terbaik untuk investasi mereka dalam perangkat gaming yang tahan lama dan berkinerja tinggi. Kebutuhan akan model prediksi harga yang akurat menjadi semakin penting untuk memberikan panduan yang jelas kepada kedua belah pihak. Model prediksi yang efektif dapat membantu konsumen memahami tren harga dan mengidentifikasi waktu terbaik untuk membeli, sementara pengecer dan produsen dapat menggunakan informasi ini untuk mengembangkan strategi harga yang lebih efektif dan responsif terhadap perubahan pasar.

Penelitian ini mengusulkan penggunaan algoritma Long Short-Term Memory (LSTM) untuk memprediksi harga laptop gaming, sebuah pendekatan yang telah terbukti sukses dalam berbagai aplikasi seperti pengenalan suara, pemrosesan bahasa alami, dan analisis deret waktu. Dalam konteks prediksi harga, LSTM menonjol karena kemampuannya dalam menangkap pola-pola kompleks dari data deret waktu, yang relevan dengan fluktuasi harga laptop gaming yang dipengaruhi oleh faktor-faktor seperti kemajuan teknologi, preferensi konsumen, dan dinamika tren pasar yang cepat [2]. Selain LSTM, penelitian ini juga akan membandingkan kinerja model LSTM dengan algoritma regresi klasik seperti Linear Regression dan Multi-layer Perceptron (MLP). Tujuan dari perbandingan ini adalah untuk mengevaluasi apakah LSTM dapat memberikan prediksi harga yang lebih akurat dan andal dibandingkan dengan pendekatan regresi klasik. Linear Regression sering digunakan sebagai baseline dalam prediksi harga karena sifatnya yang sederhana dan interpretatif, sementara MLP merupakan jaringan saraf yang lebih kompleks yang dapat menangani pola-pola non-linear dalam data.

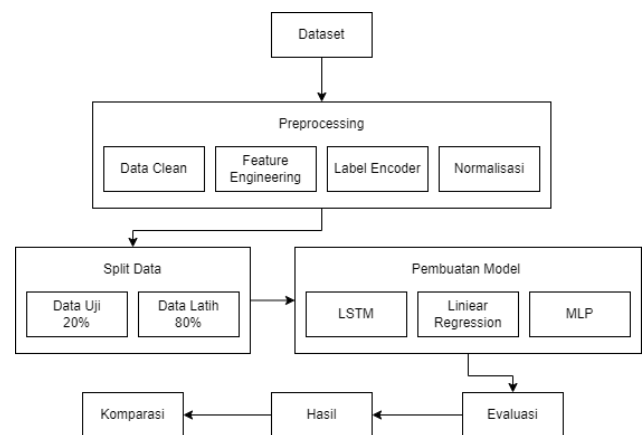
Penelitian sebelumnya telah memberikan wawasan tentang berbagai aspek dinamika penjualan laptop dan penentu harga. Misalnya, Ayu Saputri et al. [1] melakukan analisis mendalam tentang tren penjualan laptop di Banjarbaru, menekankan pentingnya kemitraan strategis dan diversifikasi produk dalam mendorong pertumbuhan penjualan. Selain itu, para peneliti telah mengeksplorasi berbagai metodologi untuk sistem prediksi harga dan rekomendasi yang disesuaikan dengan pasar laptop. Lavindi dan Rohmani [3] mengembangkan pendekatan penyaringan hibrid dan Naïve Bayes untuk merekomendasikan pembelian laptop, mencapai tingkat akurasi yang signifikan. Shaik et al. [4] dan Sharma dan Bhushan [5] menggunakan algoritma pembelajaran mesin seperti Random Forest, Linear Regression, dan Decision Trees untuk meramalkan harga laptop dengan akurasi yang baik. Namun, sifat dinamis dari pasar laptop gaming memerlukan pendekatan komprehensif untuk prediksi harga.

Penelitian ini bertujuan untuk mengembangkan model prediksi harga yang kuat untuk laptop gaming, terutama di Indonesia, guna membantu konsumen membuat keputusan pembelian yang tepat dan membantu produsen menetapkan harga yang sesuai dengan pasar. Dengan menggunakan data

harga historis yang ekstensif dan fitur-fitur yang komprehensif, termasuk spesifikasi teknis laptop, reputasi merek, ulasan pengguna, dan tren pasar, penelitian ini berusaha memberikan solusi holistik dan akurat dalam memprediksi harga laptop gaming. Penelitian ini juga mempertimbangkan berbagai faktor yang mempengaruhi harga, seperti perkembangan teknologi, perubahan preferensi konsumen, dan dinamika persaingan di pasar. Data yang digunakan diambil dari berbagai sumber terpercaya untuk memastikan keakuratan dan relevansi model prediksi yang dikembangkan. Dengan menerapkan algoritma machine learning seperti LSTM, Linear Regression, dan Multi-layer Perceptron, penelitian ini mengeksplorasi kekuatan dan kelemahan masing-masing metode dalam konteks prediksi harga. Evaluasi kinerja model dilakukan menggunakan metrik evaluasi yang umum dalam regresi, seperti Root Mean Squared Error (RMSE), Mean Squared Error (MSE), dan Skor R^2 . Hasil penelitian ini diharapkan memberikan panduan berharga bagi konsumen dan produsen serta berkontribusi pada literatur ilmiah di bidang prediksi harga dan aplikasi machine learning dalam analisis pasar, sehingga memiliki potensi untuk meningkatkan efisiensi dan efektivitas pengambilan keputusan terkait harga laptop gaming di Indonesia.

II. METODE PENELITIAN

Dalam penelitian ini dilakukan beberapa tahapan yang penting, yaitu diawali dengan menganalisis dataset hingga melakukan evaluasi dan komparasi.



Gambar 1. Tahapan Penelitian

Gambar 1 menunjukkan mengenai tahapan penelitian yang dilakukan pada penelitian ini. Berikut adalah penjelasan dari masing-masing tahapannya.

A. Dataset

Penelitian ini menggunakan Dataset Laptop Gaming yang merupakan Dataset Publik yang di ambil dari *Kaggle* [6]. Dataset ini terdiri dari kumpulan data yang mencerminkan daftar laptop gaming yang saat ini dijual di sebuah toko.

Dataset tersebut diambil pada awal November 2023 dari website toko KliknKlik. Dataset ini berisi 646 baris dan 17 kolom yang di antaranya Nama, Berat, Dimensi, Garansi, VGA, Tipe Penyimpanan, Processor, Penyimpanan, Ukuran Layar, Warna, RAM, Keyboard, Microsoft Office, Sistem Operasi, Brand, Tipe Layar, dan Harga. Variabel-variabel ini mencerminkan spesifikasi teknis dan fitur yang mempengaruhi harga laptop gaming..

B. Preprocessing Data

Dalam penelitian ini, langkah awal yang dilakukan setelah membaca dataset adalah preprocessing data. Proses tersebut mencakup pembersihan data, rekayasa fitur, pelabelan dan normalisasi. Berikut penjelasan lebih lengkapnya:

1) *Data Clean*: Dataset Laptop Gaming dilakukan pembersihan data di mana kolom yang tidak relevan atau tidak digunakan dihapus untuk menyederhanakan dataset. Dalam kasus ini kolom Keyboard dan Microsoft Office dihilangkan karena kedua kolom tersebut memiliki banyak nilai Null serta kolom Berat, Dimensi dan Warna dihilangkan karena kurang relevan untuk memprediksi harga laptop.

2) *Feature Engineering*: Selanjutnya, dilakukan transformasi data agar dapat diproses oleh model yang akan dibangun, termasuk penanganan nilai yang hilang dan konversi fitur kategorikal. Pada proses ini nilai dari kolom Berat, Ukuran Layar, dan RAM dibuat menjadi numerik. Selain itu untuk kolom Penyimpanan dilakukan konversi dari string ke numerik. Contoh hasil dari proses Feature Engineering dapat dilihat pada Gambar 2.

Penyimpanan	Ukuran Layar	RAM	Penyimpanan	Ukuran Layar	RAM
2TB	18 Inch	16GB	2000	18	16
2TB	18 Inch	16GB	2000	18	16
2TB	16 Inch	16GB	2000	16	16
1TB	16 Inch	16GB	1000	16	16
1TB	16 Inch	8GB	1000	16	8

Gambar 2. Contoh Transformasi Data

3) *Label Encoder*: LabelEncoder digunakan untuk mengonversi kolom data yang berisi label teks (kategori) menjadi angka yang sesuai. Misalnya untuk kolom Tipe Layar untuk layar yang Non Touchscreen diberi label 0 dan layar yang Touchscreen diberi label 1. Hasil dari proses Label Encoder dapat dilihat pada Gambar 3.

Tipe Layar	Tipe Layar
Non Touchscreen	0
Non Touchscreen	0
Non Touchscreen	0
Non Touchscreen	0
Non Touchscreen	0

Gambar 3. Contoh Labeling Data

4) *Normalisasi*: Langkah terakhir dalam preprocessing data adalah normalisasi fitur yang dilakukan untuk menyeimbangkan rentang nilai dari berbagai fitur dalam dataset, sehingga meningkatkan kinerja model.

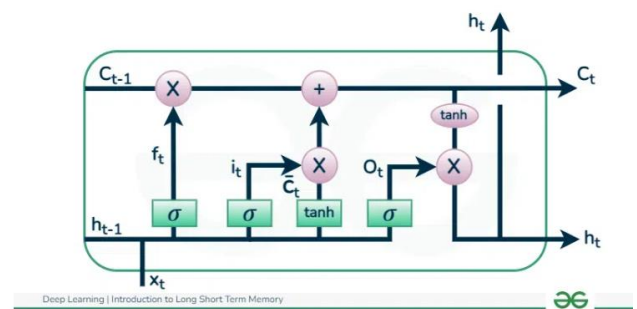
C. Split Data

Proses selanjutnya adalah pembagian data menjadi dua bagian yaitu data latih (training data) dan data uji (testing data). Disini data dibagi menjadi 20% untuk data uji dan 80% data latih.

D. Model

Penelitian ini menggunakan algoritma LSTM (Long Short-Term Memory), Linear Regression dan Multi-layer Perceptron (MLP). Berikut adalah penjelasan dari masing-masing algoritma yang diusulkan:

1) *LSTM (Long Short-Term Memory)*: LSTM adalah versi perbaikan dari jaringan saraf berulang yang dirancang oleh Hochreiter & Schmidhuber. RNN tradisional memiliki satu keadaan tersembunyi yang melewati waktu, sehingga menyulitkan jaringan untuk mempelajari ketergantungan jangka panjang. Model LSTM mengatasi masalah ini dengan memperkenalkan sel memori, yang merupakan wadah yang dapat menyimpan informasi untuk jangka waktu lama. Arsitektur LSTM mampu mempelajari ketergantungan jangka panjang dalam data sekuensial, sehingga cocok untuk tugas-tugas seperti terjemahan bahasa, pengenalan ucapan, dan perkiraan deret waktu. LSTM unggul dalam tugas prediksi urutan, menangkap ketergantungan jangka panjang. Ideal untuk deret waktu, terjemahan mesin, dan pengenalan suara karena ketergantungan pesanan [2]. LSTM dipilih karena kemampuannya dalam menangkap pola-pola kompleks dari data deret waktu, terutama yang memiliki ketergantungan jangka panjang. Berbeda dengan ARIMA yang lebih cocok untuk data linear dan Prophet yang lebih sederhana, LSTM mampu memproses data dengan dinamika yang lebih kompleks seperti fluktuasi harga laptop gaming yang dipengaruhi oleh banyak faktor. Rumus dasar untuk gate dalam LSTM dapat dilihat pada Gambar 4.



Gambar 4. Arsitektur LSTM

Gambar 4 menunjukkan Arsitektur LSTM memiliki struktur rantai yang berisi empat jaringan saraf dan blok memori berbeda yang disebut sel. Kemudian berikut adalah rumus dasar gate dalam LSTM:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\hat{c}_t = \tanh(WC \cdot [h_{t-1}, x_t] + bC) \tag{3}$$

$$C_t = f_t * C_t - 1 + i_t * \hat{C}_t \tag{4}$$

$$O_t = \sigma(W_o[h_t - 1, x_t] + b_o) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

Keterangan:

- W_f mewakili matriks bobot yang terkait dengan gerbang lupa
- $h_t - 1, x_t$ menunjukkan rangkaian input saat ini dan status tersembunyi sebelumnya.
- b_f adalah bias dengan gerbang lupa.
- σ adalah fungsi aktivasi sigmoid.

2) *Linear Regression*: Linear Regression adalah jenis algoritma supervised machine learning yang menghitung hubungan linier antara variabel terikat dan satu atau lebih fitur independen dengan menyesuaikan persamaan linier ke data yang diamati. Interpretabilitas regresi linier merupakan kekuatan penting. Persamaan model memberikan koefisien yang jelas yang menjelaskan dampak setiap variabel independen terhadap variabel dependen, sehingga memfasilitasi pemahaman yang lebih mendalam tentang dinamika yang mendasarinya [7]. Dalam kasus ini menggunakan Regresi Linier Berganda yang dimana melibatkan lebih dari satu variabel independen dan satu variabel dependen. Persamaan regresi linier berganda adalah:

$$Y = B_0 + B_1X + B_2X + \dots + B_nX \tag{7}$$

Keterangan:

- Y adalah variabel terikat
- X_1, X_2, \dots, X_p merupakan variabel bebas
- β_0 adalah intersep
- $\beta_1, \beta_2, \dots, \beta_n$ adalah lerengnya

3) *Multi-layer Perceptron (MLP)*: MLP adalah jenis jaringan saraf tiruan feedforward dengan beberapa lapisan, termasuk lapisan input, satu atau lebih lapisan tersembunyi, dan lapisan output. Setiap lapisan terhubung sepenuhnya ke lapisan berikutnya. Jaringan Saraf MultiLayer Perceptron adalah Jaringan Saraf dengan beberapa lapisan, dan semua lapisannya terhubung. Ia menggunakan algoritma BackPropagation untuk melatih model. MultiLayer Perceptron adalah salah satu kelas Pembelajaran Mendalam, yang juga dikenal sebagai MLP [8]. Susunan dari struktur MLP dijelaskan pada Gambar 5.

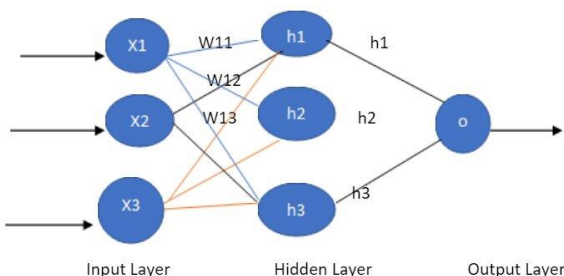


Diagram Of MultiLayer Perceptron Neural Network

Gambar 5. Arsitektur MLP

Dari Gambar 5 dijelaskan MLP memiliki lapisan input, lapisan tersembunyi, dan lapisan output. Lapisan input adalah lapisan awal atau permulaan dari Multilayer Perceptron. Lapisan ini mengambil input dari dataset pelatihan dan meneruskannya ke lapisan tersembunyi. Ada n node input di lapisan input. Jumlah node input bergantung pada jumlah fitur dalam dataset. Setiap variabel vektor input didistribusikan ke masing-masing node di lapisan tersembunyi. Kemudian lapisan tersembunyi adalah inti dari semua jaringan saraf tiruan. Lapisan ini mencakup semua perhitungan dari jaringan saraf. Tapi dari lapisan tersembunyi memiliki bobot yang dikalikan dengan nilai node. Lapisan ini menggunakan fungsi aktivasi. Kemudian yang terakhir adalah lapisan output, Dimana lapisan ini memberikan perkiraan output dari Jaringan Saraf. Jumlah node di lapisan output tergantung pada jenis masalahnya. Untuk satu variabel target, gunakan satu node. Pada masalah klasifikasi N , Jaringan Saraf Tiruan (ANN) menggunakan N node di lapisan output.

Dengan menggunakan berbagai algoritma ini, penelitian ini bertujuan untuk memprediksi harga laptop gaming dengan lebih akurat. LSTM akan menangani aspek temporal dari data harga historis, Random Forest dan Gradient Boosting akan menangani fitur yang beragam dengan kuat, dan MLP akan memodelkan hubungan non-linear yang kompleks dalam data. Kombinasi metode ini diharapkan dapat memberikan prediksi harga yang lebih akurat dan handal.

E. Evaluasi

Untuk membandingkan kinerja dari masing-masing metode yang diusulkan dalam penelitian ini, dilakukan evaluasi menggunakan beberapa metrik evaluasi yang umum digunakan dalam regresi, yaitu Root Mean Squared Error (RMSE), Mean Squared Error (MSE) dan Skor R^2 . Berikut adalah penjelasan masing-masing metrik:

1) *Root Mean Squared Error (RMSE)*: RMSE merupakan akar kuadrat dari nilai MSE. Dalam RMSE, kita mengakar kuadrat dari MSE karena kita ingin mengevaluasi kesalahan prediksi dengan skala yang sama dengan data yang diukur. RMSE digunakan untuk mengukur kesalahan prediksi dengan mengambil akar kuadrat dari selisih antara nilai prediksi dengan nilai aktual [9].

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \tag{8}$$

2) *Mean Squared Error (MSE)*: MSE adalah rata-rata dari selisih kuadrat antara nilai prediksi dengan nilai aktual. MSE digunakan untuk mengukur seberapa besar rata-rata kesalahan prediksi dari nilai aktual [9].

$$MSE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \tag{9}$$

3) *Skor R^2* : Skor R-squared mewakili proporsi varians variabel dependen yang dapat diprediksi dari variabel independen. Nilai 1 berarti seluruh variabilitas pada variabel dependen dijelaskan oleh variabel independen, sedangkan

nilai 0 menunjukkan bahwa variabel independen tidak menjelaskan satupun variabilitas [10].

$$R^2 = 1 - \frac{\text{Variasi yang tidak dapat dijelaskan}}{\text{Variasi total}} \quad (10)$$

Dengan menggunakan metrik-metrik ini, penelitian ini dapat mengevaluasi dan membandingkan kinerja dari algoritma LSTM, Linear Regression dan Multi-layer Perceptron dalam memprediksi harga laptop gaming. Hasil evaluasi ini akan membantu dalam menentukan metode mana yang memberikan prediksi paling akurat dan andal.

III. HASIL DAN PEMBAHASAN

Dalam penelitian ini terdapat beberapa proses penting, dimulai dari menganalisis data hingga melakukan evaluasi dan komparasi terhadap beberapa model yang diajukan. Berikut adalah penjelasan lengkapnya:

A. Data

Penelitian ini menggunakan dataset publik yang di ambil dari website *Kaggle* [6]. Dimana dataset tersebut berisi 646 baris dan 17 kolom. Contoh dari dataset yang saya pakai dapat dilihat pada Tabel I.

TABEL I
DATASET

No	Nama	Berat	Dimensi	...	Harga
1	RAZER BLADE ADVANCE	5 kg	40 × 20 × 10 cm		Rp94.799.000
2	MSI VECTOR	5 kg	40 × 20 × 10 cm		Rp47.499.000
3	LENOVO YOGA SLIM	5 kg	40 × 20 × 10 cm		Rp19.499.000
4	LENOVO LEGION 5I PRO	5 kg	40 × 20 × 10 cm		Rp18.199.000
5	ASUS TUF	5 kg	40 × 20 × 10 cm		Rp12.199.000

B. Preprocessing Data

Sebelum dibuat model, data dilakukan preprocessing agar data dapat digunakan untuk membangun model. Dalam penelitian ini dilakukan beberapa tahap preprocessing data dari mulai clean data hingga normalisasi data.

1) *Data Clean*: Langkah pertama adalah melakukan pembersihan data. Terlihat pada Tabel II kolom Keyboard dan Microsoft Office terdapat banyak data yang nilainya Null sehingga kedua kolom tersebut dihapus. Sedangkan untuk kolom Berat, Dimensi dan Warna dihapus karena dianggap kurang relevant untuk memprediksi harga laptop. Data tersebut dapat dilihat pada Tabel II.

TABEL II
PENGECEKAN KOLOM

No	Nama Kolom	Data Null
1	Nama	0
2	Berat	0
3	Dimensi	0
4	Garansi	4
5	VGA	0
6	Tipe Penyimpanan	0
7	Processor	0
8	Penyimpanan	0
9	Ukuran Layar	0
10	Warna	0
11	RAM	0
12	Keyboard	419
13	Microsoft Office	264
14	Sistem Operasi	0
15	Brand	0
16	Tipe Layar	0
17	Harga	0

2) *Feature Enggining*: Setelah data clean selesai kemudian dilakukan transformasi data yang dimana proses ini bertujuan untuk merubah data string menjadi data numerik agar semua data dapat digunakan untuk membangun model. Langkah pertama adalah menghilangkan “Rp” dan “.” pemisah angka ribuan di kolom Harga. Pada Tabel III menunjukkan kolom Harga sebelum dan sesudah dilakukan feature enggining. Hasil dari Feature Enggining dapat dilihat pada Tabel III.

TABEL III
HASIL FEATURE ENGGINER

Harga	
Sebelum	Sesudah
Rp94.799.000	94799000
Rp84.899.000	84899000
Rp66.999.000	66999000
Rp66.999.000	66999000
Rp51.999.000	51999000

Kemudian langkah selanjutnya adalah mengubah nilai dari kolom Ukuran Layar, RAM, dan Penyimpanan dari string menjadi numerik. Pada proses ini masih menggunakan feature enggining dengan menghilangkan “Inch” pada kolom Ukuran Layar, menghilangkan “GB” pada kolom RAM dan mengubah nilai TB menjadi dikali 1000 dan nilai GB tetap. Berikut adalah contoh hasil dari proses ini yang ditunjukkan pada Tabel IV yang merupakan data awal terlihat nilai setiap kolom masih berbentuk string/object kemudian setelah diproses menggunakan feature enggining data berubah menjadi numerik seperti pada Tabel V.

TABEL IV
DATA AWAL

Ukuran Layar	RAM	Penyimpanan
18 Inch	16GB	2TB
18 Inch	16GB	2TB
18 Inch	16GB	2TB
18 Inch	16GB	1TB
18 Inch	8GB	1TB

TABEL V
HASIL FEATURE ENGINERING

Ukuran Layar	RAM	Penyimpanan
18	16	2000
18	16	2000
18	16	2000
18	16	1000
18	8	1000

3) *Label Encoder*: Kemudian langkah selanjutnya adalah memeriksa semua kolom untuk mengetahui tipe data pada setiap kolom yang ada. Berikut adalah hasil dari pengecekan data yang dijelaskan pada Tabel VI.

TABEL VI
PENGECEKAN TIPE DATA

No	Nama Kolom	Type Data
1	Nama	object
2	Garansi	object
3	VGA	object
4	Tipe Penyimpanan	object
5	Processor	object
6	Penyimpanan	int64
7	Ukuran Layar	int64
8	RAM	int64
9	Sistem Operasi	object
10	Brand	object
11	Tipe Layar	object
12	Harga	int64

Setelah di analisis masih banyak kolom yang memiliki nilai object sehingga langkah selanjutnya adalah melakukan labelling menggunakan *Label Encoder*. Tabel VII menunjukkan hasil dari proses labelling data dimana nilai object diubah menjadi nilai numerik. Misalnya adalah kolom Tipe Penyimpanan, dimana nilai SSD diberi label 1 dan nilai HDD diberi label 0. Begitu juga dengan kolom Tipe Layar, laptop yang memiliki layar Non Touchscreen diberi label 0 dan layar Touchscreen diberi label 1. Berikut adalah contoh hasil dari proses Feature Engginering yang disajikan pada Tabel VII.

TABEL VII
CONTOH HASIL FEATURE ENGINERING

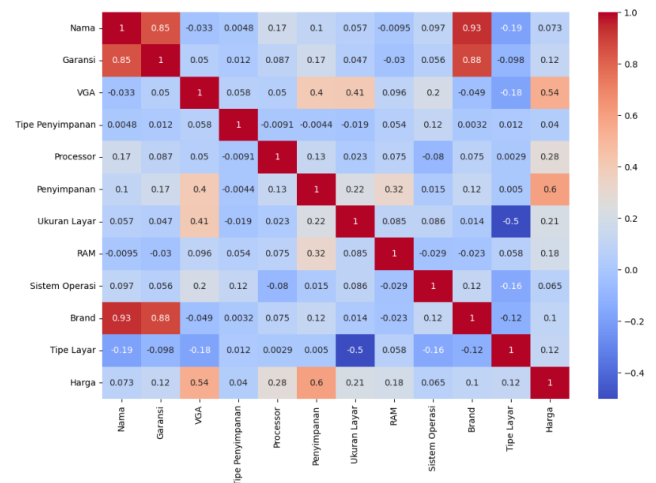
Tipe Penyimpanan	Processor	Sistem Operasi	Brand	Tipe Layar
1	6	2	8	0
1	6	2	8	0
1	6	2	8	0
1	6	2	8	0
1	6	2	8	0

4) *Normalisasi*: Langkah terakhir adalah melakukan normalisasi data menggunakan MinMaxScaler. Tujuannya adalah untuk menskalakan setiap fitur dalam dataset ke rentang 0 hingga 1. Normalisasi ini penting karena memastikan bahwa semua fitur memiliki kontribusi yang setara dalam proses pelatihan model, yang dapat meningkatkan kecepatan konvergensi algoritma optimasi berbasis gradien seperti yang digunakan dalam model LSTM, serta meningkatkan kinerja dan stabilitas model secara keseluruhan.

C. *Pemilihan dan Pembagian Data*

Setelah data selesai dilakukan preprocessing data, langkah selanjutnya adalah pemilihan fitur dan pembagian data. Berikut adalah penjelasannya:

1) *Pemilihan Data*: Dalam pemilihan data saya memakai korelasi untuk melakukan analisis korelasi pada dataframe untuk memahami hubungan antar fitur, khususnya dengan variabel target Harga. Pertama-tama menghitung dan menampilkan matriks korelasi, serta memvisualisasikannya menggunakan heatmap. Selanjutnya, adalah menghitung korelasi absolut setiap fitur dengan Harga dan mengurutkannya dari yang tertinggi hingga terendah. Fitur-fitur yang memiliki korelasi absolut lebih tinggi dari ambang batas 0.15 dipilih untuk analisis lebih lanjut, dan Harga ditambahkan ke daftar fitur terpilih jika belum termasuk. Hal ini membantu mengidentifikasi fitur-fitur yang paling berkaitan dengan Harga, yang dapat digunakan untuk membangun model prediktif yang lebih efektif. Berikut adalah hasilnya yang dijelaskan pada Gambar 6.



Gambar 6. Korelasi

Dari Gambar 6 diatas diperoleh data korelasi yang dijelaskan dalam Tabel VIII yang Dimana merupakan hasil dari korelasi antar fitur terutama terhadap fitur Harga. Seperti penjelasan tadi fitur-fitur yang memiliki korelasi absolut lebih tinggi dari ambang batas 0.15 dipilih untuk analisis lebih lanjut. Dari hasil korelasi tersebut diperoleh fitur Harga,

Penyimpanan, VGA, Processor, Ukuran Layar, dan RAM. Nilai korelasi antar fitur dapat dilihat pada Tabel VIII.

TABEL VIII
HASIL KORELASI

Fitur	Nilai Korelasi
Harga	1.000000
Penyimpanan	0.599989
VGA	0.543470
Processor	0.277496
Ukuran Layar	0.212018
RAM	0.184938
Garansi	0.118855
Tipe Layar	0.118040
Brand	0.099581
Nama	0.073203
Sistem Operasi	0.064932
Tipe Penyimpanan	0.040073

2) *Pembagian Data*: Setelah diperoleh fitur yang akan dipakai, langkah selanjutnya adalah pembagian data. Data yang telah diproses tadi dibagi menjadi dua bagian yaitu fitur (X) dan target (y). Fitur (X) diambil dari semua kolom kecuali kolom Harga, sedangkan target (y) diambil dari kolom Harga. Setelah itu, dataset dipisahkan menjadi data pelatihan (training) dan data uji (testing) dengan rasio 80:20. Data pelatihan akan digunakan untuk melatih model, sedangkan data uji akan digunakan untuk mengevaluasi kinerja model. Pembagian data dijelaskan pada Tabel IX.

TABEL IX
PEMBAGIAN DATA

Data Latih	Data Uji
80%	20%

D. Model dan Evaluasi

1) *LSTM (Long Short-Term Memory)*: Pertama, data X_train dan X_test diubah bentuknya agar sesuai dengan input yang diterima oleh LSTM, yakni dengan menambahkan dimensi waktu (time step) sebagai dimensi kedua. Kemudian, model LSTM dibangun menggunakan Keras Sequential API dengan dua lapis LSTM yang masing-masing memiliki 50-unit neuron, dimana lapisan pertama mengembalikan urutan (return_sequences=True) untuk digunakan oleh lapisan LSTM berikutnya. Lapisan dropout dengan tingkat dropout 0.2 digunakan untuk mengurangi overfitting, dan lapisan Dense terakhir dengan satu unit neuron digunakan untuk output regresi. Model di-compile dengan optimizer Adam dan fungsi loss mean squared error (MSE), kemudian dilatih menggunakan data X_train_lstm dan y_train selama 50 epoch dengan ukuran batch 32, dengan data validasi X_test_lstm dan y_test. Berikut adalah parameter yang digunakan seperti yang dijelaskan pada Tabel X.

TABEL X
PARAMETER MODEL LSTM

Parameter	Nilai
Neuron in Hidden Layers	50
Batch Size	32
Epochs	50
Loss Function	MSE
Optimizer	Adam

Selanjutnya dievaluasi menggunakan RMSE, MSE, dan Skor R2. LSTM dilatih dengan menggunakan backpropagation through time (BPTT). Parameter tuning dilakukan dengan grid search untuk menentukan jumlah neuron, batch size, learning rate, dan jumlah epoch yang optimal. Model ini diimplementasikan menggunakan library Keras dengan backend TensorFlow. Data dibagi menjadi data latih dan data uji dengan rasio 80:20, dan normalisasi dilakukan pada data fitur sebelum pelatihan. Grid search diterapkan pada kombinasi hyperparameter LSTM, dan cross-validation 5-fold digunakan untuk mengevaluasi performa setiap kombinasi. Metrik evaluasi seperti MSE dan RMSE digunakan untuk memilih hyperparameter yang memberikan performa terbaik sambil menghindari overfitting. Berikut adalah hasil evaluasinya yang dijelaskan pada Tabel XI.

TABEL XI
EVALUASI MODEL LSTM

Evaluasi	Hasil
RMSE	0.09011
MSE	0.00812
R-squared	0.90016

2) *Linear Regression*: Kemudian yang kedua, model Regresi Linier dibangun menggunakan objek `LinearRegression()` dari library scikit-learn, dan dilatih dengan menggunakan data X_train dan y_train. Selanjutnya, model digunakan untuk memprediksi nilai target untuk data uji X_test, dan hasil prediksi tersebut digunakan untuk menghitung nilai Root Mean Squared Error (RMSE), Mean Squared Error (MSE), dan koefisien determinasi (R-squared). Berikut adalah parameter yang digunakan yang dijelaskan pada Tabel XII.

TABEL XII
PARAMETER MODEL LINEAR REGRESSION

Parameter	Nilai
Intercept	True
Normalize	False
Fit Intercept	True

Selanjutnya karena Regresi linear tidak memerlukan tuning hyperparameter yang kompleks. Grid search tidak digunakan untuk regresi linear. Evaluasi model dilakukan menggunakan cross-validation 5-fold untuk memverifikasi performa model dan memastikan bahwa model tidak overfit. Namun, untuk memastikan hasil yang optimal, model ini dilatih menggunakan seluruh dataset dengan fitur yang telah diproses. Evaluasi dilakukan dengan memeriksa koefisien

determinasi (R^2) dan error metrik seperti Mean Squared Error (MSE) dan Root Mean Squared Error (RMSE). Berikut adalah hasil dari evaluasi model Linear Regression yang dijelaskan pada Tabel XIII.

TABEL XIII
EVALUASI MODEL LINEAR REGRESSION

Evalausi	Hasil
RMSE	0.09075
MSE	0.00823
R-squared	0.89873

3) *Multi-layer Perceptron (MLP)*: Kemudian yang terahir adalah membuat model Multi-layer Perceptron (MLP) untuk melakukan prediksi berdasarkan fitur yang diberikan. Model MLP dibangun menggunakan `MLPRegressor` dari library scikit-learn dengan konfigurasi satu lapisan tersembunyi (hidden layer) berukuran 100 neuron, menggunakan fungsi aktivasi ReLU, solver 'adam' untuk optimisasi parameter, maksimal iterasi sebanyak 200 kali, dan nilai seed random 42 untuk memastikan reproduktibilitas hasil. Setelah dilatih menggunakan data X_{train} dan y_{train} , model digunakan untuk memprediksi nilai target untuk data uji X_{test} . Berikut adalah parameter yang digunakan seperti yang dijelaskan pada Tabel XIV.

TABEL XIV
PARAMETER MODEL MLP

Parameter	Nilai
Hidden Layer Size	100
Activation Function	ReLU
Solver	Adam
Random State	42
Max Iterations	200

Grid search digunakan untuk mencari kombinasi terbaik dari hyperparameter di atas. Cross-validation 5-fold diterapkan pada setiap kombinasi untuk mengevaluasi kinerja model dan memilih kombinasi hyperparameter yang menghasilkan performa terbaik tanpa overfitting. Kombinasi terbaik yang ditemukan adalah dua lapisan tersembunyi dengan masing-masing 100 dan 50 neuron, menggunakan fungsi aktivasi ReLU dan solver Adam. Hasil prediksi tersebut dievaluasi menggunakan RMSE, MSE, dan R squared. Hasil dari evaluasi model MLP dapat dilihat pada Tabel XV.

TABEL XV
EVALUASI MODEL MLP

Evalausi	Hasil
RMSE	0.09891
MSE	0.00978
R-squared	0.87971

E. Komparasi Model

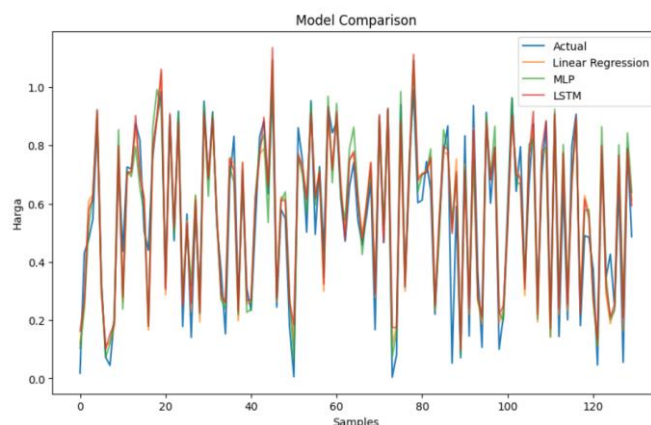
Setelah semua model dievaluasi selanjutnya adalah membandingkan hasil dari ketiga model yang sudah dibuat. Perbandingan hasil evaluasi dari ketiga model dapat dilihat pada Tabel XVI.

TABEL XVI
HASIL KOMPARASI MODEL

Evalausi	RMSE	MSE	R squared
LSTM	0.09011	0.00812	0.90016
Linear Regression	0.09075	0.00823	0.89873
MLP	0.09891	0.00978	0.87971

Dari hasil komparasi dapat disimpulkan bahwa model LSTM dan Regresi Linier memiliki performa prediksi yang hampir serupa, dengan nilai RMSE yang sangat mendekati satu sama lain, yaitu sekitar 0.09, serta nilai MSE dan R-squared yang juga mirip. Kedua model ini menunjukkan kemampuan yang baik dalam menghasilkan prediksi yang mendekati nilai sebenarnya dengan akurasi yang tinggi, sekitar 0.90 untuk R-squared. Di sisi lain, model Multi-layer Perceptron (MLP) memiliki nilai RMSE dan MSE yang sedikit lebih tinggi, menunjukkan bahwa prediksi dari model ini cenderung memiliki deviasi yang sedikit lebih besar dari nilai aktual, meskipun masih memberikan nilai R-squared yang cukup baik, sekitar 0.88. Dalam konteks ini, LSTM menjadi pilihan yang baik untuk tugas prediksi ini meski hanya selisih sedikit dengan Linear Regression.

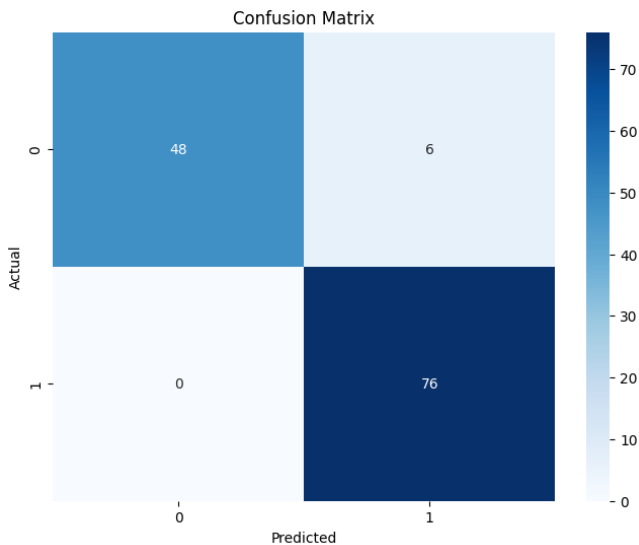
Kemudian berikut adalah grafik komparasi antara data actual dengan model LSTM, Linear Regression, dan MLP yang dijelaskan pada Gambar 7.



Gambar 7. Grafik Komparasi Model

Gambar 7 diatas menunjukkan perbandingan antara nilai actual dan prediksi dari ketiga model secara bersamaan dalam satu grafik garis. Ini memungkinkan untuk melihat bagaimana masing-masing model berperilaku terhadap data actual secara keseluruhan. Berdasarkan komparasi di atas dalam penelitian ini model terbaik adalah model LSTM sehingga selanjutnya adalah membuat confusion matrix. Langkah pertama adalah mengilustrasikan konversi output regresi menjadi klasifikasi biner dengan menggunakan thresholding. Output prediksi dari model LSTM (y_{pred_lstm}) diubah menjadi kelas biner dengan mengaplikasikan threshold 0.5. Jika nilai prediksi lebih besar dari threshold, dianggap sebagai kelas positif (1), dan jika tidak, dianggap sebagai kelas negatif (0). Hal yang sama dilakukan pada data uji (y_{test}), dimana nilai actual juga diubah menjadi kelas biner dengan threshold yang sama. Dari

kedua kelas tersebut, confusion matrix dihasilkan untuk menunjukkan jumlah prediksi yang tepat dan yang salah dari masing-masing kelas.



Gambar 8. Confusion Matrix

Gambar 8 menjelaskan bahwa Confusion matrix tersebut menunjukkan hasil dari suatu klasifikasi di mana model berhasil mengidentifikasi 48 data sebagai true negatives (TN), yaitu data yang benar-benar negatif dan diprediksi dengan benar sebagai negatif, serta 76 data sebagai true positives (TP), yang merupakan data yang benar-benar positif dan diprediksi dengan benar sebagai positif. Namun, terdapat 6 false positives (FP), di mana data sebenarnya negatif tetapi salah diprediksi sebagai positif, dan tidak ada false negatives (FN), yang berarti tidak ada data yang sebenarnya positif tetapi salah diprediksi sebagai negatif.

Selanjutnya dibawah ini merupakan hasil dari classification report yang memberikan detail performa model klasifikasi, termasuk precision, recall, dan F1-score untuk masing-masing kelas. Pendekatan ini bermanfaat dalam mengubah model regresi menjadi alat klasifikasi biner yang dapat digunakan untuk masalah-masalah seperti deteksi anomali atau pengambilan keputusan berbasis threshold tertentu yang disajikan pada Tabel XVII.

TABEL XVII
HASIL KOMPARASI MODEL

	precision	recall	f1-score	support
0	1.00	0.89	0.94	54
1	0.93	1.00	0.96	76
accuracy			0.95	130
macro avg	0.96	0.94	0.95	130
weigghted avg	0.96	0.95	0.95	130

Penelitian ini memiliki kelebihan dalam pendekatan metodologis yang komprehensif dengan membandingkan kinerja LSTM dengan model regresi klasik seperti Linear

Regression dan Multi-layer Perceptron (MLP), serta penggunaan data yang relevan dan aktual dari pasar laptop gaming di Indonesia. Model LSTM menunjukkan performa unggul dalam menangani data deret waktu dan pola kompleks, menghasilkan prediksi yang lebih akurat. Namun, penelitian ini juga memiliki beberapa kekurangan, seperti keterbatasan dataset yang hanya mencakup data dari satu toko online, serta kompleksitas model LSTM yang membutuhkan sumber daya komputasi lebih tinggi. Selain itu, beberapa faktor eksternal mungkin belum sepenuhnya diperhitungkan dalam model.

IV. KESIMPULAN

Penelitian ini berhasil mengembangkan model prediksi harga laptop gaming di Indonesia menggunakan algoritma Long Short-Term Memory (LSTM) dan membandingkan kinerjanya dengan model regresi klasik seperti Linear Regression dan Multi-layer Perceptron (MLP). Hasil menunjukkan bahwa model LSTM memiliki performa yang lebih baik dengan Root Mean Squared Error (RMSE) sebesar 0.090, Mean Squared Error (MSE) sebesar 0.008, dan Skor R² sebesar 0.900, dibandingkan model regresi klasik. Ini mengindikasikan bahwa LSTM lebih mampu menangkap pola-pola kompleks dalam data harga laptop gaming. Dengan mengadopsi model LSTM, prediksi harga menjadi lebih akurat, membantu konsumen membuat keputusan pembelian yang lebih tepat dan membantu produsen dalam menetapkan harga yang sesuai dengan pasar.

Penelitian ini juga menghadapi keterbatasan dalam ketergantungan pada kualitas dan kelengkapan dataset. Meskipun LSTM menunjukkan performa yang baik, model ini memerlukan waktu pelatihan yang lebih lama dan sumber daya komputasi yang lebih besar dibandingkan dengan model regresi klasik. Selain itu, hasil prediksi sangat tergantung pada variabel yang digunakan, sehingga variabel eksternal seperti perubahan kebijakan ekonomi atau teknologi baru yang tidak termasuk dalam dataset dapat mempengaruhi akurasi prediksi. Untuk penelitian selanjutnya, disarankan untuk memperluas dataset dengan mengumpulkan data dari berbagai sumber toko online dan offline, serta mempertimbangkan faktor eksternal seperti kebijakan impor dan fluktuasi nilai tukar mata uang. Selain itu, eksplorasi model machine learning lain yang mungkin lebih efisien dalam hal sumber daya komputasi dapat menjadi arah pengembangan yang berpotensi meningkatkan kinerja prediksi.

UCAPAN TERIMA KASIH

Penulis mengucapkan terimakasih kepada Program Studi Informatika Fakultas Ilmu Komputer Universitas Amikom Yogyakarta yang telah memberikan bimbingan dan dukungan dalam penelitian ini, sehingga penelitian ini dapat terselesaikan dengan baik.

DAFTAR PUSTAKA

- [1] B. Ayu Saputri, A. Samhudi, E. Zamilah, and U. Islam Kalimantan Muhammad Arsyad Al-Banjari, "Analisis Peningkatan dan Prediksi Penjualan Laptop Pada IT Komp Banjarbaru," 2021.
- [2] Aakarshachug, "What is LSTM – Long Short-Term Memory?" GeeksforGeeks. Accessed: Jun. 23, 2024. [Online]. Available: https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/?utm_source=auth&utm_medium=saved&utm_campaign=articles
- [3] E. Eli Lavindi and A. Rohmani, "Aplikasi Hybrid Filtering Dan Naïve Bayes Untuk Sistem Rekomendasi Pembelian Laptop Hybrid Filtering and Naïve Bayes Application for Laptop Purchase Recommendation Systems," *Journal of Information System*, vol. 4, no. 1, pp. 54–64, 2019.
- [4] M. A. Shaik, M. Varshith, S. Srivyshnavi, N. Sanjana, and R. Sujith, "Laptop Price Prediction using Machine Learning Algorithms," in *2022 International Conference on Emerging Trends in Engineering and Medical Sciences, ICETEMS 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 226–231. doi: 10.1109/ICETEMS56252.2022.10093357.
- [5] M. A. Sharma and M. S. Bhushan, "Laptop Price Prediction System Using Machine Learning," *JOURNAL OF APPLIED OPTICS*, 2024.
- [6] Fatur Febrianto M, "Gaming Laptop Dataset," Kaggle. Accessed: Jun. 23, 2024. [Online]. Available: <https://www.kaggle.com/datasets/faturfebr/gaming-laptop-dataset>
- [7] Gupta M, "Regresi Linier dalam Pembelajaran Mesin," GeeksforGeeks. Accessed: Jun. 23, 2024. [Online]. Available: <https://www.geeksforgeeks.org/ml-linear-regression/>
- [8] S. Sen, D. Sugiarto, and A. Rochman, "Komparasi Metode Multilayer Perceptron (MLP) dan Long Short Term Memory (LSTM) dalam Peramalan Harga Beras," *ULTIMATICS*, vol. XII, no. 1, p. 35, 2020.
- [9] Anugerah W, "Perbedaan RMSE dan MSE: Mengenal Metode Evaluasi Kinerja Model Regresi," Localstartupfest. Accessed: Jun. 24, 2024. [Online]. Available: <https://www.localstartupfest.id/faq/perbedaan-rmse-dan-mse/>
- [10] Fernando J, "R-Squared: Definition, Formula, Uses, and Limitations," Investopedia. Accessed: Jun. 24, 2024. [Online]. Available: <https://www.investopedia.com/terms/r/r-squared.asp>
- [11] T. B. Sianturi, I. Cholissodin, and N. Yudistira, "Penerapan Algoritma Long Short-Term Memory (LSTM) berbasis Multi Fungsi Aktivasi Terbobot dalam Prediksi Harga Ethereum," 2023. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [12] M. Rizki, S. Basuki, and Y. Azhar, "Implementasi Deep Learning Menggunakan Arsitektur Long Short Term Memory Untuk Prediksi Curah Hujan Kota Malang," *REPOSITOR*, vol. 2, no. 3, pp. 331–338, 2020.
- [13] A. A. Suryanto, A. Muqtadir, and S. Artikel, "Penerapan Metode Mean Absolute Error (MAE) dalam Algoritma Regresi Linear untuk Prediksi Produksi Padi Info Artikel : ABSTRAK," no. 1, p. 11, 2019.
- [14] Wijaya Mahendra A, "Analisis Pengaruh Brand Characteristics dan Brand Resonance Terhadap Brand Loyalty Laptop ASUS di Soloraya," 2022.
- [15] N. Erliani, K. Suryowati, M. T. Jatipaningrum, and J. Statistika, "Klasifikasi Tingkat Penjualan Laptop di e-Commerce Menggunakan Algoritma Classification and Regression Tree(CART)," *Jurnal Statistika Industri dan Komputasi*, vol. 08, no. 2, pp. 40–47, 2023. [Online]. Available: <https://www.tokopedia.com>
- [16] T. O. Hodson, "Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not," *Geoscientific Model Development*, vol. 15, no. 14. Copernicus GmbH, pp. 5481–5487, Jul. 19, 2022. doi: 10.5194/gmd-15-5481-2022.
- [17] A. Samii, H. Karami, H. Ghazvinian, A. Safari, and Y. D. Ajirlou, "Comparison of DEEP-LSTM and MLP Models in Estimation of Evaporation Pan for Arid Regions," *Journal of Soft Computing in Civil Engineering*, vol. 7, no. 2, pp. 155–175, Apr. 2023, doi: 10.22115/SCCE.2023.367948.1550.
- [18] C. A. Rahardja and H. Agung, "Rahardja, Implementasi Algoritma K-Nearest Neighbor Pada Website Rekomendasi Laptop 75 Implementasi Algoritma K-Nearest Neighbor Pada Website Rekomendasi Laptop," 2019.
- [19] C. Puspa Tria, A. Nuryaman, A. Faisol, dan Eri Setiawan, J. Soemantri Brojonegoro No, and B. Lampung, "Penerapan Algoritma K-Nearest Neighbor Pada Data Kategorik Untuk Klasifikasi Harga Jual Laptop," 2023.
- [20] Herwinsyah, "Penerapan Fuzzy Inference System (FIS) Dengan Metode Mamdani Pada Sistem Prediksi Penjualan Laptop Implementation of Fuzzy Inference System (FIS) with the Mamdani Method in Laptop Sales Prediction System," 2019.
- [21] H. W. Fondy, M. Fajar, and I. Alwiah Musdar, "Jurnal Ilmu Komputer KHARISMA TECH Implementasi Teori Support Vecto R Machine Untuk Memprediksi Harga Penjualan Laptop ASUS," 2019.