

## **IMPLEMENTASI ALGORITMA LSTM UNTUK PREDIKSI KEBUTUHAN BAHAN BAKU RESTORAN DI BALE RAOS KRATON YOGYAKARTA**

### **IMPLEMENTATION OF LSTM ALGORITHM FOR RESTAURANT RAW MATERIAL DEMAND PREDICTION AT BALE RAOS KRATON YOGYAKARTA**

**Abdul Rahman Wahid<sup>1</sup>, Arita Witanti<sup>2</sup>**

Universitas Mercu Buana Yogyakarta<sup>1,2</sup>

211110108@student.mercubuana-yogya.ac.id<sup>1</sup>, arita@mercubuana-yogya.ac.id<sup>2</sup>

#### **ABSTRACT**

*Bale Raos Kraton Yogyakarta faces challenges in raw material inventory management due to demand fluctuations. This study develops an LSTM (Long Short-Term Memory) model to forecast 36 raw materials using historical sales and seasonal calendar data (January 2023–February 2025). Preprocessing includes temporal interpolation, lagging features (1-day/7-day), one-hot encoding, and MinMax normalization. A layered LSTM architecture (256/128 units) was constructed and evaluated through data splits: training (January 2023–October 2024), validation (November–December 2024), and testing (January–February 2025). Results show optimal performance with MSE 0.0108 and MAE 0.0735. A 31-day simulation (January 29–February 28, 2025) achieved Overall Aggregated Accuracy (Macro) 90.27%, demonstrating operational effectiveness in minimizing overstock and stockout risks.*

**Keywords:** *LSTM, Raw Material Prediction, Inventory Management, Heritage Restaurant, Kraton Yogyakarta.*

#### **ABSTRAK**

Bale Raos Kraton Yogyakarta menghadapi tantangan dalam manajemen persediaan bahan baku akibat fluktuasi permintaan. Penelitian ini mengembangkan model *Long Short-Term Memory* (LSTM) untuk memprediksi kebutuhan 36 bahan baku berbasis data historis penjualan dan kalender musiman (Januari 2023–Februari 2025). Tahap *preprocessing* mencakup interpolasi temporal, pembentukan fitur *lagging* (1-hari dan 7-hari), *one-hot encoding*, dan normalisasi MinMax. Arsitektur LSTM berlapis (256/128 unit) dibangun, kemudian dievaluasi dengan pembagian data: pelatihan (Januari 2023–Oktober 2024), validasi (November–Desember 2024), dan pengujian (Januari–Februari 2025). Hasil menunjukkan kinerja optimal dengan MSE 0.0108 dan MAE 0.0735. Simulasi prediksi 31 hari (29 Januari–28 Februari 2025) mencapai *Overall Aggregated Accuracy* (Makro) 90,27%, membuktikan efektivitas model dalam meminimalkan risiko *overstock* dan *stockout* secara operasional.

**Kata Kunci:** LSTM, Prediksi Bahan Baku, Manajemen Persediaan, Restoran Tradisional, Kraton Yogyakarta.

#### **PENDAHULUAN**

Manajemen persediaan bahan baku yang akurat merupakan komponen penting dalam operasional industri restoran, terutama untuk kasus seperti Bale Raos Kraton Yogyakarta yang mengelola bahan baku rentan fluktuasi, termasuk komponen *perishable*. Fluktuasi permintaan harian, dipengaruhi faktor temporal seperti hari kerja, akhir pekan, libur nasional, dan event khusus, menciptakan tantangan kompleks dalam menjaga keseimbangan stok. Ketidakpresisian prediksi berpotensi memicu *overstock* yang berujung pada

pemborosan biaya dan kerusakan bahan, atau *stockout* yang mengganggu kontinuitas layanan.

Studi Rodrigues et al. (2023) pada layanan katering membuktikan bahwa metode berbasis *machine learning* mampu mengurangi *food waste* sebesar 14-52% dan permintaan yang tidak terpenuhi (*unmet demand*) sebesar 3-16% dibandingkan dengan metode konvensional. Implementasi solusi prediktif yang memadukan data historis dan variabel musiman tidak hanya meminimalkan risiko tersebut, tetapi juga

mengoptimalkan *working capital* dan kepuasan pelanggan.

Studi-studi terkini telah membuktikan efektivitas *Long Short-Term Memory* (LSTM) dalam peramalan permintaan sektor kuliner. Nassibi et al. (2023) menguji LSTM dan SVM pada distribusi produk kembang gula di Arab Saudi, menunjukkan keunggulan LSTM meski tipis dalam akurasi peramalan triwulanan. Panda dan Mohanty (2023) memperluas evaluasi dengan membandingkan tujuh model termasuk LSTM/BiLSTM untuk permintaan makanan.

Dalam studi Panda dan Mohanty (2023), LSTM mencapai MAPE terendah (6,56%) namun menghadapi keterbatasan tanpa variabel temporal spesifik. Di tingkat operasional, Schmidt et al. (2022) mengonfirmasi superioritas LSTM dalam peramalan penjualan restoran (sMAPE 19,5%), tetapi hanya berfokus pada pendapatan bukan kebutuhan bahan baku. Haque (2023) berinovasi dengan mengintegrasikan variabel makroekonomi ke LSTM, berhasil menekan RMSE hingga 1,60 pada data ritel. Penelitian Rodrigues et al. (2023) menegaskan potensi reduksi *food waste* hingga 52% melalui LSTM di layanan katering, dengan catatan efektivitas bergantung pada karakteristik data.

Meskipun berbagai penelitian telah mendemonstrasikan keunggulan LSTM dalam peramalan permintaan sektor kuliner, fokus utama masih terbatas pada prediksi penjualan produk jadi atau porsi makanan. Penelitian ini mengisi celah akademik dengan beralih fokus ke prediksi kebutuhan bahan baku operasional, mengintegrasikan variabel temporal spesifik yang belum dipertimbangkan secara memadai pada studi sebelumnya, serta merancang arsitektur LSTM yang disesuaikan untuk kebutuhan restoran. Oleh karena itu, penelitian ini bertujuan memprediksi kebutuhan 36 bahan baku restoran menggunakan model LSTM berbasis integrasi data historis penjualan dan variabel temporal yang belum

terakomodasi dalam pendekatan konvensional. Tujuannya meningkatkan efisiensi manajemen persediaan melalui minimisasi risiko *overstock* dan *stockout*.

*Recurrent Neural Network* (RNN) konvensional memiliki keterbatasan dalam mempelajari *long-term dependencies*. Hal ini disebabkan oleh masalah *vanishing gradient* atau *exploding gradient* yang dapat terjadi selama proses pelatihan melalui *backpropagation through time* (BPTT), sebagaimana dianalisis oleh Bengio, Simard, & Frasconi (1994). Keterbatasan inilah yang mendorong pengembangan arsitektur RNN yang lebih canggih seperti LSTM. LSTM adalah arsitektur RNN khusus yang dikembangkan oleh Hochreiter & Schmidhuber (1997) untuk mengatasi masalah kesulitan dalam mempelajari *long-term dependencies* yang sering ditemui pada RNN konvensional. Mekanisme *gate* (gerbang lupa, masukan, keluaran) dan *cell state*-nya secara selektif menyimpan pola temporal relevan (seperti pengaruh libur nasional atau tren mingguan) sambil mengabaikan noise. Keunggulan ini terutama efektif untuk data deret waktu non-linear dan kompleks seperti kebutuhan bahan baku restoran, di mana fluktuasi harian tidak hanya bergantung pada hari sebelumnya, tetapi juga pada pola periodik dan event khusus.

## METODE

Penelitian ini mengembangkan model prediksi kebutuhan bahan baku berbasis LSTM untuk mengoptimalkan manajemen persediaan di Bale Raos Kraton Yogyakarta. Pendekatan diawali dengan pengumpulan data historis penjualan harian, kebutuhan 36 bahan baku, serta kalender musiman selama periode Januari 2023 hingga Februari 2025.

Tahap metodologis mencakup *preprocessing* data untuk mengatasi *missing values* dan ketidakkonsistenan, konstruksi fitur temporal untuk menangkap pola harian dan mingguan, serta transformasi data menggunakan

teknik normalisasi dan *encoding*. Selanjutnya dirancang arsitektur LSTM berlapis untuk memodelkan ketergantungan jangka panjang dalam data deret waktu. Kinerja model dievaluasi menggunakan metrik yang relevan dengan konteks operasional restoran. Implementasi teknis memanfaatkan Python dengan pustaka Keras/TensorFlow untuk pemodelan dan Scikit-learn untuk transformasi data.

### Pengumpulan Data

Data penelitian dikumpulkan dari sistem informasi perusahaan selama periode Januari 2023 hingga Februari 2025, mencakup tiga sumber utama, data penjualan harian dari sistem *Point-of-Sale* (POS) dalam format Excel, data kebutuhan bahan baku dari sistem *Enterprise Resource Planning* (ERP) dalam format CSV, dan data kalender musiman yang dikompilasi secara manual berdasarkan operasional restoran.

**Tabel 1. Sumber Data Penelitian**

| Sumber Data          | Format | Periode             |
|----------------------|--------|---------------------|
| Penjualan Harian     | Excel  | Jan 2023 - Feb 2025 |
| Kebutuhan Bahan Baku | CSV    | Jan 2023 - Feb 2025 |
| Kalender Musiman     | Manual | Jan 2023 - Feb 2025 |

Data mencakup 36 bahan baku termasuk bahan *perishable* (Lamb Leg, Ikan Nila) dan spesifik (Sus Ragout, Bumbu Blawong) yang menjadi komponen menu utama seperti Kambing Panggang, Mangut Ikan, dan Songgo Buwono.

### Preprocessing Data

Dalam penelitian ini, *preprocessing* data dilakukan melalui serangkaian tahapan transformasi sistematis untuk menyiapkan data mentah menjadi format yang sesuai untuk pemodelan LSTM. Tahapan ini meliputi pembersihan data, konstruksi fitur, pembagian dataset secara kronologis, normalisasi numerik, dan transformasi akhir menjadi sekuens tiga

dimensi, yang bertujuan meningkatkan kualitas dan relevansi data untuk *machine learning* sekaligus mencegah *data leakage*.

#### 1. Pembersihan Data

Tahap pembersihan data diawali dengan integrasi data kebutuhan bahan baku per menu (resep) dan data penjualan harian untuk menghitung kebutuhan harian setiap bahan baku. Perhitungan dilakukan dengan menjumlahkan hasil perkalian antara kebutuhan per porsi (dalam gram) dengan jumlah porsi terjual untuk setiap menu yang menggunakan bahan baku tersebut, dirumuskan sebagai:

$$\text{Kebutuhan} = \Sigma(\text{Resep} \times \text{Penjualan})$$

Nilai nol yang muncul pada hari operasional restoran diidentifikasi sebagai ketidakakuratan data dan dikonversi menjadi nilai NaN (*Not a Number*). Nilai NaN tersebut kemudian diisi menggunakan metode interpolasi berbasis waktu yang mempertimbangkan pola historis setiap bahan baku. Pendekatan ini dipilih karena kemampuannya mempertimbangkan konteks temporal, yang krusial untuk menjaga integritas pola data deret waktu (Niako et al., 2024). Sebagai ilustrasi, kebutuhan Kelengkeng pada tanggal 11 Januari 2023 yang semula tercatat 0 gram, setelah interpolasi menjadi 126 gram berdasarkan pola kebutuhan di tanggal terdekat (140 gram pada 10 Januari dan 112 gram pada 12 Januari 2023). Proses ini menjamin kontinuitas pola temporal sekaligus mengatasi anomali pencatatan operasional restoran.

#### 2. Pembentukan Fitur

Pembentukan fitur dilakukan untuk memperkaya model dengan informasi yang relevan dan meningkatkan kemampuannya dalam mengenali pola kompleks serta memperbaiki akurasi prediksi (Janiesch et al., 2021). Proses ini mencakup konstruksi fitur *lagging* untuk merepresentasikan ketergantungan temporal, di mana pemilihan nilai *time-lag* yang tepat merupakan langkah krusial karena secara signifikan memengaruhi

akurasi prediksi model (Surakhi et al., 2021). Fitur lagging meliputi kebutuhan 1 hari sebelumnya ( $\{bahan\}_{lag\_1}$ ) dan 7 hari sebelumnya ( $\{bahan\}_{lag\_7}$ ) untuk setiap bahan baku. Selain itu, dilakukan pengintegrasian variabel musiman melalui teknik *one-hot encoding* pada 28 kategori event khusus (seperti Tahun Baru dan Ramadan) serta faktor musiman (Puncak, Tinggi, Normal, Rendah), sementara variabel operasional seperti status buka/tutup restoran dan pola hari dalam minggu dipertahankan dalam format biner.

**Tabel 2. Kategori Fitur yang Dikonstruksi**

| Kategori Fitur     | Contoh Fitur             | Tipe Data | Dimensi |
|--------------------|--------------------------|-----------|---------|
| Historis (Lagging) | Ayam_Dadu_lag_1          | Numerik   | 72      |
|                    | Ayam_Dadu_lag_7          |           |         |
| Kalender           | Is_Senin, Libur_Nasional | Biner     | 11      |
| Event Khusus (OHE) | Event_Khusus_Ramadan     | Biner     | 28      |
| Musiman (OHE)      | Faktor_Musim_Puncak      | Biner     | 4       |
| Operasional        | Resto_Buka               | Biner     | 1       |

Total fitur setelah transformasi mencapai 115 dimensi, menggabungkan 72 fitur *lagging* ( $36 \text{ bahan baku} \times 2$ ) dengan 43 fitur kontekstual. Keseluruhan fitur membentuk representasi multidimensional yang menangkap dinamika historis kebutuhan bahan baku, konteks temporal, dan kondisi operasional restoran, menyediakan basis input komprehensif untuk model LSTM.

### 3. Pembagian Data

Pembagian dataset dilakukan secara kronologis untuk mempertahankan integritas temporal data deret waktu dan mencegah *data leakage*, di mana informasi dari masa depan dapat bocor ke dalam data pelatihan (Albeladi et al., 2023). Dengan pendekatan ini, model dilatih menggunakan data dari masa lalu dan diuji kemampuannya untuk memprediksi data di masa depan yang belum pernah dilihat

sebelumnya. Proses ini mengikuti urutan waktu dengan konfigurasi sebagai berikut:

**Tabel 3. Pembagian Dataset secara Kronologis**

| Set Data   | Periode             | Jumlah Baris | Persentase |
|------------|---------------------|--------------|------------|
| Training   | Jan 2023 - Okt 2024 | 663          | 84.55%     |
| Validation | Nov 2024 - Des 2024 | 61           | 7.78%      |
| Testing    | Jan 2025 - Feb 2025 | 59           | 7.67%      |

Pembagian ini menghindari praktik *processing* seluruh data sebelum pemisahan, yang dapat menyebabkan data dari set pengujian terserap ke dalam proses pelatihan dan menghasilkan evaluasi model yang menunjukkan kinerja *false high* yang tidak merefleksikan kemampuan sebenarnya (Liu et al., 2022). Validasi dilakukan pada November-Desember 2024 untuk simulasi prediksi jangka pendek menjelang tahun baru, sementara *testing* menggunakan data Januari-Februari 2025 yang merepresentasikan pola permintaan aktual terkini.

### 4. Normalisasi Data

Normalisasi data penting karena model neural network, termasuk LSTM, dapat mengalami kesulitan konvergensi jika input memiliki skala yang sangat berbeda (Pranolo et al., 2024). Transformasi numerik menggunakan MinMaxScaler (rentang 0-1) diterapkan dengan strategi khusus untuk mencegah kebocoran data, parameter scaling dihitung eksklusif dari subset *training*, kemudian diaplikasikan secara konsisten ke data *validation* dan *testing* (Liu et al., 2022). Fitur hasil *one-hot encoding* dan variabel biner seperti status operasional restoran dikecualikan dari normalisasi karena secara intrinsik telah berada dalam rentang [0,1] yang kompatibel (Pranolo et al., 2024). Proses normalisasi dilakukan terpisah untuk fitur input dan variabel target guna menjaga interpretabilitas satuan fisik, dengan pilihan MinMaxScaler didukung oleh studi yang membuktikan efektivitasnya dalam prediksi deret waktu (Abdullah et al., 2025).

**Tabel 4. Strategi Normalisasi Data**

| Komponen               | Metode              | Rentang |
|------------------------|---------------------|---------|
| Fitur Numerik          | MinMaxScaler        | [0, 1]  |
| Target (36 bahan baku) | MinMaxScaler        | [0, 1]  |
| Fitur Biner & OHE      | Tidak dinormalisasi | -       |

### 5. Pembentukan Sekuens

Model LSTM memproses data dalam bentuk sekuens historis untuk menangkap dependensi temporal kompleks dalam deret waktu, termasuk siklus harian dan tren musiman (Guo et al., 2024). Tahap ini mentransformasikan data menjadi format tiga dimensi dengan panjang sekuens tetap 28 hari (*sequence\_length=28*), di mana setiap sampel merepresentasikan pola historis 4 minggu untuk memprediksi kebutuhan 36 bahan baku pada hari berikutnya (*prediction\_horizon=1*). Pemilihan panjang sekuens ini merupakan pertimbangan penting karena panjang sekuens terlalu pendek dapat membatasi konteks temporal, sementara terlalu panjang meningkatkan kompleksitas komputasi dan risiko *overfitting* (Guo et al., 2024).

Setiap sampel input (X) terdiri dari matriks  $28 \times 115$  yang berisi kebutuhan historis bahan baku, fitur *lagging* (1-hari dan 7-hari sebelumnya), variabel kalender dan musiman, serta status operasional restoran. Sedangkan output target (y) merupakan vektor kebutuhan 36 bahan baku pada hari ke-29. Struktur sekuens ini mengimplementasikan mekanisme *gate* LSTM yang secara selektif menyimpan informasi relevan dari urutan panjang (Siami-Namini et al., 2019), memungkinkan model belajar pola temporal multidimensi yang mencakup dinamika operasional restoran.

**Tabel 5. Dimensi Data Sekuens Akhir**

| Set Data   | Dimensi X                  | Dimensi y       |
|------------|----------------------------|-----------------|
| Training   | $635 \times 28 \times 115$ | $635 \times 36$ |
| Validation | $33 \times 28 \times 115$  | $33 \times 36$  |
| Testing    | $31 \times 28 \times 115$  | $31 \times 36$  |

## Desain Model

Desain model menggunakan arsitektur LSTM untuk memprediksi kebutuhan 36 bahan baku secara simultan berdasarkan pola temporal multidimensi dari data historis yang telah diproses. Model ini memanfaatkan kemampuan LSTM dalam menangkap ketergantungan *long-term dependencies* pada data deret waktu, mengubah input sekuensial 28 hari menjadi prediksi harian untuk seluruh bahan baku.

### 1. Arsitektur Model

Model dirancang dengan arsitektur berlapis untuk memproses data sekuensial kebutuhan bahan baku. Lapisan pertama adalah LSTM dengan 128 unit yang mengembalikan seluruh sekuens (*return\_sequences=True*) untuk mengekstrak pola temporal pada setiap langkah waktu. Keluaran lapisan ini kemudian diikuti oleh Dropout (*rate=0.2*) untuk mengurangi *overfitting*. Selanjutnya, lapisan LSTM kedua (64 unit) mengompres informasi temporal dan hanya mengembalikan output pada langkah akhir (*return\_sequences=False*). Lapisan Dropout kedua kembali diterapkan untuk regularisasi. Akhirnya, lapisan Dense linear (36 unit) menghasilkan prediksi kebutuhan harian untuk 36 bahan baku.

**Tabel 6. Arsitektur Model**

| Lapisan   | Unit | Konfigurasi                   | Output Shape |
|-----------|------|-------------------------------|--------------|
| Input     | -    | -                             | (28, 115)    |
| LSTM-1    | 128  | <i>return_sequences=True</i>  | (28, 128)    |
| Dropout-1 | -    | <i>rate=0.2</i>               | (28, 128)    |
| LSTM-2    | 64   | <i>return_sequences=False</i> | (64)         |
| Dropout-2 | -    | <i>rate=0.2</i>               | (64)         |
| Dense     | 36   | <i>activation='linear'</i>    | (36)         |

Dropout rate 0.2 diterapkan setelah setiap lapisan LSTM untuk mitigasi *overfitting* tanpa mengorbankan kapasitas pembelajaran (Srivastava et al., 2014). Arsitektur ini dirancang untuk menangkap ketergantungan jangka panjang dalam data

deret waktu kebutuhan bahan baku, mengacu pada kemampuan LSTM dalam menyimpan informasi relevan melalui mekanisme *gate* dan *cell state* (Hochreiter & Schmidhuber, 1997).

## 2. Konfigurasi Pelatihan

Model dilatih dengan optimizer Adam (*learning rate* 0.0005) untuk efisiensi komputasi dan kemampuan adaptifnya dalam menangani pola data kompleks (Goodfellow et al., 2016). Fungsi *loss* Mean Squared Error (MSE) dipilih karena sensitivitasnya terhadap kesalahan besar. Evaluasi kinerja menggunakan metrik Mean Absolute Error (MAE) untuk interpretasi kesalahan dalam satuan fisik. Pelatihan dilakukan dengan batch size 64 dan maksimal 200 epoch, didukung mekanisme EarlyStopping (*patience*=20) yang menghentikan pelatihan jika validasi loss tidak membaik (Srivastava et al., 2014). Dropout rate 0.2 diterapkan setelah setiap lapisan LSTM untuk mitigasi *overfitting*. Proses ini diimplementasikan menggunakan callback ModelCheckpoint guna menyimpan bobot model terbaik secara otomatis.

**Tabel 7. Konfigurasi Hyperparameter Pelatihan**

| Parameter     | Nilai           | Peran Utama                        |
|---------------|-----------------|------------------------------------|
| Optimizer     | Adam            | Efisiensi adaptasi gradien         |
| Learning Rate | 0.0005          | Kecepatan pembaruan bobot          |
| Loss Function | MSE             | Penalti kesalahan besar            |
| Metrics       | MAE             | Interpretasi langsung              |
| Batch Size    | 64              | Stabilisasi pelatihan              |
| Max Epochs    | 200             | Iterasi maksimal                   |
| EarlyStopping | Patience=20     | Penghentian dini tanpa peningkatan |
| Dropout Rate  | 0.2             | Mitigasi overfitting               |
| Callback      | ModelCheckpoint | Penyimpanan bobot optimal          |

## Evaluasi Model

Evaluasi kinerja model dilakukan menggunakan MSE dan MAE. Pemilihan metrik ini didasarkan pada relevansinya dengan konteks operasional restoran.

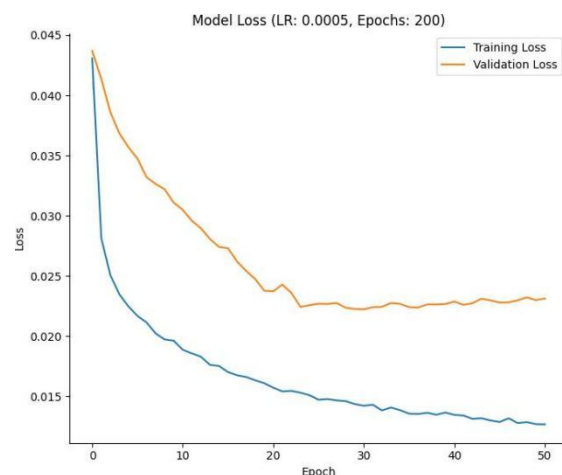
Proses evaluasi dilakukan pada data pengujian (*testing set*) yang tidak digunakan selama pelatihan model, yaitu periode Januari hingga Februari 2025.

### 1. Mean Squared Error (MSE)

MSE merupakan metrik yang mengukur rata-rata kuadrat selisih antara nilai prediksi  $\hat{y}_i$  dan nilai aktual  $y_i$  untuk seluruh  $N$  sampel data pengujian, dengan rumus matematis:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

Metrik ini secara khusus memberikan penalti lebih berat pada kesalahan prediksi yang besar, sehingga sangat sensitif terhadap potensi risiko operasional kritis seperti *stockout* bahan baku esensial atau *overstock* bahan baku yang mudah rusak. Nilai MSE 0.0113 yang diperoleh dari evaluasi model merefleksikan tingkat akurasi prediksi yang baik.



**Gambar 1. MSE Konfigurasi Base**

Grafik pelatihan dan validasi MSE (Gambar 1) menunjukkan dinamika pembelajaran model selama 50 epoch pertama dari total 200 epoch. *Training loss* dimulai pada 0.45, mengalami penurunan signifikan menjadi 0.20 dalam 10 epoch pertama, dan akhirnya stabil di sekitar 0.015 pada epoch ke-50. Sementara itu, *validation loss* menunjukkan pola serupa dengan inisiasi pada 0.40, penurunan ke 0.25 di epoch 10, dan stabilisasi di 0.22

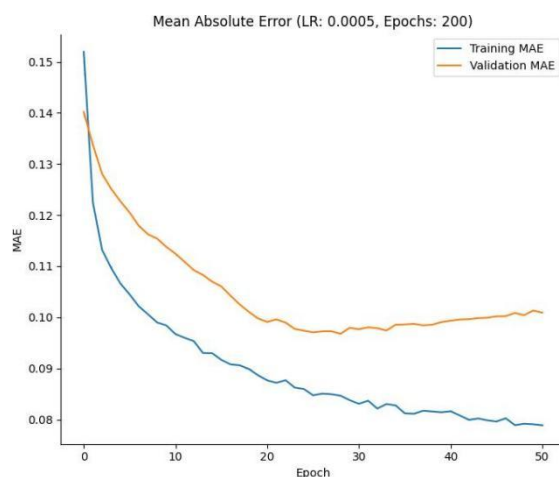
hingga epoch 50. Meskipun teridentifikasi sedikit indikasi *overfitting* (*training loss* lebih rendah dari *validation loss*), hasil akhir pada data *testing* (MSE 0.0113) justru lebih baik daripada *training loss* di epoch 50, mengindikasikan kemampuan generalisasi model yang terus meningkat selama pelatihan penuh.

## 2. Mean Absolute Error (MAE)

MAE menghitung rata-rata selisih absolut antara nilai prediksi dan nilai aktual melalui rumus:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|$$

Metrik ini dapat memberikan interpretasi langsung dalam satuan fisik. Nilai MAE 0.0755 menunjukkan tingkat presisi yang baik dalam memprediksi kebutuhan harian bahan baku.



Gambar 2. MAE Konfigurasi Base

Grafik perkembangan MAE (Gambar 2) mengungkap konsistensi pembelajaran model. *Training MAE* dimulai dari 0.15, turun cepat ke 0.10 pada epoch 10, dan mencapai 0.08 di epoch 50. *Validation MAE* menunjukkan pola komplementer dengan inisiasi di 0.14, penurunan ke 0.11 di epoch 10, dan stabilisasi di 0.10. Meski terdapat gap kecil antara *training* dan *validation MAE* yang mengindikasikan *overfitting* ringan, nilai akhir pada data *testing* (0.0755) justru lebih rendah daripada *training MAE* di epoch 50.

Fenomena ini mengonfirmasi kapasitas *generalization* model yang baik, terutama karena proses pelatihan berlangsung penuh hingga 200 epoch dengan mekanisme *early stopping* yang optimal.

## HASIL DAN PEMBAHASAN

Model prediksi berbasis LSTM telah berhasil dirancang untuk mengestimasi kebutuhan harian 36 bahan baku. Hasil evaluasi menunjukkan kemampuan model dalam menangkap pola fluktuasi permintaan dengan baik. Pada bagian ini, temuan kunci dipaparkan melalui analisis contoh prediksi, performa model, dan prediksi iteratif.

### 1. Contoh Prediksi

Sebagai sampel representatif, Tabel 8 menyajikan perbandingan prediksi dengan nilai aktual untuk lima bahan baku pertama pada tanggal 31 Januari 2025. Hasil ini dihasilkan oleh model LSTM konfigurasi base (128/64 unit).

Tabel 8. Perbandingan Prediksi dan Aktual

| Bahan Baku      | Prediksi | Aktual  | Selisih (Persentase) |
|-----------------|----------|---------|----------------------|
| Ayam Dadu       | 319.61   | 275.00  | +44.61 (+16.22%)     |
| Ayam Kampung    | 2981.89  | 3200.00 | -218.11 (-6.82%)     |
| Ayam Pedaging   | 2618.06  | 3135.00 | -516.94 (-16.49%)    |
| Bayam           | 268.38   | 240.00  | +28.38 (+11.82%)     |
| Belimbing Wuluh | 105.60   | 127.00  | -21.40 (-16.85%)     |

Meski terdapat selisih, model dapat menghasilkan akurasi dengan baik untuk sampel ini. Analisis lebih lanjut terhadap selisih ini akan membantu dalam mengidentifikasi bahan baku mana yang prediksinya sudah baik dan mana yang memerlukan perbaikan.

### 2. Performa Model

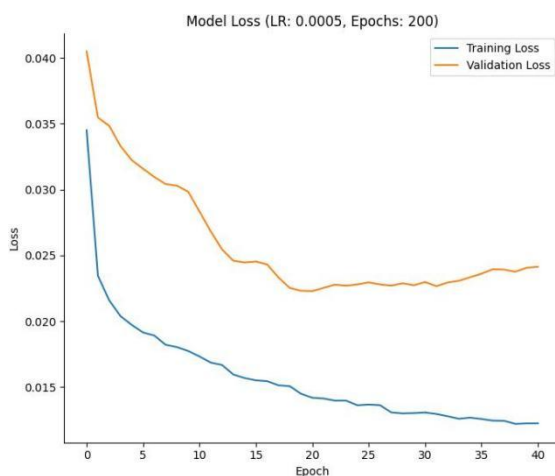
Untuk mengidentifikasi konfigurasi optimal, dilakukan evaluasi komparatif terhadap sembilan variasi *hyperparameter*. Tabel 9 menyajikan hasil uji coba dengan metrik MAE dan MSE pada data *testing*. Konfigurasi base dibandingkan dengan

variasi ukuran lapisan, *learning rate*, *dropout rate*, dan *batch size*.

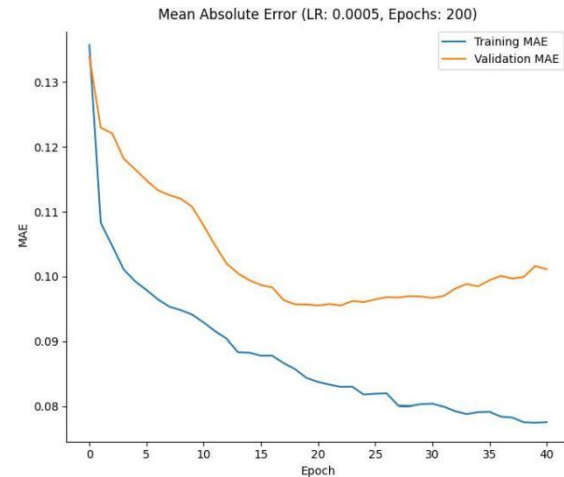
**Tabel 9. Perbandingan Berbagai Konfigurasi**

| Config           | L1  | L2  | LR     | BS  | D   | MAE    | MSE    |
|------------------|-----|-----|--------|-----|-----|--------|--------|
| Base             | 128 | 64  | 0.0005 | 64  | 0.2 | 0.0755 | 0.0113 |
| Unit Kecil       | 64  | 32  | 0.0005 | 64  | 0.2 | 0.0796 | 0.0126 |
| Unit Besar       | 256 | 128 | 0.0005 | 64  | 0.2 | 0.0735 | 0.0108 |
| LR Tinggi        | 128 | 64  | 0.0010 | 64  | 0.2 | 0.0768 | 0.0113 |
| LR Rendah        | 128 | 64  | 0.0001 | 64  | 0.2 | 0.0909 | 0.0164 |
| Tanpa Dropout    | 128 | 64  | 0.0005 | 64  | 0.0 | 0.0747 | 0.0108 |
| Dropout Tinggi   | 128 | 64  | 0.0005 | 64  | 0.3 | 0.0760 | 0.0111 |
| Batch Size Kecil | 128 | 64  | 0.0005 | 32  | 0.2 | 0.0758 | 0.0109 |
| Batch Size Besar | 128 | 64  | 0.0005 | 128 | 0.2 | 0.0739 | 0.0111 |

Analisis hasil menunjukkan bahwa konfigurasi Unit Besar (256/128 unit) mencapai performa terbaik dengan MAE 0.0735 dan MSE 0.0108, mengungguli Base maupun varian lainnya.



**Gambar 3. MSE Konfigurasi Unit Besar**



**Gambar 4. MAE Konfigurasi Unit Besar**

Kinerja superior ini tercermin dalam grafik konvergensi pelatihan pada Gambar 3 dan Gambar 4. Grafik konfigurasi Unit Besar menunjukkan hasil yang lebih optimal. Pada grafik konfigurasi Base, baik *validation loss* maupun *validation MAE* mulai mendatar dan menunjukkan sedikit kenaikan setelah sekitar 25 epoch, mengindikasikan potensi *overfitting*. Sebaliknya, grafik konfigurasi Unit Besar menampilkan kurva *validation loss* dan *validation MAE* yang lebih stabil dengan tren penurunan berkelanjutan hingga epoch terakhir, menunjukkan kemampuan generalisasi model yang lebih baik dan ketahanan terhadap *overfitting* yang lebih baik dalam durasi pelatihan yang sama.

### 3. Prediksi Iteratif

Untuk memvalidasi kinerja model, dilakukan simulasi prediksi iteratif selama 31 hari (29 Januari–28 Februari 2025) menggunakan konfigurasi Unit Besar terpilih (256/128 unit). Model dijalankan secara autoregresif dengan input data 28 hari sebelumnya (1–28 Januari 2025). Tabel 10 menyajikan ringkasan akurasi untuk lima bahan baku pertama.

**Tabel 10. Hasil Simulasi Prediksi 31 Hari**

| Bahan Baku   | Prediksi | Aktual   | Akurasi |
|--------------|----------|----------|---------|
| Ayam Dadu    | 6621.07  | 7247.50  | 91.36%  |
| Ayam Kampung | 83149.71 | 86400.00 | 96.24%  |

|                 |          |          |        |
|-----------------|----------|----------|--------|
| Ayam Pedaging   | 74050.18 | 79530.00 | 93.11% |
| Bayam           | 6943.69  | 6800.00  | 97.89% |
| Belimbing Wuluh | 2872.29  | 3218.00  | 89.26% |

Akurasi dihitung menggunakan dua pendekatan teknis, *Overall Aggregated Accuracy* (Makro) yang mengukur akurasi total agregat seluruh bahan baku dan *Average Per-Item Accuracy* (Mikro) yang menghitung rata-rata akurasi individual tiap bahan baku.

a. Rumus Akurasi Makro

$$\text{Makro} = \left(1 - \frac{|\sum \text{Aktual} - \sum \text{Prediksi}|}{\sum \text{Aktual}}\right) \times 100\%$$

b. Rumus Akurasi Mikro

$$\text{Mikro} = \frac{1}{N} \sum_{i=1}^N \left(1 - \frac{|\text{Aktual}_i - \text{Prediksi}_i|}{\text{Aktual}_i}\right) \times 100\%$$

Simulasi prediksi iteratif 31 hari untuk 36 bahan baku mencapai akurasi makro 90,27% untuk total kebutuhan bahan baku dan akurasi mikro 89,69% per bahan baku. Hasil ini menunjukkan kemampuan model dalam mengestimasi kebutuhan bahan baku secara komprehensif.

## SIMPULAN

Penelitian ini berhasil mengembangkan model LSTM untuk memprediksi kebutuhan 36 bahan baku di Bale Raos Kraton Yogyakarta. Tujuan utama tercapai melalui analisis pola historis dan musiman yang membuktikan fluktuasi permintaan dipengaruhi faktor temporal (hari kerja, akhir pekan, libur nasional, dan event khusus). Arsitektur LSTM berlapis (256/128 unit) yang dioptimalkan dengan *preprocessing* komprehensif, interpolasi temporal, fitur *lagging* (1-hari/7-hari), *one-hot encoding*, dan normalisasi MinMaxScaler, mencapai kinerja optimal dengan MSE 0.0108 dan MAE 0.0735. Simulasi prediksi iteratif 31 hari (Januari-Februari 2025) *Overall*

*Aggregated Accuracy* (Makro) 90.27%, meminimalkan risiko *overstock* dan *stockout*.

Saran untuk penelitian lanjutan, diusulkan eksplorasi model *hybrid* (misalnya LSTM-Transformer) untuk menangani pola non-periodik, integrasi variabel eksternal seperti promosi, cuaca, kebijakan pemerintah, serta validasi model pada berbagai konteks restoran guna menguji generalisasi dan adaptabilitasnya.

## DAFTAR PUSTAKA

- Nassibi, N., Fasihuddin, H., & Hsairi, L. (2023). Demand Forecasting Models for Food Industry by Utilizing Machine Learning Approaches. *International Journal of Food Science & Technology*, 14(3), 892–898. doi: 10.14569/IJACSA.2023.01403101
- Panda, S. K., & Mohanty, S. N. (2023). Time Series Forecasting and Modeling of Food Demand Supply Chain Based on Regressors Analysis. *Journal of Food Engineering*, 11, 42679–42700. doi: 10.1109/ACCESS.2023.3266275
- Schmidt, A., Kabir, M. W. U., & Hoque, M. T. (2022). Machine Learning Based Restaurant Sales Forecasting. *Machine Learning and Knowledge Extraction*, 4, 105–130. doi: 10.3390/make4010006
- Haque, M. S. (2023). Retail Demand Forecasting Using Neural Networks and Macroeconomic Variables. *Journal of Mathematics and Statistics Studies*, 4(3), 1–6. doi: 10.32996/jmss.2023.4.3.1
- Rodrigues, M., Migueis, V., Freitas, S., & Machado, T. (2023). Machine Learning Models for Short-Term Demand Forecasting in Food Catering Services: A Solution to Reduce Food Waste. *Journal of Cleaner Production*, 435, 1–16. doi: 10.1016/j.jclepro.2023.140265
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning Long-Term Dependencies with Gradient Descent

- is Difficult. *IEEE Transactions on Neural Networks*, 5(2), 157–166. doi: 10.1109/72.279181
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780. doi: 10.1162/neco.1997.9.8.1735
- Shihua, C., Xin, M., & Wen, L. (2024). Analysis and Optimization Strategies for Demand Forecasting Issues at Mixue. *International Journal on Economics, Finance and Sustainable Development (IJEFS)*, 7, 742–755. doi: 10.31149/ijefsd.v6i12.5346
- Niako, N., Melgarejo, J. D., Maestre, G. E., & Vatcheva, K.P. (2024). Effects of Missing Data Imputation Methods on Univariate Blood Pressure Time Series Data Analysis and Forecasting with ARIMA and LSTM. *BMC Medical Research Methodology*, 24, 1–32. doi: 10.1186/s12874-024-02448-3
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). New York, NY, USA: *Springer*.
- Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine Learning and Deep Learning. *Electronic Markets*, 31, 685–695. doi: 10.1007/s12525-021-00475-2
- Surakhi, O., Zaidan, M. A., Fung, P. L., Motlagh, N. H., Serhan, S., AlKhanafseh, M., Ghoniem, R. M., & Hussein, T. (2021). Time-Lag Selection for Time-Series Forecasting Using Neural Network and Heuristic Algorithm. *Electronics*, 10(20). doi: 10.3390/electronics10202518
- Albeladi, K., Zafar, B., & Mueen, A. (2023). Time Series Forecasting using LSTM and ARIMA. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 14(1), 313–320. doi: 10.14569/ijacsa.2023.0140133
- Liu, F., Chen, L., Zheng, Y., & Feng, Y. (2022). A Prediction Method with Data Leakage Suppression for Time Series. *Electronics*, 11(22). doi: 10.3390/electronics11223701
- Pranolo, A., Setyaputri, F.U., Paramarta, A. K. I., Triono, A. P. P., Fadhillah, A. F., Akbari, A. K. G., Utama, A. B. P., Wibawa, A. P., & Uriu, W. (2024). Enhanced Multivariate Time Series Analysis Using LSTM: A Comparative Study of Min-Max and Z-Score Normalization Techniques. *ILKOM Jurnal Ilmiah*, 16(2), 210–220. doi: 10.33096/ilkom.v16i2.2333.210-220
- Abdullah, E., Wee, M. I. H. M. A., Mohamad, R., & Hidayat, N. M. (2025). Enhancing Solar Irradiance Forecasting Using LSTM and Meteorological Data. *Journal of Information Systems Engineering and Management*, 10, 161–166. doi: 10.52783/jisem.v10i37s.6388
- Guo, F., Mo, H., Wu, J., Pan, L., Zhou, H., Zhang, Z., Li, L., & Huang, F. (2024). A Hybrid Stacking Model for Enhanced Short-Term Load Forecasting. *Electronics*, 13. doi: 10.3390/electronics13142719
- Siami-Namini, S., Tavakoli, N., & Namin, A.S. (2019). The Performance of LSTM and BiLSTM in Forecasting Time Series. *2019 IEEE International Conference on Big Data (Big Data)*, 1394–1401. doi: 10.1109/ICMLA.2018.00228
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15, 1929–1958. doi: 10.5555/2627435.2670313
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. Cambridge, MA, USA: *MIT Press*.