

Peramalan Kebutuhan Saldo K_One Top Up Menggunakan Algoritma *Long-Short Term Memory*(LSTM) Berdasarkan Data *Time Series*

^{1*}**Rachmad Abadi, ²Ardi Sanjaya, ³Ahmad Bagus Setiawan**

^{1 2 3}Teknik Informatika, Universitas Nusantara PGRI Kediri

E-mail: [*1rachmadabadi7@gmail.com](mailto:1rachmadabadi7@gmail.com), 2dersky@gmail.com, 3ahmadbagus@unpkediri.ac.id

Penulis Korespondens : Rachmad Abadi

Abstrak— Pengelolaan saldo yang akurat menjadi kunci dalam menjaga kelancaran layanan bisnis digital seperti top up pulsa, *voucher game*, dan *e-wallet*. K_One Top Up masih menerapkan sistem manual dalam mengelola saldo, sehingga rentan terhadap kesalahan dan inefisiensi. Penelitian ini bertujuan untuk meramalkan kebutuhan saldo harian menggunakan algoritma *Long Short-Term Memory* (LSTM) berbasis data *time series*. Model dibangun dengan dua lapisan LSTM bertingkat dan satu lapisan *output*, serta diuji melalui beberapa konfigurasi *neuron* dan *epoch*. Hasil pengujian menunjukkan bahwa LSTM mampu mengenali pola tren saldo dengan cukup baik, khususnya pada produk dengan fluktuasi moderat seperti pulsa. Konfigurasi terbaik memberikan nilai evaluasi akurasi yang paling rendah, menunjukkan potensi LSTM dalam membantu pengambilan keputusan pengelolaan saldo secara prediktif. Hasil penelitian ini penting untuk meningkatkan efisiensi operasional dan kualitas layanan K_One Top Up melalui sistem peramalan yang andal.

Kata Kunci— LSTM, Deep Learning, Time Series, Peramalan

Abstract— Accurate balance management is key to maintaining the smooth running of digital business services such as credit top up, game vouchers, and e-wallets. K_One Top Up still implements manual system in managing balances, making it prone to errors and inefficiencies. This study aims to forecast daily balance needs using the Long Short-Term Memory (LSTM) algorithm based on time series data. The model is built with two layers of nested LSTM and one output layer, and tested through several neuron and epoch configurations. The test results show that LSTM is able to recognize balance trend patterns quite well, especially for products with moderate fluctuations such as credit. The best configuration provides the lowest accuracy evaluation value, indicating the potential of LSTM in helping to make predictive balance management decisions. The results of this study are important for improving operational efficiency and service quality of K_One Top Up through a reliable forecasting system.

Keywords— LSTM, Deep Learning, Time Series, Forecasting

This is an open access article under the CC BY-SA License.



I. PENDAHULUAN

Pengelolaan saldo merupakan aspek krusial dalam operasional bisnis digital seperti usaha *top up* pulsa, *voucher game*, dan *e-wallet*. Ketersediaan saldo yang tepat dapat menjamin kelancaran layanan, sementara ketidaktepatan dalam perencanaan dapat menyebabkan kekurangan atau kelebihan saldo yang berdampak pada kepuasan pelanggan dan stabilitas operasional bisnis. Salah satu unit usaha yang mengalami tantangan tersebut adalah K_One Top

Up, yang berlokasi di Desa Bangsongan, Kabupaten Kediri. Saat ini, pengelolaan saldo di K_One Top Up masih dilakukan secara manual, yang dinilai kurang efisien dan rawan kesalahan. Untuk mengatasi permasalahan ini, dibutuhkan pendekatan prediktif yang mampu memperkirakan kebutuhan saldo secara akurat.

Forecasting atau peramalan merupakan teknik yang digunakan untuk memprediksi nilai atau kejadian di masa depan berdasarkan analisis data historis. *Forecasting* memiliki beberapa karakteristik penting, yaitu berbasis data historis, menggunakan model analitis, berorientasi pada masa depan, dan mengandung unsur ketidakpastian[1]. Dalam konteks pengelolaan saldo, *forecasting* berperan penting untuk memberikan gambaran mengenai permintaan saldo di masa mendatang, sehingga pengambilan keputusan dapat dilakukan secara lebih tepat.

Salah satu bentuk data yang umum digunakan dalam *forecasting* adalah data *time series*. Data *time series* merupakan data yang dikumpulkan berdasarkan urutan waktu dalam interval tertentu, dan digunakan untuk menganalisis pola atau tren yang berulang[2]. Karakteristik utama dari data *time series* adalah adanya dependensi temporal, yaitu hubungan antara data masa lalu dan masa depan. Komponen penting dalam analisis *time series* meliputi tren jangka panjang, variasi siklis, pola musiman, dan fluktuasi tak beraturan[3]. Keempat komponen ini memberikan gambaran menyeluruh terhadap dinamika yang terjadi dalam data waktu, sehingga menjadi dasar penting dalam membangun model prediktif.

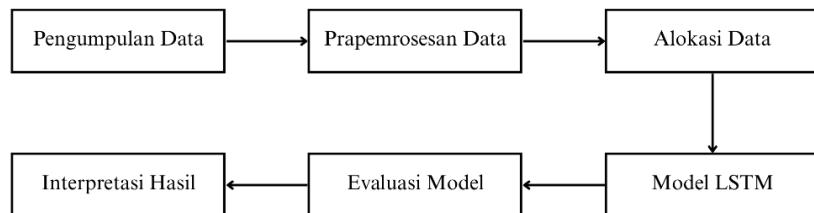
Salah satu algoritma yang banyak digunakan dalam analisis data *time series* adalah *Long Short-Term Memory* (LSTM). LSTM merupakan pengembangan dari *Recurrent Neural Network* (RNN) yang dirancang untuk mengatasi kelemahan RNN dalam mengingat informasi jangka panjang [4]. LSTM mampu mengelola informasi melalui tiga gerbang utama: *forget gate* (untuk menghapus informasi yang tidak relevan), *input gate* (untuk menyimpan informasi baru), dan *output gate* (untuk menghasilkan keluaran berdasarkan informasi terbaru dan memori). Kemampuan LSTM dalam menyimpan memori jangka panjang dan pendek menjadikannya unggul dalam memprediksi data yang memiliki ketergantungan temporal kuat, seperti data saldo harian.

Sejumlah penelitian terdahulu telah menunjukkan efektivitas LSTM dalam memprediksi data *time series*. Penelitian oleh Yanti dkk.[5] membuktikan bahwa LSTM berhasil memprediksi kebutuhan stok obat dengan tingkat akurasi tinggi, ditunjukkan oleh nilai MAPE terendah 4,82%. Sementara Kurniawati dkk.[6] menyimpulkan bahwa LSTM lebih unggul dibandingkan metode ARIMA dalam hal akurasi prediksi stok obat.

Berdasarkan keberhasilan LSTM dalam berbagai studi, metode ini dinilai potensial untuk diterapkan dalam peramalan kebutuhan saldo di K_One Top Up. Dengan penerapan metode LSTM, diharapkan pengelolaan saldo dapat menjadi lebih efektif dan responsif terhadap fluktuasi permintaan, sehingga meningkatkan kualitas layanan yang diberikan kepada konsumen.

II. METODE

Ada beberapa tahapan penelitian yang terbagi menjadi enam tahapan, antara lain pengumpulan data, prapemrosesan data, alokasi data, model LSTM, evaluasi model dan interpretasi hasil.



Gambar 1. Tahapan Penelitian

A. Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah data transaksi dari K_ONE Top Up mulai tanggal 1 Januari 2024 sampai 31 Desember 2024 dengan tiga kategori produk yaitu pulsa, *game*, dan *e-wallet*. Adapun atribut yang digunakan adalah tanggal dan saldo harian terpakai.

Tabel 1. Sampel data

Tanggal	Pulsa	Game	E-Wallet
2024-01-01	20045	50644	0
2024-01-02	61886	29865	25270
2024-01-03	32248	18500	0
2024-01-04	45470	17042	0
2024-01-05	61886	34325	10000

B. Prapemrosesan Data

Pemrosesan data dilakukan untuk memastikan data yang digunakan dalam pelatihan model berada dalam kondisi optimal. Tahapan ini mencakup dua proses utama, yaitu penanganan outlier dan normalisasi data.

Penanganan *outlier* dilakukan menggunakan metode *Interquartile Range* (IQR) untuk mendeteksi nilai-nilai ekstrem yang menyimpang dari pola umum data. *Outlier* yang ditemukan kemudian ditangani agar tidak memengaruhi hasil pelatihan model.

$$IQR = Q_3 - Q_1 \quad (1)$$

Kuartil pertama Q_1 merupakan nilai yang membagi 25% data terbawah, sedangkan kuartil ketiga Q_3 adalah nilai yang membagi 75% data terbawah atau 25% data teratas.

Normalisasi data dilakukan dengan metode *Min-Max Scaling*, yang mengubah skala data ke rentang 0 hingga 1[7]. Langkah ini penting karena model LSTM sensitif terhadap perbedaan skala, sehingga normalisasi membantu meningkatkan akurasi dan efisiensi proses pelatihan model.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2)$$

Pada rumus diatas, X_{norm} adalah hasil dari normalisasi, X merupakan nilai data asli yang akan dinormalisasi, sedangkan X_{max} nilai tertinggi keseluruhan data dan X_{min} adalah nilai terendah keseluruhan data.

C. Alokasi Data

Data dibagi menjadi dua bagian utama, yaitu data training dan data testing. Data *training* digunakan untuk melatih model agar dapat mempelajari pola historis dari data *time*

series, sedangkan data *testing* digunakan untuk mengevaluasi kinerja model terhadap data yang belum pernah dilihat sebelumnya.

Pembagian data dilakukan dengan proporsi 80% untuk data *training* dan 20% untuk data *testing*. Pemisahan ini dilakukan secara berurutan berdasarkan waktu untuk menjaga urutan kronologis data, yang merupakan hal penting dalam analisis deret waktu.

D. Model LSTM

Model yang digunakan dalam penelitian ini adalah *Long Short-Term Memory* (LSTM), yang merupakan salah satu jenis *Recurrent Neural Network* (RNN) yang dirancang untuk mengatasi permasalahan dalam mempelajari dependensi jangka panjang pada data *time series*[4].

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (3)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

$$\hat{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (5)$$

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \quad (6)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (7)$$

$$h_t = o_t * \tanh(C_t) \quad (8)$$

Rumus diatas merupakan representasi matematis dari mekanisme internal sel *Long Short Term Memory*(LSTM). Pada setiap waktu (t), *input gate* (i_t) mengatur seberapa banyak data baru dari *input* (x_t) dan *hidden state* sebelumnya (h_{t-1}) yang akan masuk ke memori, melalui fungsi aktivasi *sigmoid*. *Forget gate* (f_t) menentukan seberapa banyak data lama dari memori sebelumnya (C_{t-1}) yang perlu dilupakan. *Cell state* baru (\hat{C}_t) dihitung menggunakan fungsi *tanh* dan dikombinasikan dengan (i_t) untuk memperbarui *cell state* (C_t) bersama dengan $f_t * C_{t-1}$. *Output gate* (o_t) menentukan bagian mana dari memori yang akan dihasilkan, dan *hidden state* (h_t) dihitung sebagai hasil dari (o_t) dikalikan dengan *tanh* dari (C_t). Semua parameter W dan b adalah bobot dan bias yang dipelajari selama pelatihan model LSTM. Kombinasi komponen ini memungkinkan LSTM untuk mempertahankan dan memproses informasi secara efektif dalam jangka waktu panjang[8].

E. Evaluasi Model

Evaluasi model dilakukan untuk mengukur tingkat akurasi dan kinerja model prediksi yang telah dibangun. Dalam penelitian ini, digunakan dua metrik evaluasi utama, yaitu *Mean Absolute Error* (MAE) dan *Root Mean Squared Error* (RMSE). Kedua metrik ini umum digunakan dalam analisis data *time series* karena mampu memberikan gambaran tentang seberapa dekat hasil prediksi model terhadap nilai aktual.

MAE mengukur rata-rata selisih absolut antara nilai prediksi dengan nilai aktual[9]. Metrik ini memberikan informasi tentang seberapa besar kesalahan prediksi dalam satuan yang sama dengan data asli. MAE dirumuskan sebagai berikut:

$$MAE = \sqrt{\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|} \quad (9)$$

RMSE memberikan penalti yang lebih besar terhadap kesalahan prediksi yang besar karena menggunakan kuadrat dari selisih antara nilai prediksi dan aktual sebelum dihitung

rata-ratanya. RMSE lebih sensitif terhadap outlier dibandingkan MAE[10]. Rumus RMSE adalah sebagai berikut:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (10)$$

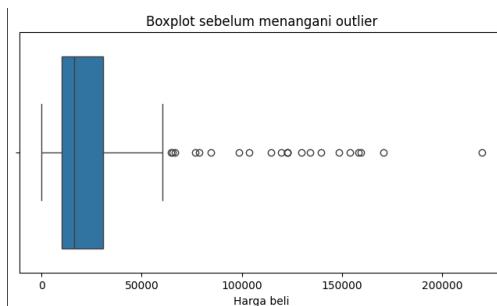
Semakin kecil nilai MAE dan RMSE, maka semakin baik kinerja model dalam memprediksi data saldo di masa mendatang. Penggunaan kedua metrik ini secara bersamaan memberikan evaluasi yang lebih komprehensif terhadap performa model LSTM.

F. Interpretasi Hasil

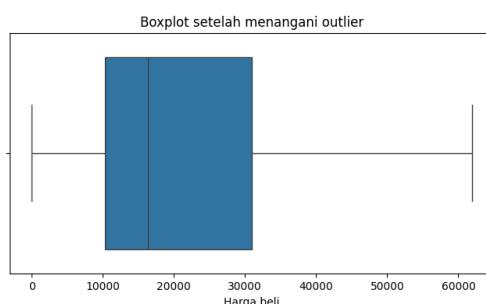
Pada tahap interpretasi hasil, evaluasi performa model dilakukan dengan membandingkan nilai prediksi yang dihasilkan oleh model LSTM terhadap data aktual. Penggunaan metrik evaluasi seperti *Mean Absolute Error* (MAE) dan *Root Mean Squared Error* (RMSE) membantu dalam menilai akurasi dan kesalahan prediksi secara keseluruhan.

III. HASIL DAN PEMBAHASAN

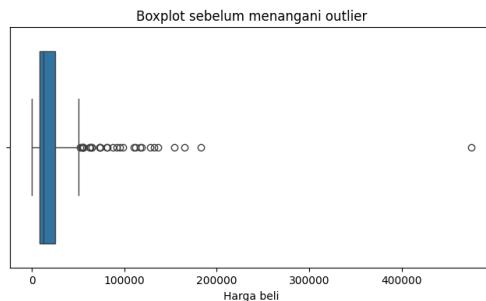
Sebelum dilakukan proses modeling, data menjalani prapemrosesan yang dilakukan dengan pengecekan *outlier* pada setiap produk. Berikut hasil dari pengecekan dan penanganan *outlier*:



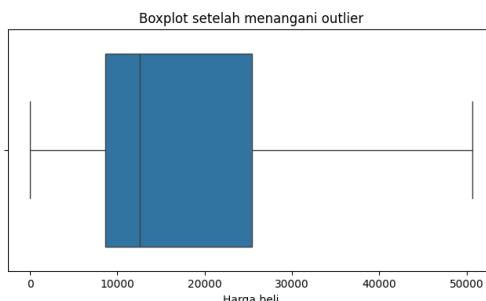
Gambar 2. Pengecekan *outlier* produk pulsa



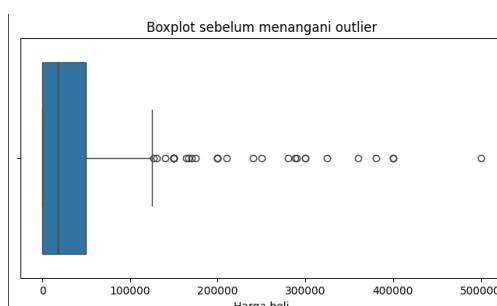
Gambar 3. Hasil penanganan *outlier* produk pulsa



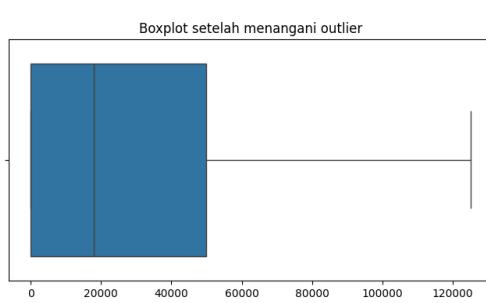
Gambar 4. Pengecekan *outlier* produk *game*



Gambar 5. Hasil penanganan *outlier* produk *game*



Gambar 6. Pengecekan *outlier* produk *ewallet*



Gambar 7. Hasil penanganan *outlier* produk *ewallet*

Dari hasil pengecekan dan penanganan *outlier* terdapat data *outlier* pada produk pulsa, *game* dan *ewallet*, maka dari itu dilakukan penanganan *outlier* dengan masing masing pulsa dengan batas bawah 0.00, batas atas 61886.75, *game* dengan batas bawah 0.00, batas atas 50644.62, serta *ewallet* dengan bawah 0.00, batas atas 125000.00.

Langkah selanjutnya adalah normalisasi data menggunakan *min-max scaling* untuk mengubah rentang nilai asli data menjadi skala antara 0 sampai 1. Berikut perbandingan data sebelum dan sesudah normalisasi:

Tabel 2. Sebelum normalisasi data

Tanggal	Pulsa	Game	E-Wallet
2024-01-01	20045	50644	0
2024-01-02	61886	29865	25270
2024-01-03	32248	18500	0
2024-01-04	45470	17042	0
2024-01-05	61886	34325	10000

Tabel 3. Sesudah normalisasi data

Tanggal	Pulsa	Game	E-Wallet
2024-01-01	0.323898	1.000000	0.00000
2024-01-02	1.000000	0.589697	0.20216
2024-01-03	0.521081	0.365290	0.00000
2024-01-04	0.734729	0.336502	0.00000
2024-01-05	1.000000	0.677762	0.08000

Berdasarkan tabel 2 dan 3, proses normalisasi data menggunakan *min-max scaling* menunjukkan hasil yang baik dengan mempertahankan pola historis data saldo. Normalisasi membantu mempercepat konvergensi selama pelatihan dan menghindari dominasi fitur dengan nilai yang lebih besar terhadap model. Oleh karena itu, langkah ini menjadi salah satu tahapan krusial dalam *preprocessing* data sebelum dilakukan pelatihan model.

Pada tahap pemodelan, LSTM dibangun dengan dua lapisan bertingkat, dimana lapisan pertama dilakukan dengan percobaan 16, 32, 64 unit *neuron* dengan *drop out* 0.2. Lapisan LSTM kedua dilakukan dengan percobaan 8, 16, 32 unit *neuron*. Lapisan akhir berupa lapisan *Dense* dengan satu *neuron* sebagai *output layer* yang menghasilkan nilai prediksi saldo. Model menggunakan *optimizer Adam* dengan *learning rate* sebesar 0.001 untuk mempercepat proses konvergensi. Model dilatih dengan percobaan 300, 400, 500 *epoch* dan 16 *batch size* agar model dapat sepenuhnya belajar dari keseluruhan siklus *epoch* yang ditentukan. Berikut adalah tabel hasil uji coba dari beberapa konfigurasi LSTM yang menggunakan matriks evaluasi MAE dan RMSE:

Tabel 4. Hasil produk pulsa

Neuron Lapisan 1	Neuron Lapisan 2	Jumlah Epoch	RMSE	MAE
16	8	300	18002	13673
		400	17974	13672
		500	17745	13604
32	16	300	17750	13321
		400	18520	14183
		500	18926	14802
64	32	300	19101	14782
		400	20001	15631
		500	21310	16336

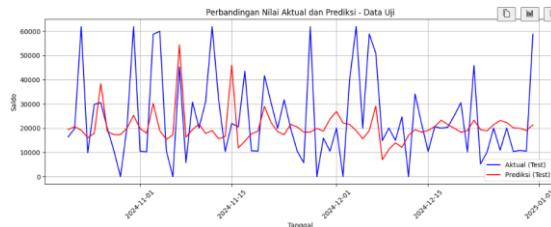
Tabel 5. Hasil produk *game*

Neuron Lapisan 1	Neuron Lapisan 2	Jumlah Epoch	RMSE	MAE
16	8	300	12086	9685
		400	12024	9650
		500	12086	9725
32	16	300	12389	9777
		400	13302	10618
		500	14542	11275
64	32	300	13971	11026
		400	15534	12416
		500	17417	14116

Tabel 6. Hasil produk *ewallet*

Neuron Lapisan 1	Neuron Lapisan 2	Jumlah Epoch	RMSE	MAE
16	8	300	34598	29273
		400	34502	28452
		500	35337	28833
32	16	300	34758	28901
		400	35976	28508
		500	38333	27916
64	32	300	39419	29180
		400	45059	31878
		500	47242	35219

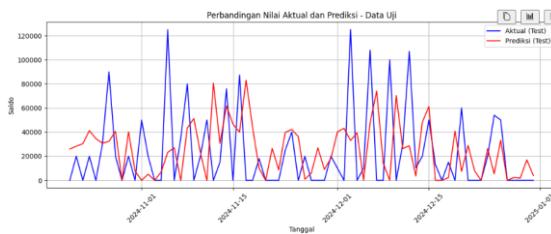
Tabel 4, 5 dan 6 merupakan hasil evaluasi dari setiap produk. Berdasarkan hasil pengujian model LSTM terhadap tiga jenis produk (pulsa, *game*, dan *e-wallet*), diperoleh konfigurasi terbaik masing-masing berdasarkan nilai RMSE dan MAE terendah. Untuk produk pulsa, konfigurasi optimal adalah pada lapisan pertama dengan 32 *neuron*, lapisan kedua 16 *neuron*, dan 300 *epoch*, menghasilkan RMSE 17.750 dan MAE 13.321. Produk *game* menunjukkan hasil terbaik pada konfigurasi 16 *neuron* di lapisan pertama, 8 *neuron* di lapisan kedua, dan 400 *epoch*, dengan RMSE 12.024 dan MAE 9.650. Sementara itu, produk *e-wallet* memperoleh hasil terbaik dengan konfigurasi 32 *neuron* di lapisan pertama, 16 *neuron* di lapisan kedua, dan 500 *epoch*, menghasilkan RMSE 38.333 dan MAE 27.916. Meskipun nilai RMSE dan MAE masih tergolong besar, konfigurasi ini menghasilkan kesalahan paling rendah dibanding konfigurasi lainnya.



Gambar 8. Perbandingan nilai aktual dan prediksi produk pulsa



Gambar 9. Perbandingan nilai aktual dan prediksi produk game



Gambar 10. Perbandingan nilai aktual dan prediksi produk ewallet

Berdasarkan Gambar 8, 9 dan 10, diperoleh gambaran visual perbandingan antara nilai aktual dan hasil prediksi dari model LSTM untuk masing-masing produk, yaitu pulsa, game, dan ewallet. Secara umum, kurva prediksi (garis merah) mampu mengikuti pola umum tren saldo yang tercermin dari kurva aktual (garis biru), meskipun terdapat selisih nilai terutama pada titik-titik ekstrim atau lonjakan nilai saldo. Pada produk pulsa (Gambar 8), model mampu menangkap ritme naik-turun data dengan lebih baik, meskipun masih terjadi deviasi pada beberapa titik puncak. Sementara itu, pada produk game (Gambar 9), prediksi tampak lebih halus dan cenderung stabil, namun kurang responsif terhadap perubahan mendadak yang terjadi pada data aktual. Sedangkan untuk produk e-wallet (Gambar 10), model menunjukkan kesulitan dalam menangkap fluktuasi tinggi, yang ditandai dengan banyaknya deviasi antara nilai aktual dan prediksi. Hal ini menunjukkan bahwa model LSTM yang digunakan lebih akurat dalam memprediksi pola data yang memiliki fluktuasi moderat (seperti pada produk pulsa), dibandingkan data dengan pola yang sangat dinamis atau ekstrem seperti e-wallet.

IV. KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa model *Long Short-Term Memory* (LSTM) dapat digunakan untuk memprediksi kebutuhan saldo K_One Top Up dari tiga jenis produk digital, yaitu pulsa, game, dan e-wallet. Berdasarkan hasil evaluasi menggunakan metrik RMSE dan MAE, diperoleh bahwa konfigurasi optimal untuk produk pulsa adalah 32-16 *neuron* dengan 300 *epoch* menghasilkan RMSE 17.750 dan MAE 13.321, untuk produk game adalah

16-8 *neuron* dengan 400 *epoch* menghasilkan RMSE 12.024 dan MAE 9.650, dan untuk produk *e-wallet* adalah 32-16 *neuron* dengan 500 *epoch* menghasilkan RMSE 38.333 dan MAE 27.916. Hasil visualisasi grafik menunjukkan bahwa model cukup baik dalam menangkap pola tren saldo, khususnya untuk produk dengan fluktuasi moderat seperti pulsa. Namun, untuk produk dengan volatilitas tinggi seperti *e-wallet*, model mengalami kesulitan dalam mengikuti lonjakan data secara akurat. Dengan demikian, LSTM dapat memprediksi data *time series* saldo produk digital, meskipun performanya dipengaruhi oleh tingkat kestabilan pola data yang dianalisis. Harapannya temuan ini berpotensi digunakan sebagai alat bantu dalam pengambilan keputusan pengelolaan saldo di K_One Top Up.

DAFTAR PUSTAKA

- [1] M. F. Rizkillah and S. Widyanesti, “Prediksi Harga Cryptocurrency Menggunakan Algoritma Long Short Term Memory (LSTM),” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 1, pp. 25–31, Feb. 2022, doi: 10.29207/resti.v6i1.3630.
- [2] M. L. Tauryawati and M. I. Irawan, “Perbandingan Metode Fuzzy Time Series Cheng dan Metode Box-Jenkins untuk Memprediksi IHSG,” *J. SAINS DAN SENI POMITS*, vol. 3, no. 2, 2014, [Online]. Available: www.idx.co.id
- [3] S. Nawangwulan and D. Angesti, “ANALISIS TIME SERIES METODE WINTER JUMLAH PENDERITA GASTROENTERITIS RAWAT INAP BERDASARKAN DATA REKAM MEDIS DI RSUD DR. SOETOMO SURABAYA,” *J. Manaj. Kesehat. STIKES Yayasan RS. Dr. Soetomo*, vol. 2, no. 1, pp. 17–32, 2016.
- [4] S. Zahara, S. Sugianto, and M. B. Ilmuddafiq, “Prediksi Indeks Harga Konsumen Menggunakan Metode Long Short Term Memory (LSTM) Berbasis Cloud Computing,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 3, pp. 357–363, 2019.
- [5] F. Yanti, B. Nurina Sari, and S. Defiyanti, “IMPLEMENTASI ALGORITMA LSTM PADA PERAMALAN STOK OBAT (STUDI KASUS: PUSKESMAS BEBER),” *J. Mhs. Tek. Inform.*, vol. 8, no. 4, 2024.
- [6] A. Kurniawati, M. Sabri Ahmad, M. Fhadli, and S. Lutfi, “ANALISIS PERBANDINGAN METODE TIME SERIES FORECASTING UNTUK PREDIKSI PENJUALAN OBAT DI APOTEK (STUDI KASUS: KIMIA FARMA APOTEK TAKOMA),” vol. 3, no. 1, pp. 96–106, 2023.
- [7] F. C. Yulianto and N. Latifah, “Peramalan Penjualan Laptop Menggunakan Metode Long Short Term Memory (LSTM),” *J. FASILKOM*, vol. 14, no. 2, pp. 428–436, 2024, [Online]. Available: https://github.com/C4AnN/Laptop_Lens/blob/main/M
- [8] R. M. S. Adi and S. Sudianto, “Prediksi Harga Komoditas Pangan Menggunakan Algoritma Long Short-Term Memory (LSTM),” *Build. Informatics, Technol. Sci.*, vol. 4, no. 2, Sep. 2022, doi: 10.47065/bits.v4i2.2229.
- [9] A. Santoso, A. I. Purnamasari, and I. Ali, “PREDIKSI HARGA BERAS MENGGUNAKAN METODE RECURRENT NEURAL NETWORK DAN LONG SHORT-TERM MEMORY,” *J. PROSISKO*, vol. 11, no. 1, 2024.
- [10] H. D. E. Sinaga and N. Irawati, “Perbandingan Double Moving Average dengan Double Exponential Smoothing pada Peramalan Bahan Medis Habis Pakai,” *J. Teknol. dan Sist. Inf.*, vol. 4, no. 2, pp. 197–204, 2018.