

Evaluasi Performa Model ARIMA dan LSTM dengan Permintaan Inventarisasi Produk pada Perusahaan Produksi

Aris Martono^{*1}, Padeli², Muhamad iip Suhaepi³, Anur Rahmah Tia Wulandari⁴

^{*1,3}Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Raharja

²Program Studi Pendidikan Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Raharja

Email: aris.martono@raharja.info, padeli@raharja.info, iip@raharja.info, anur.rahmah@raharja.info

Abstrak

Penelitian ini bertujuan untuk mengevaluasi dan membandingkan performa dua pendekatan prediksi deret waktu, yaitu model statistik klasik ARIMA dan model deep learning LSTM, dalam konteks prediksi permintaan inventaris produk pada perusahaan produksi. Data yang digunakan merupakan data historis permintaan harian sejumlah 100 record dan 200 record, yang dianalisis untuk mengidentifikasi pola linier dan non-linier. Model ARIMA dipilih karena keandalannya dalam memodelkan data stasioner dan bermusim, sedangkan LSTM digunakan untuk menangkap pola temporal kompleks melalui arsitektur jaringan saraf berlapis. Hasil pengujian menggunakan metrik MSE dan RMSE menunjukkan bahwa pada kedua dataset, model ARIMA memiliki performa prediksi yang lebih baik (100 record, RMSE=45.61% dan 200 record, RMSE=44.72%) dibandingkan LSTM yaitu 100 record, RMSE=45.93% dan 200 record, RMSE=49.54%. Meskipun LSTM memiliki keunggulan dalam menangani dinamika non-linier, performa ARIMA lebih unggul pada data bersifat linier. Penelitian ini menegaskan pentingnya pemilihan model berdasarkan karakteristik data dan membuka peluang untuk eksplorasi model hibrida di masa depan. Referensi yang mendasari penelitian ini antara lain adalah karya Hyndman & Athanasopoulos (2018), Hochreiter & Schmidhuber (1997), dan Makridakis et al. (2018) yang memberikan dasar teoritis dan empiris dalam penggunaan model prediktif untuk analisis deret waktu.

Kata Kunci– Linier dan non-linier, ARIMA, LSTM.

Abstract

This study aims to evaluate and compare the performance of two time series forecasting approaches: the classical statistical ARIMA model and the deep learning-based LSTM model, in the context of forecasting product inventory demand in a production company. The data used consists of historical daily demand records, totaling 100 and 200 records, which were analyzed to identify linear and non linear patterns. The ARIMA model was selected for its reliability in modeling stationary and seasonal data, while the LSTM model was utilized to capture complex temporal patterns through its layered neural network architecture. The test results using the MSE and RMSE metrics show that in both datasets, the ARIMA model has better prediction performance (100 records, RMSE=45.61% and 200 records, RMSE=44.72%) compared to LSTM, namely 100 records, RMSE=45.93% and 200 records, RMSE=49.54%. Although LSTM excels in handling non-linear dynamics, ARIMA outperformed it on data with linear. This study highlights the importance of selecting forecasting models based on data characteristics and suggests opportunities for future exploration of hybrid models. The theoretical and empirical foundations of this research are supported by the works of

Hyndman & Athanasopoulos (2018), Hochreiter & Schmidhuber (1997), and Makridakis et al. (2018), which provide critical insight into predictive modeling for time series analysis.

Keywords– *Linear and non-linear, ARIMA, LSTM.*

1. PENDAHULUAN

Dalam dunia bisnis dan industri modern, prediksi berbasis data menjadi tulang punggung pengambilan keputusan yang efektif. Salah satu tantangan utama yang dihadapi perusahaan produksi saat ini adalah bagaimana memprediksi fenomena yang berlangsung secara berurutan dari waktu ke waktu—seperti permintaan produk, harga pasar, dan konsumsi energi—dengan akurasi tinggi dan efisien. Di sinilah analisis deret waktu (*time series analysis*) memainkan peran penting.

Permasalahan utama yang dihadapi dalam analisis deret waktu adalah kemampuan untuk mengenali pola dan tren dari data historis, serta menggunakannya untuk memprediksi nilai masa depan secara akurat [1]. Banyak perusahaan produksi mengalami kerugian besar karena kesalahan dalam estimasi permintaan atau penjadwalan yang buruk akibat pemodelan data deret waktu yang kurang tepat.

Dalam beberapa dekade terakhir, dua pendekatan utama telah digunakan dalam analisis deret waktu, yaitu model statistik klasik seperti ARIMA (AutoRegressive Integrated Moving Average) dan pendekatan berbasis pembelajaran mesin mendalam seperti LSTM (Long Short-Term Memory). Masing-masing metode memiliki kelebihan dan kekurangan dalam hal akurasi, interpretabilitas, dan kemampuan menangkap kompleksitas pola data.

ARIMA dikenal luas sebagai model statistik linier yang efektif dalam menangani data yang memiliki pola stabil atau tren musiman yang jelas [2]. Namun, model ini memiliki keterbatasan dalam menangani pola non-linear dan data yang sangat dinamis.

Sebaliknya, LSTM, yang merupakan bagian dari arsitektur jaringan saraf dalam, mampu mengenali pola-pola kompleks dan interaksi non-linier dalam data time series. LSTM memanfaatkan memori jangka panjang yang dapat menangkap ketergantungan temporal lebih dalam [3].

Tantangan besar dalam penerapan ARIMA maupun LSTM adalah bagaimana menyesuainya dengan berbagai sektor industri yang memiliki karakteristik data yang berbeda. Dalam konteks bisnis ritel dan e-commerce, ketepatan prediksi permintaan sangat penting untuk menjaga keseimbangan antara ketersediaan dan efisiensi biaya persediaan [4][5].

Isu lain yang tidak kalah penting adalah kemampuan model untuk beradaptasi dengan big data. Dengan semakin meningkatnya volume data dari berbagai sumber, model prediktif harus mampu mengolah informasi dalam skala besar secara efisien tanpa mengorbankan akurasi [6].

Kebutuhan akan pemodelan yang fleksibel juga muncul dari kondisi pasar yang semakin volatile, seperti di sektor keuangan. LSTM telah digunakan secara luas untuk memprediksi harga saham dan aset lainnya karena mampu menangkap dinamika pasar yang kompleks [7].

Dalam industri energi, penggunaan ARIMA telah terbukti efektif untuk memprediksi permintaan harian atau musiman, yang kemudian digunakan untuk perencanaan pasokan dan efisiensi produksi [8]. Meskipun demikian, pendekatan ini bisa kurang adaptif terhadap perubahan drastis atau kejadian tak terduga.

Penelitian yang menunjukkan bahwa model ARIMA memiliki keandalan tinggi dalam konteks data stasioner [9], namun harus dioptimalkan dengan teknik diferensiasi agar dapat digunakan pada data non-stasioner. Sementara itu, penelitian menekankan pentingnya analisis ACF—Auto Correlation Function—dan Parsial ACF untuk pemilihan parameter model ARIMA. [2][10]

Penelitian pada dasarnya mengembangkan LSTM untuk mengatasi permasalahan vanishing gradient pada jaringan saraf konvensional. [3] Penemuan ini membuka jalan bagi penggunaan model deep learning pada time series dengan hasil prediksi yang lebih canggih dan presisi.

Dalam studi kasus prediksi permintaan inventaris yang diulas dalam dokumen ini, terbukti bahwa kombinasi pemodelan statistik (ARIMA) dan pembelajaran mendalam (LSTM) dapat memberikan wawasan komprehensif terhadap dinamika permintaan pasar. Keduanya dibandingkan dari sisi performa prediksi menggunakan metrik seperti MSE dan RMSE.

Ulasan literatur menunjukkan bahwa pemodelan ARIMA tetap relevan dan bermanfaat, terutama ketika digunakan sebagai baseline atau pembanding bagi model berbasis machine learning. Sedangkan LSTM menunjukkan potensi besar dalam meningkatkan akurasi prediksi, khususnya pada data yang menunjukkan perilaku fluktuatif dan tidak linier.

Ke depan, integrasi ARIMA dan LSTM, baik dalam bentuk hibrida maupun ensemble learning, menjadi tren yang semakin kuat dalam dunia penelitian time series. Pendekatan ini mencoba menggabungkan kekuatan interpretabilitas ARIMA dan fleksibilitas non-linier dari LSTM[6].

Oleh karena itu, kajian mendalam terhadap model ARIMA dan LSTM sangat penting untuk menjawab tantangan prediksi data deret waktu yang semakin kompleks di era big data. Pemahaman terhadap fondasi teori, isu implementasi, serta hasil-hasil penelitian sebelumnya menjadi dasar kuat dalam pengembangan solusi prediktif yang adaptif dan akurat.

2. METODE PENELITIAN

Penelitian ini bertujuan untuk mengevaluasi kinerja dua pendekatan utama dalam prediksi data deret waktu, yaitu metode statistik klasik ARIMA dan model pembelajaran mendalam LSTM. Pendekatan ini dipilih karena masing-masing mewakili paradigma yang berbeda dalam analisis data deret waktu—yakni statistik konvensional dan kecerdasan buatan.

Tahapan pertama dalam metode penelitian ini adalah analisis karakteristik data. Data yang digunakan berupa dataset historis permintaan inventaris selama periode waktu tertentu, yang mencakup atribut seperti tanggal, permintaan harian, biaya produksi, dan harga jual. Dataset ini terdiri dari 100 dan 200 entri data harian yang mencerminkan dinamika permintaan inventaris aktual di sektor bisnis.

Untuk menganalisis data tersebut, dilakukan pemrosesan awal yang mencakup transformasi tipe data tanggal, pengaturan ulang indeks, penanganan missing values, serta visualisasi awal menggunakan matplotlib untuk memahami tren dan fluktuasi data. Visualisasi ini menjadi dasar penting untuk menentukan model yang sesuai—baik ARIMA maupun LSTM.

Dalam tahap selanjutnya, diterapkan model ARIMA, yang merupakan gabungan dari tiga komponen: *autoregressive (AR)*, *integrated (I)*, dan *moving average (MA)*. Model AR menangkap hubungan antara nilai saat ini dengan nilai masa lalu, komponen I melakukan diferensiasi untuk mencapai kestasioneran, dan MA memodelkan kesalahan prediksi sebagai kombinasi kesalahan sebelumnya [9].

Secara matematis, model ARIMA dinyatakan dalam bentuk umum:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} \dots + \phi_p Y_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \dots + \theta_q \epsilon_{t-q}$$

Di mana Y_t adalah nilai saat ini, ϕ adalah koefisien AR, θ adalah koefisien MA, dan ϵ_t adalah kesalahan acak.

Proses identifikasi parameter ARIMA (p,d,q) dilakukan secara otomatis menggunakan fungsi `auto_arima()` dari pustaka Python `pmdarima`, yang mengevaluasi berbagai kombinasi parameter dengan meminimalkan AIC (Akaike Information Criterion). Setelah parameter optimal diperoleh, model dilatih menggunakan data latih, kemudian diuji dengan data uji.

Pada order ARIMA (1,1,1), $p = 1$. Ini berarti model menggunakan nilai deret waktu pada satu langkah waktu sebelumnya untuk memprediksi nilai saat ini.

I (Integrated) yaitu komponen ini menangani data deret waktu yang tidak stasioner (non-stationary) dengan melakukan proses differencing (pengurangan). Data dikatakan tidak stasioner jika sifat statistiknya (seperti rata-rata dan varians) berubah seiring waktu. Differencing membantu membuat data menjadi stasioner. Parameter d menunjukkan berapa kali differencing dilakukan.

Pada order ARIMA (1,1,1), $d = 1$. Ini berarti model melakukan *satu kali differencing* pada data asli untuk membuatnya stasioner sebelum menerapkan komponen AR dan MA. Deret waktu yang sudah didifferencing satu kali adalah selisih antara nilai saat ini dengan nilai satu langkah waktu sebelumnya.

MA (Moving Average) ialah komponen ini menunjukkan hubungan antara nilai saat ini dengan error prediksi (residual) dari model *moving average* sebelumnya. Parameter q menunjukkan jumlah *lag* dari error prediksi yang digunakan dalam model moving average.

Pada order ARIMA (1,1,1), $q = 1$. Ini berarti model menggunakan *error* prediksi dari satu langkah waktu sebelumnya untuk memprediksi nilai saat ini.

Untuk model kedua, yaitu LSTM, digunakan pendekatan jaringan saraf dalam (deep learning) yang sangat efektif dalam mengenali pola temporal yang kompleks dan non-linier dalam data sekuensial [3]. LSTM mampu mempertahankan informasi dari waktu sebelumnya melalui mekanisme sel memori dan tiga jenis gerbang.

Arsitektur jaringan LSTM dalam penelitian ini terdiri atas beberapa *layer* LSTM berurutan, dilengkapi dengan *Dropout layer* untuk menghindari overfitting, serta *Dense layer* di output. Model ini dibangun menggunakan pustaka TensorFlow Keras dengan optimasi melalui algoritma *Adam*.

Fungsi-fungsi utama dalam arsitektur LSTM meliputi:

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

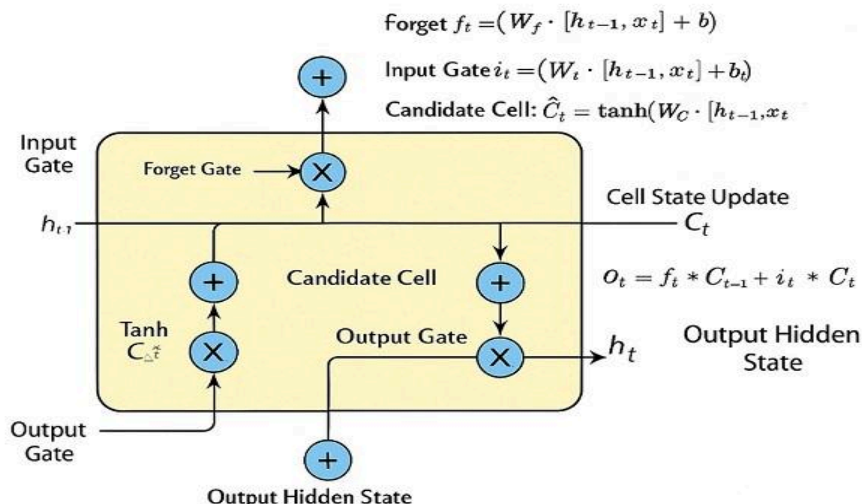
$$\text{Input Gate: } i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\text{Candidate Cell: } \tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

$$\text{Output Gate: } o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$\text{Cell State Update: } C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Diagram arsitektur jaringan LSTM dalam bentuk layer-layer berurutan dan dilengkapi dengan fungsi-fungsi utamanya dapat dilihat pada gambar 1.



Gambar 1. Arsitektur jaringan LSTM berbentuk layer-layer

Data masukan untuk model LSTM dinormalisasi menggunakan *MinMaxScaler*, agar nilai-nilai data berada pada skala yang konsisten, mempercepat konvergensi model selama pelatihan. Data kemudian diubah menjadi format sekuensial dengan parameter *time_step* yang menentukan panjang jendela input.

Proses pelatihan model dilakukan dalam 100 epoch dengan *batch_size* sebesar 32. Data dibagi ke dalam data latih dan uji. Fungsi aktivasi yang digunakan dalam LSTM adalah *tanh* dan *sigmoid*, sedangkan *loss function* yang digunakan adalah *mean squared error (MSE)*.

Evaluasi performa kedua model dilakukan dengan membandingkan nilai prediksi terhadap data aktual menggunakan metrik statistik seperti:

$$MSE \text{ (Mean Squared Error)} = \frac{1}{n} \sum_{i=1}^n \left(y_1 - \hat{y}_2 \right)^2$$

$$RMSE \text{ (Root Mean Squared Error)} = \sqrt{MSE}$$

Hasil prediksi dari kedua model divisualisasikan dengan membandingkan data aktual terhadap hasil prediksi dalam satu grafik. Visualisasi ini memberikan wawasan intuitif tentang tingkat kecocokan model terhadap realitas. Model dengan garis prediksi yang paling mendekati garis aktual dianggap lebih akurat.

Dalam studi kasus ini, ARIMA menunjukkan performa yang baik untuk data dengan pola musiman dan linier, sementara LSTM unggul dalam mengenali fluktuasi yang tidak reguler dan non-linier, terutama ketika pola permintaan tidak mudah dijelaskan oleh faktor tunggal.

Dengan membandingkan kedua metode ini, penelitian ini tidak hanya menguji performa prediksi, tetapi juga menyajikan strategi pemodelan yang berbeda. Model ARIMA memberikan dasar statistik yang kuat, sementara LSTM membawa kekuatan adaptasi dari *deep learning*. Integrasi antara keduanya juga menjadi potensi untuk penelitian lanjutan.

3. HASIL DAN PEMBAHASAN

Dataset yang digunakan mencakup informasi harian terkait permintaan inventaris (demand), biaya produksi (production_cost), harga jual (selling_price), dan margin keuntungan (profit_margin) selama kurun waktu yang konsisten. Fokus utama penelitian adalah memodelkan permintaan (demand) sebagai variabel target untuk diprediksi.

Data awal dimuat ke dalam program menggunakan pustaka pandas. Tanggal diubah ke format datetime dan dijadikan indeks time series agar sesuai dengan kebutuhan analisis deret waktu. Proses ini penting karena jaringan LSTM mengandalkan urutan temporal dalam data input.

Visualisasi tren permintaan dilakukan untuk mengamati pola musiman atau outlier sebelum pemodelan. Dalam konteks ini, pola yang fluktuatif dan tidak sepenuhnya linier mengindikasikan bahwa LSTM lebih cocok dibanding ARIMA, karena LSTM dapat menangani hubungan non-linear dan jangka panjang.

Data permintaan kemudian dinormalisasi ke skala [0,1] menggunakan MinMaxScaler. Ini diperlukan karena jaringan saraf sangat sensitif terhadap skala input. Skala yang seragam mempercepat proses pelatihan dan meningkatkan kestabilan model.

Dataset diubah ke dalam format sekuensial, di mana setiap input terdiri dari 10 langkah waktu sebelumnya (time_step=10) untuk memprediksi permintaan pada langkah berikutnya. Ini membentuk himpunan fitur X dan target y dalam format tiga dimensi sesuai input LSTM: [samples, time steps, features].

Arsitektur jaringan LSTM kemudian dibangun berdasarkan diagram fungsional gerbang yang dilampirkan. Model memiliki dua lapisan LSTM, masing-masing dengan 50 unit neuron dan diikuti oleh lapisan Dropout untuk mengurangi overfitting. Output akhir diproses melalui Dense layer dengan satu neuron (regresi).

Dalam implementasi LSTM ini, forget gate digunakan untuk menentukan informasi mana dari status sebelumnya yang akan dihapus, menggunakan fungsi sigmoid. Ini membantu dalam menyaring informasi yang tidak relevan dari f_t .

Input gate bekerja dengan mengontrol informasi baru yang masuk ke sel memori. Fungsi sigmoid dan tanh digunakan untuk memilih dan mentransformasikan input sebelum diperbarui dalam cell state.

Candidate cell menghasilkan informasi baru yang berpotensi ditambahkan ke memori internal. Informasi ini berperan penting dalam memperbarui status konteks temporal yang relevan.

Output gate mengontrol informasi mana dari sel memori yang akan dikeluarkan dan dijadikan sebagai hidden state saat ini (h_t). Ini memungkinkan jaringan untuk hanya memproyeksikan informasi penting yang akan digunakan untuk prediksi selanjutnya.

Optimasi model dilakukan menggunakan algoritma Adam, yaitu kombinasi dari RMSprop dan momentum. Adam dipilih karena kemampuannya untuk menyesuaikan laju pembelajaran secara adaptif untuk setiap parameter, yang sangat bermanfaat dalam pelatihan model LSTM yang kompleks.

Model dilatih selama 100 epoch dengan batch_size sebesar 32. Selama pelatihan, fungsi loss yang digunakan adalah mean squared error (MSE) karena kita sedang mengerjakan regresi nilai permintaan.

Setelah pelatihan selesai, model digunakan untuk memprediksi data uji. Hasil prediksi kemudian di-inverse transform agar kembali ke skala permintaan asli, memungkinkan interpretasi dan evaluasi yang relevan terhadap nilai sebenarnya.

Visualisasi hasil prediksi dibandingkan dengan data aktual menunjukkan bahwa LSTM mampu menangkap dinamika fluktuatif dalam permintaan harian secara cukup akurat. Kurva prediksi mengikuti pola aktual dengan penyimpangan minimum pada sebagian besar titik waktu.

Hasil Evaluasi Model

Evaluasi kedua model ini menggunakan 2 dataset permintaan inventaris sejumlah 100 record dan 200 record. Untuk mengolah dataset ini menggunakan internet python notebook(ipynb) dengan google colab dan memanfaatkan tool gemini. Berdasarkan hasil eksekusi dengan sejumlah 100 record ini menunjukkan bahwa Model LSTM menghasilkan MSE = 2109.72 sedangkan RMSE mencapai 45.93%. Model ARIMA dengan order (1, 1, 1) menunjukkan MSE = 2080.40 dan RMSE menghasilkan 45.61%.

Dengan demikian perbandingan dan evaluasi Model ARIMA memiliki MSE (2080.40) yang sedikit lebih rendah dibandingkan model LSTM (2109.72). Ini menunjukkan bahwa rata-rata kuadrat error prediksi ARIMA lebih kecil. Sedangkan Model ARIMA juga memiliki RMSE 45.61% yang sedikit lebih rendah dibandingkan model LSTM (RMSE=45.93%). Maka RMSE memberikan gambaran yang lebih intuitif mengenai rata-rata selisih prediksi dalam unit data asli. Dalam kasus ini, rata-rata error prediksi ARIMA sedikit lebih kecil dibandingkan LSTM.

Berdasarkan metrik MSE dan RMSE pada dataset sejumlah 100 record, model ARIMA (dengan order (1, 1, 1)) menunjukkan performa yang sedikit lebih baik dibandingkan model LSTM, karena memiliki nilai MSE dan RMSE yang lebih rendah. Ini berarti, pada dataset ini, model ARIMA rata-rata membuat kesalahan prediksi yang sedikit lebih kecil.

Kemudian berdasarkan hasil eksekusi dengan dataset sejumlah 200 record menunjukkan bahwa Model LSTM menghasilkan MSE yaitu 2453.76 dan RMSE = 49.54%. Sedangkan Model ARIMA dengan orde(1, 1, 1) menghasilkan MSE yaitu 2000.18 dan RMSE= 44.72%. Dari hasil Perbandingan dan Evaluasi kedua model ini yaitu Model ARIMA memiliki MSE (2000.18) yang jauh lebih rendah dibandingkan model LSTM (2453.76). Ini menunjukkan bahwa rata-rata kuadrat error prediksi ARIMA lebih kecil pada dataset ini. Model ARIMA juga memiliki RMSE =44.72% yang lebih rendah dibandingkan model LSTM (RMSE=49.54%). Ini berarti rata-rata selisih prediksi ARIMA lebih kecil dalam unit data asli. Dengan demikian pada dataset sejumlah 200 record ini model ARIMA (dengan order (1, 1, 1)) menunjukkan performa yang lebih baik secara signifikan dibandingkan model LSTM, karena memiliki nilai MSE dan RMSE yang jauh lebih rendah. Hasil perbandingan dan evaluasi kedua model ini dapat dilihat pada table 1. Hasil evaluasi menunjukkan bahwa model ARIMA memiliki MSE yang lebih rendah dibandingkan LSTM, terutama pada dataset tertentu [11].

Tabel 1 Hasil Perbandingan dan Evaluasi Kedua Model

Model	Dataset (record)	MSE	RMSE (%)	Keterangan	Kesimpulan
LSTM	100	2109.72	45.93	Semakin meningkat RMSEnya	Performa prediksi ARIMA lebih baik dari pada LSTM
	200	2453.76	49.54		
ARIMA (1,1,1)	100	2080.40	45.61	Semakin menurun RMSEnya	
	200	2000.18	44.72		

4. KESIMPULAN

Dari hasil eksperimen ini menunjukkan bahwa model ARIMA lebih baik performanya dari pada model LTSM meskipun seperti ditunjukkan table 1 dengan jumlah dataset yang berbeda yaitu 100 dan 200 record. Performa model sangat bergantung pada karakteristik data dan tuning parameter model itu sendiri atau dengan dataset yang berbeda. Ini menunjukkan bahwa karakteristik data dan tuning parameter model sangat berpengaruh terhadap hasil prediksi [12]. Penelitian lebih lanjut dapat mengeksplorasi optimasi model dan integrasi fitur eksternal untuk meningkatkan akurasi prediksi.

Penting juga untuk menyelidiki alasan di balik perbedaan performa yang signifikan antara ARIMA dan LSTM pada dataset ini. ARIMA adalah model statistik linier yang dirancang khusus untuk data deret waktu dengan asumsi tertentu mengenai stasioneritas dan autokorelasi. Di sisi lain, LSTM adalah model non-linier yang dapat menangkap pola temporal yang kompleks. Performa superior ARIMA dalam kasus ini mungkin mengindikasikan bahwa pola dalam dataset sejumlah 200 record lebih bersifat linier atau bahwa data tersebut relatif "bersih" tanpa terlalu banyak noise atau anomali yang memerlukan kemampuan pemodelan non-linier LSTM. Analisis karakteristik data yang lebih mendalam, termasuk pemeriksaan outlier dan komponen tren/musiman, dapat memberikan wawasan lebih lanjut.

5. SARAN

Ada beberapa arah penelitian lebih lanjut yang dapat dieksplorasi untuk memperdalam analisis dan potensi peningkatan akurasi peramalan permintaan inventaris.

Salah satu area utama untuk penelitian lanjutan adalah optimasi model. Meskipun ARIMA (1,1,1) menunjukkan hasil yang baik, penting untuk mengeksplorasi order ARIMA lain (nilai p, d, q) yang mungkin memberikan *fit* yang lebih baik pada data. Pendekatan sistematis seperti pencarian grid (grid search) atau penggunaan fungsi *auto_arima* dari pustaka *pmdarima* dapat digunakan untuk secara otomatis mengidentifikasi order ARIMA terbaik berdasarkan kriteria informasi seperti AIC. Selain itu, mempertimbangkan model SARIMA (*Seasonal ARIMA*) bisa relevan jika data permintaan menunjukkan pola musiman yang jelas, yang mungkin tidak sepenuhnya ditangkap oleh model ARIMA non-musiman.[13][14]

Untuk model LSTM, penelitian lebih lanjut dapat fokus pada peningkatan arsitektur dan parameter model. Arsitektur LSTM yang digunakan saat ini relatif sederhana. Mengeksplorasi jumlah layer LSTM yang lebih banyak, jumlah unit (neuron) di setiap layer, penggunaan bidirectional LSTM, atau penambahan layer dense tambahan dapat meningkatkan kemampuan model untuk mempelajari pola kompleks dalam data. Optimasi hyperparameter seperti *learning rate*, ukuran *batch*, dan *dropout rate* juga krusial untuk memaksimalkan performa model LSTM.[15] Penggunaan teknik *early stopping* dan *regularization* dapat membantu mencegah *overfitting*. [16]

Terakhir, penelitian dapat diperluas dengan memasukkan fitur eksternal (exogenous variables) yang relevan ke dalam model. Dataset saat ini hanya menggunakan data permintaan itu sendiri. Namun, faktor-faktor seperti harga produk, aktivitas promosi, hari libur, kondisi ekonomi, atau bahkan cuaca dapat memengaruhi permintaan inventaris. Menggabungkan fitur-fitur ini ke dalam model (misalnya, menggunakan model ARIMAX atau LSTM dengan multiple input) dapat meningkatkan akurasi prediksi secara signifikan karena model dapat mempertimbangkan faktor-faktor eksternal yang mendorong perubahan permintaan. Perbandingan performa model dengan dan tanpa fitur eksternal akan menjadi kontribusi penelitian yang berharga.

DAFTAR PUSTAKA

- [1] Iwa Sungkawa dan Ries Tri Megasari, "Penerapan Ukuran Ketepatan Nilai Ramalan Data Deret Waktu dalam Seleksi Model Peramalan Volume Penjualan PT Satriamandiri Citramulia", *Jurnal ComTech*, Vol.2 No. 2, Desember 2011: 636-645.
- [2] Hyndman, R. J., & Athanasopoulos, G., "*Forecasting: Principles and Practice*. OTexts", 2018.
- [3] Hochreiter, S., & Schmidhuber, J., "Long Short-Term Memory", *Neural Computation*, 9(8), p.1735–1780. 1997.
- [4] Choi, H., Lee, H., & Kang, K., "Predicting Product Demand in E-Commerce using RNN", *Journal of Retailing and Consumer Services*, 55, 2020.
- [5] Hamid, U., et al., "Time Series Forecasting using ARIMA Model: A Case Study of Retail Sales. *International Journal of Data Science and Analytics*", 9(2), 75–8, 2020.
- [6] Zhang, G. P., et al., "A Survey on Time Series Forecasting", *IEEE Transactions on Neural Networks and Learning Systems*, 30(7), p. 2050–2071. 2019.
- [7] Fischer, T., & Krauss, C. "Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions", *Expert Systems with Applications*, 115, p. 257–272, 2018.
- [8] Mao, Y., et al., "A Novel ARIMA-Based Approach for Energy Demand Forecasting", *Energy*, 138, p.1111–1121, 2018.
- [9] Box, G. E. P., Jenkins, G. M., & Reinsel, G. C., "*Time Series Analysis: Forecasting and Control*", Wiley, 2015).
- [10] Adhitio Satyo Bayangkari Karno, "Analisis Data Time Series Menggunakan LSTM (Long Short Term Memory) Dan ARIMA (Autocorrelation Integrated Moving Average) Dalam Bahasa Python", *Ultima InfoSys: Jurnal Ilmu Sistem Informasi*, 11(1), 1-7, 2020.
- [11] Makridakis, S., Spiliotis, E., & Assimakopoulos, V., "*M4 Competition: Results, conclusions and way forward*", *International Journal of Forecasting*, 34(4), 802-804, 2018.
- [12] Wang, Y., & Li, Y., "*A hybrid model for forecasting financial time series based on LSTM and ARIMA*", *Journal of Computational and Applied Mathematics*, p. 350, 123-133, 2019.
- [13] Sari, N. A., & Wibowo, M. A., "Peramalan Penjualan Motor Menggunakan Model SARIMA", *Jurnal Teknik Informatika Polinema*, 6(2), p. 121–128, 2020. ISSN: 2502-1370,
- [14] Assaad, M., Bonnet, S., & Abdellatif, T., "SARIMA Model for Forecasting Seasonal Demand of Electricity Consumption", *Energy Reports*, 7, 961–971, 2021.
- [15] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R., "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", *Journal of Machine Learning Research*, 15, p. 1929–1958, 2014. ISSN: 1532-4435
- [16] Goodfellow, I., Bengio, Y., & Courville, A., "Deep Learning", MIT Press, 2016. ISBN: 9780262035613.