

## Implementasi Algoritma LSTM Pada Sistem Monitoring Iot Untuk Penanganan Resiko Kebakaran

Ahmad Nawawi<sup>1</sup>, Kamarudin<sup>2</sup>, Finki Dona Marleny<sup>3</sup>

<sup>1,2,3</sup> Prodi S1 Informatika, Universitas Muhammadiyah Banjarmasin, Indonesia

\*<sup>1</sup>ahmad\_nawawi\_2155201110030@umbjm.ac.id, <sup>2</sup>kamarudin.skom@gmail.com, <sup>3</sup>finkidona@umbjm.ac.id

### ABSTRACT

This research implements a Long Short-Term Memory (LSTM) algorithm to predict temperature patterns as part of an IoT-based monitoring system designed to mitigate fire risks. The dataset is collected from ESP32-based sensors that record temperature and timestamp data. The LSTM model was trained using normalized temperature data, with five-step ahead predictions. Preprocessing included combining date and time into a datetime index, followed by scaling and reshaping data for supervised learning. The model architecture consists of a single LSTM layer and a dense output layer. The prediction results show a low Mean Squared Error (MSE), indicating the LSTM model is effective for early detection of potential fire hazards. This work contributes to real-time risk mitigation by improving the predictive accuracy in IoT environments.

**Keywords:** IoT, LSTM, Fire Risk, Temperature Forecasting, Deep Learning

### ABSTRAK

Penelitian ini mengimplementasikan algoritma Long Short-Term Memory (LSTM) untuk memprediksi pola suhu sebagai bagian dari sistem monitoring berbasis IoT dalam rangka mitigasi risiko kebakaran. Dataset diperoleh dari sensor berbasis ESP32 yang merekam data suhu dan waktu. Model LSTM dilatih menggunakan data suhu yang telah dinormalisasi dengan prediksi lima langkah ke depan. Pra-pemrosesan meliputi penggabungan data tanggal dan waktu menjadi indeks waktu, kemudian dilanjutkan dengan normalisasi dan pembentukan data dalam format pembelajaran terawasi. Arsitektur model terdiri dari satu lapisan LSTM dan satu lapisan keluaran dense. Hasil prediksi menunjukkan nilai Mean Squared Error (MSE) yang rendah, menandakan efektivitas model LSTM dalam mendeteksi potensi bahaya kebakaran secara dini. Penelitian ini berkontribusi pada upaya mitigasi risiko secara real-time melalui peningkatan akurasi prediksi pada lingkungan IoT.

**Kata Kunci :** IoT, LSTM, Risiko Kebakaran, Prediksi Suhu, Deep Learning

### INFORMASI ARTIKEL

**Submit**  
10, Mei 2025

**Diterima**  
15, Juni 2025

**Publish Online**  
30, November 2025

## PENDAHULUAN

Penggunaan teknologi *Internet of Things* (IoT) dalam sistem monitoring telah berkembang pesat, terutama dalam bidang pemantauan lingkungan. Secara umum IoT diartikan sebagai rangkaian untuk menghubungkan alat atau objek cerdas agar dapat berinteraksi dengan objek lainnya melalui jaringan internet [1]. Sistem monitoring berbasis IoT dapat mengumpulkan data secara langsung dari berbagai sensor yang diletakkan di lokasi tertentu. Data yang dikumpulkan, seperti suhu, kelembapan, dan konsentrasi gas, dapat digunakan untuk mendeteksi kondisi abnormal yang berpotensi berbahaya, seperti kebakaran.

Kebakaran adalah kondisi dimana bangunan atau lingkungan di sekitar terbakar oleh api [2]. Pada umumnya sistem yang dirancang adalah dengan pendekatan ambang batas, di mana sistem akan memberikan peringatan jika nilai suhu melebihi batas tertentu. Meskipun metode ini sederhana dan mudah diimplementasikan, pendekatan tersebut tidak cukup efektif untuk mendeteksi tren perubahan suhu yang mengarah pada potensi bahaya secara dini. Oleh karena itu, dibutuhkan metode yang lebih adaptif dan mampu mengenali pola data berdasarkan waktu.

Metode yang banyak diterapkan dalam analisis data runtun waktu adalah algoritma pembelajaran mesin, khususnya *Long Short-Term Memory* (LSTM). LSTM merupakan varian dari unit Recurrent Neural Network (RNN). LSTM sangat cocok bagi prediksi pada data deret waktu karena mampu menangani jeda waktu yang tidak dapat ditebak antara peristiwa-peristiwa penting dalam deret data [3]. Kemampuan LSTM dalam mengingat informasi historis membuatnya unggul dalam memprediksi tren berdasarkan data masa lalu.

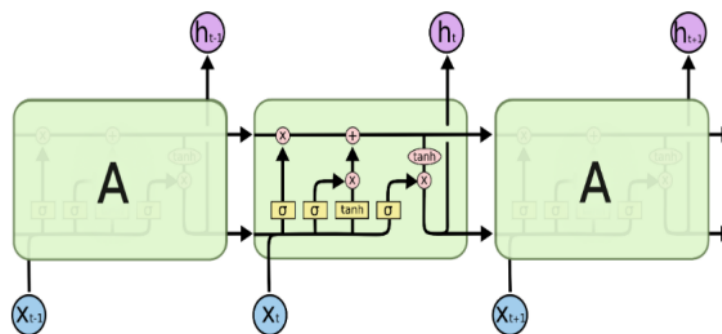
LSTM secara prinsip dasar dapat diterapkan dengan perangkat IoT dalam berbagai bidang, termasuk prediksi suhu, deteksi anomali, dan pengawasan lingkungan berbasis IoT [4]. Kombinasi antara kemampuan LSTM dan data dari sensor IoT membuka peluang untuk membangun sistem deteksi dini yang lebih cerdas dan responsif. Dengan menerapkan model prediksi suhu berbasis LSTM, sistem monitoring tidak hanya dapat mendeteksi kondisi saat ini, tetapi juga memperkirakan kondisi mendatang yang berpotensi membahayakan.

Tinjauan ini menunjukkan pentingnya pemanfaatan algoritma untuk memprediksi data berbentuk time series dalam sistem monitoring IoT [5]. Penelitian ini mencoba melanjutkan upaya tersebut dengan merancang dan mengimplementasikan model prediksi berbasis LSTM menggunakan data dari sensor IoT, sebagai langkah preventif terhadap potensi risiko kebakaran.

## TINJAUAN PUSTAKA

### A. Algoritma LSTM

Long Short Term Memory (LSTM) merupakan algoritma dengan kemampuan mumpuni dalam membangun model prediksi. LSTM merupakan turunan dari Recurrent Neural Network (RNN) sebuah metode yang di desain untuk mengolah data sequence. RNN memiliki masalah vanishing dan exploding gradient. LSTM dibangun untuk mengatasi masalah gradien menghilang pada RNN ketika menghadapi vanishing dan exploding gradien [6]. Arsitektur LSTM terdiri dari tiga lapisan, yaitu lapisan input, lapisan tersembunyi, dan lapisan output seperti pada gambar 1.



Gambar 1. Arsitektur Algoritma LSTM

Lapisan tersembunyi memiliki tiga gerbang memory cell , yaitu forget gate, input gate, dan output gate [7]. Untuk rumus forget gate di definisikan dengan rumus (1) sebagai berikut.

$$F_t = \sigma(W_f \times [h_{t-1}, X_t] + b_f) \tag{1}$$

Gerbang pertama dalam arsitektur LSTM dikenal sebagai *forget gate*, yang berfungsi untuk menyaring informasi pada *cell state* menggunakan Persamaan 1. Jika nilai *forget gate* mendekati 0, maka informasi tersebut akan dihapus dari *cell state*, sedangkan jika mendekati 1, informasi tersebut akan dipertahankan. [8].

Terdapat 2 gates yang akan di jalankan pada input Gate, memori jangka panjang kandidat dalam keadaan sel saat ini  $C_t$  dan tingkat penyimpanan  $i_t$  dihitung menggunakan persamaan (2) dan (3) [9].

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

$$C_t = \tanh(W_c \times [h_{t-1}, X_t] + b_c) \tag{3}$$

Persamaan (4) dan (5) adalah rumus hasil akhir dari LSTM yang mana  $h_t$  adalah hasil akhir,  $O_t$  merupakan nilai output gate, dan  $C_t$  merupakan nilai memory cell state [10].

$$O_t = \sigma(W_o \times [h_{t-1}, X_t] + b_o) \tag{4}$$

$$h_t = O_t \times \tanh(C_t) \tag{5}$$

B. Metric Evaluation

1) Mean Absolute Error (MAE)

Mean Absolute Error (MAE) merupakan sebuah indikator yang menentukan nilai tengah kesalahan mutlak antara nilai aktual dan hasil prediksi. Perhitungan MAE diberikan pada Persamaan (6) [11].

$$MAE = \frac{1}{n} \sum_{i=1}^n |e_i| \tag{6}$$

2) Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE) digunakan untuk menghitung besaran error dalam memprediksi suatu data. RMSE menghitung perbedaan antara nilai sesungguhnya dan nilai yang diharapkan dan membagi hasil total penjumlahan yang diperoleh dengan banyaknya waktu prediksi dan menarik akarnya. Perhitungan RMSE diberikan pada Persamaan (7) [12].

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (X_t - F_t)^2}{n}} \quad (7)$$

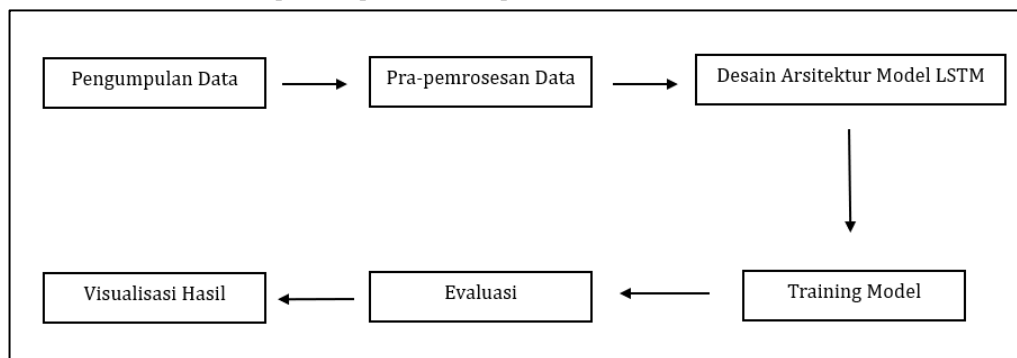
### 3) R-Squared

R Squared ( $R^2$ ) menunjukkan seberapa baik sebuah model regresi dapat menjelaskan atau memprediksi nilai data yang diamati (variabel dependen) berdasarkan variabel-variabel lain (variabel independen). Nilai  $R^2$  berada di antara 0 sampai 1, yang menggambarkan seberapa besar pengaruh keseluruhan variabel independen terhadap variabel dependen. Perhitungan nilai ini bisa dilihat pada rumus di Persamaan (8). [13].

$$R^2 = 1 - \frac{\sum_{i=1}^n (X_t - F_t)^2}{\sum_{i=1}^n (Y_t - F_t)^2} \quad (8)$$

## METODE PENELITIAN

Penelitian ini terbagi menjadi beberapa tahapan penelitian yaitu pengumpulan data, pra-pemrosesan data, desain arsitektur model LSTM, training model, evaluasi, dan visualisasi hasil. Alur tahapan dapat di lihat pada Gambar 2.



Gambar 2. Tahapan Penelitian

#### A. Pengumpulan Data

Data diperoleh dari sistem IoT berbasis mikrokontroler ESP32 yang terhubung dengan sensor suhu dan waktu. Sensor mencatat suhu lingkungan dan menyimpannya dalam format tabular dengan kolom *date*, *time*, dan *temperature*. Dataset ini diunduh dalam bentuk file CSV dengan nama *tabel\_esp32\_record.csv* untuk diolah lebih lanjut.

#### B. Pra-pemrosesan Data

Langkah awal pra-pemrosesan dilakukan dengan menggabungkan kolom *date* dan *time* menjadi satu kolom *datetime*. Kolom ini kemudian diatur sebagai indeks dan data diurutkan berdasarkan waktu. Selanjutnya, hanya kolom *temperature* yang digunakan sebagai fokus prediksi.

Data suhu kemudian dinormalisasi dengan cara Min-Max Scaling ke dalam kisaran  $[0, 1]$  untuk mempercepat konvergensi saat pelatihan model. Setelah itu, data dikonversi menjadi bentuk sekuensial menggunakan pendekatan *sliding window*, di mana setiap 10 data suhu digunakan untuk memprediksi suhu pada 5 langkah waktu berikutnya.

#### C. Desain Arsitektur Model LSTM

Model prediksi suhu dalam penelitian ini dirancang menggunakan arsitektur jaringan saraf dalam yang berbasis Long Short-Term Memory (LSTM). Arsitektur dibangun menggunakan framework TensorFlow dan Keras, yang menyediakan

kemudahan dalam membentuk model deep learning secara modular. Model terdiri dari satu lapisan LSTM yang berfungsi sebagai inti pemrosesan data sekuensial. Lapisan ini memiliki 64 unit neuron yang mampu menangkap hubungan temporal atau pola berurutan dalam data suhu. Dengan struktur memori internalnya, LSTM memungkinkan model untuk mengingat informasi penting dari langkah waktu sebelumnya, sehingga sangat sesuai untuk menangani prediksi berbasis time series seperti data suhu lingkungan.

Setelah lapisan LSTM, model dilanjutkan dengan satu lapisan Dense (fully connected) yang berfungsi sebagai lapisan output. Lapisan ini terdiri dari lima neuron, yang masing-masing menghasilkan prediksi suhu untuk lima langkah waktu ke depan. Model ini dikompilasi menggunakan fungsi kerugian Mean Squared Error (MSE), karena MSE efektif dalam mengukur akurasi prediksi pada permasalahan regresi kontinu. Untuk optimasi, digunakan algoritma Adam Optimizer yang dikenal cepat dan stabil dalam menemukan nilai bobot optimal pada jaringan saraf dalam. Seluruh arsitektur dirancang sesederhana mungkin untuk menjaga efisiensi komputasi, mengingat implementasinya ditujukan untuk sistem monitoring real-time berbasis IoT yang umumnya memiliki keterbatasan sumber daya.

#### D. Training Model

Setelah arsitektur model selesai dirancang, proses pelatihan dilakukan untuk menyesuaikan bobot-bobot jaringan saraf agar model mampu mengenali pola hubungan antara data suhu sebelumnya dengan suhu di masa depan. Dataset yang telah dinormalisasi dan dikonversi ke dalam format time series terbagi menjadi dua, yaitu data latih dan data validasi. Data latih digunakan untuk memperbarui bobot model melalui proses pembelajaran, sementara data validasi berfungsi untuk mengukur kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya. Model dilatih selama 50 epoch, di mana setiap epoch memproses seluruh data latih satu kali penuh. Untuk efisiensi komputasi, proses pelatihan menggunakan batch size sebesar 32, yang berarti data dibagi dalam kelompok kecil selama iterasi pembaruan bobot. Fungsi kerugian yang digunakan adalah Mean Squared Error (MSE), yang efektif dalam mengukur rata-rata kesalahan prediksi numerik.

Selama pelatihan berlangsung, nilai kerugian (loss) pada data latih dan validasi dipantau untuk memastikan bahwa model tidak mengalami overfitting. Penurunan nilai loss yang konsisten menunjukkan bahwa model berhasil mempelajari pola data dengan baik. Jika terjadi perbedaan signifikan antara nilai loss data latih dan validasi, hal ini dapat menjadi indikator bahwa model terlalu menyesuaikan diri pada data pelatihan. Oleh karena itu, pemantauan performa dilakukan secara berkala sepanjang proses pelatihan. Setelah pelatihan selesai, model diuji terhadap data baru untuk mengukur akurasi prediksi suhu lima langkah ke depan. Model yang telah terlatih ini selanjutnya digunakan dalam skenario simulasi real-time untuk mengantisipasi potensi kenaikan suhu yang dapat mengindikasikan awal mula risiko kebakaran.

#### E. Evaluasi

Evaluasi performa model dilakukan menggunakan tiga metrik utama, yaitu Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), dan R-squared ( $R^2$ ). MAE mengukur rata-rata selisih absolut antara nilai aktual dan hasil prediksi, yang memberikan gambaran umum tentang besar kesalahan prediksi tanpa memperhitungkan arah selisih tersebut. Nilai MAE yang kecil menunjukkan bahwa rata-rata kesalahan prediksi model relatif rendah terhadap nilai suhu sebenarnya.

Sementara itu, RMSE memberikan bobot lebih besar pada kesalahan yang besar karena menggunakan kuadrat selisih, sehingga lebih sensitif terhadap outlier. RMSE yang rendah menunjukkan bahwa kesalahan besar jarang terjadi selama proses prediksi. Metrik ketiga, yaitu R-squared (koefisien determinasi), berfungsi untuk mengukur seberapa baik model menjelaskan keragaman data target. Jika nilai mendekati 1 berarti bahwa model mampu mendefinisikan sebagian besar variasi data aktual dengan akurat. Dengan menggunakan ketiga metrik ini secara bersamaan, evaluasi model menjadi lebih komprehensif, karena mampu menangkap baik tingkat kesalahan rata-rata, sensitivitas terhadap kesalahan besar, maupun kemampuan penyesuaian model terhadap pola data aktual.

#### F. Visualisasi Hasil

Selain evaluasi numerik, hasil prediksi juga divisualisasikan dalam bentuk grafik time series untuk memberikan gambaran visual mengenai perbandingan antara suhu aktual dan suhu hasil prediksi. Visualisasi ini bertujuan untuk menilai sejauh mana model dapat mengikuti pola pergerakan suhu pada data uji, terutama pada perubahan suhu yang signifikan. Grafik tersebut memperlihatkan deret waktu suhu aktual dan lima langkah prediksi ke depan, yang ditampilkan secara bersamaan agar perbedaan tren dapat diamati dengan jelas. Melalui pendekatan ini, dapat dievaluasi secara kualitatif apakah model berhasil mendeteksi tren naik atau turun secara konsisten. Visualisasi juga menjadi alat penting dalam mendukung interpretasi hasil prediksi, karena memudahkan pengguna untuk memahami performa model secara intuitif dalam aplikasi monitoring real-time.

## HASIL DAN PEMBAHASAN

#### A. Metrik Evaluasi

Setelah model LSTM dilatih menggunakan data suhu dari sistem monitoring berbasis ESP32, langkah berikutnya adalah menguji model tersebut dengan data uji untuk melihat seberapa baik hasil prediksinya. Dari hasil evaluasi menggunakan beberapa metrik performa, model ini menunjukkan hasil yang cukup memuaskan. Nilai Mean Absolute Error (MAE) yang rendah berarti rata-rata perbedaan antara prediksi dan suhu sebenarnya cukup kecil. Selain itu, nilai Root Mean Squared Error (RMSE) yang juga rendah menunjukkan bahwa kesalahan besar dalam prediksi jarang terjadi. Semua ini menandakan bahwa model mampu memberikan prediksi suhu yang akurat dan mendekati nilai asli dalam waktu nyata. Ditambah lagi, nilai R-squared ( $R^2$ ) yang hampir mencapai 1 memperkuat bahwa model sangat efektif dalam menangkap dan menjelaskan perubahan suhu berdasarkan data historis yang ada.

```
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

# Evaluasi Langsung pada data diskalakan
mae = mean_absolute_error(y_test[:, 0], predictions[:, 0])
rmse = np.sqrt(mean_squared_error(y_test[:, 0], predictions[:, 0]))
r2 = r2_score(y_test[:, 0], predictions[:, 0])

print(f"Mean Absolute Error (MAE): {mae:.4f}")
print(f"Root Mean Squared Error (RMSE): {rmse:.4f}")
print(f"R2 Score: {r2:.4f}")
```

Gambar 3. Code program perhitungan matrik evaluasi

Kode Jupyter Notebook di atas menggambarkan penilaian seberapa baik model LSTM dalam memprediksi data suhu yang sudah diskalakan. Penilaian ini memakai tiga ukuran, yaitu Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), dan R<sup>2</sup> Score. Hasilnya menunjukkan nilai MAE dan RMSE yang cukup rendah, artinya kesalahan prediksi suhu dibandingkan data sebenarnya tidak besar. Selain itu, nilai R<sup>2</sup> yang hampir mencapai 1 menandakan bahwa model ini cukup mampu menangkap pola dan variasi dari data suhu dengan baik.

```
Mean Absolute Error (MAE): 0.0012
Root Mean Squared Error (RMSE): 0.0014
R2 Score: 0.9988
```

Gambar 4. Hasil merik evaluasi

Hasil evaluasi kemudian ditampilkan dengan format angka desimal empat digit melalui fungsi print() seperti pada gambar 4. Evaluasi ini penting untuk menilai sejauh mana model LSTM berhasil memprediksi nilai suhu (atau target lainnya) dengan akurat setelah proses pelatihan selesai.

## B. Visualisasi Hasil Prediksi

Untuk mendukung hasil evaluasi tersebut, dilakukan pula visualisasi hasil prediksi suhu terhadap data aktual. Grafik yang ditampilkan menunjukkan bahwa tren prediksi model mengikuti pola perubahan suhu dengan cukup baik. Prediksi lima langkah ke depan berhasil mengantisipasi kenaikan dan penurunan suhu secara konsisten, meskipun pada beberapa titik terdapat sedikit deviasi yang masih dalam batas toleransi. Hasil ini menunjukkan bahwa model memiliki tingkat keakuratan yang cukup tinggi dan dapat diandalkan untuk diterapkan dalam sistem monitoring suhu berbasis IoT. Dengan adanya prediksi suhu beberapa langkah ke depan, sistem dapat memberikan peringatan dini sebelum suhu mencapai ambang batas kritis, sehingga memberikan waktu yang cukup untuk melakukan mitigasi risiko kebakaran.

```
# 9. Visualisasi hasil
plt.figure(figsize=(10, 6))
plt.plot(y_test_rescaled[:, 0], label='Actual')
plt.plot(predictions_rescaled[:, 0], label='Predicted')
# plt.title('Prediksi Temperature - 1 dari 5 Langkah ke depan')
plt.xlabel('Time Step')
plt.ylabel('Temperature')
plt.legend()
plt.grid(True)
plt.show()
```

Gambar 5. Code program untuk visulisasi hasil

Kode Jupyter Notebook di atas digunakan untuk memvisualisasikan hasil prediksi model terhadap data suhu aktual setelah proses pembalikan skala (rescaling). Fungsi `plt.figure(figsize=(10, 6))` digunakan untuk mengatur ukuran plot agar lebih jelas dan proporsional. Dua garis plot kemudian digambarkan menggunakan `plt.plot()`, di mana `y_test_rescaled[:, 0]` merepresentasikan data suhu aktual dan `predictions_rescaled[:, 0]` merepresentasikan hasil prediksi dari model. Kedua garis ini diberi label "Actual" dan "Predicted" untuk membedakan keduanya dalam legenda. Meskipun judul grafik dikomentari (`# plt.title(...)`), sumbu x dan y tetap diberi label dengan `plt.xlabel("Time Step")` dan `plt.ylabel("Temperature")`, yang menunjukkan bahwa data ini direpresentasikan terhadap waktu. Fungsi `plt.legend()` menampilkan legenda untuk menjelaskan garis yang ditampilkan, dan `plt.grid(True)` menambahkan grid pada grafik agar lebih mudah membaca nilai-nilainya. Terakhir, `plt.show()` digunakan untuk menampilkan grafik secara keseluruhan.



Gambar 6. Visualisasi hasil

Pada Gambar 6 di atas menunjukkan hasil dari prediksi algoritma LSTM. Visualisasi ini penting untuk mengevaluasi secara intuitif sejauh mana hasil prediksi mendekati nilai aktual, serta untuk mengidentifikasi pola atau kesalahan prediksi model dari waktu ke waktu.

## KESIMPULAN DAN SARAN

Penelitian ini telah berhasil mengimplementasikan algoritma Long Short-Term Memory (LSTM) pada sistem monitoring IoT untuk memprediksi suhu lingkungan sebagai upaya mitigasi risiko kebakaran. Model dirancang dengan memanfaatkan data suhu yang dikumpulkan secara real-time menggunakan sensor berbasis ESP32. Hasil pelatihan dan evaluasi menunjukkan bahwa model LSTM mampu memprediksi suhu lima langkah ke depan dengan tingkat akurasi yang tinggi, ditunjukkan oleh nilai MAE dan RMSE yang rendah serta nilai R-squared yang mendekati 1. Visualisasi hasil prediksi juga menunjukkan bahwa model dapat mengikuti tren perubahan suhu secara konsisten dan mendekati nilai aktual.

Dengan kemampuan prediktif tersebut, sistem ini dapat menjadi alat bantu penting dalam memberikan peringatan dini terhadap potensi kebakaran, khususnya di area-area yang rawan terhadap peningkatan suhu ekstrem. Model ini juga dapat dikembangkan lebih lanjut untuk diintegrasikan dengan sistem pengambilan keputusan otomatis yang dapat mengaktifkan alarm atau sistem pemadaman dini saat deteksi suhu mencurigakan terkonfirmasi oleh prediksi.

Sebagai saran untuk pengembangan selanjutnya, penelitian ini dapat diperluas dengan menambahkan parameter lingkungan lain seperti kelembapan dan konsentrasi gas berbahaya untuk meningkatkan akurasi dan reliabilitas sistem. Selain itu, pengujian pada kondisi lapangan yang lebih kompleks dan penguatan fitur real-time pada implementasi IoT juga menjadi langkah penting dalam meningkatkan kesiapan sistem untuk digunakan secara praktis di lingkungan nyata.

## DAFTAR PUSTAKA

- [1] F. Trisnawati and P. Prasetyawan, "SEMMUDIK : Selamat Mudik Menggunakan Helm Berbasis Internet Of Things (Iot)," *Jurnal ICTEE*, vol. 1, no. 1, pp. 1–5, 2020.
- [2] H. Isyanto, D. Almada, and H. Fahmiansyah, "Perancangan IoT Deteksi Dini Kebakaran dengan Notifikasi Panggilan Telepon dan Share Location," *Jetri: Jurnal Ilmiah Teknik Elektro*, vol. 18, no. 1, pp. 2–16, Aug. 2020.
- [3] A. Khumaidi, R. Raafi'udin, and I. P. Solihin, "Pengujian Algoritma Long Short Term Memory untuk Prediksi Kualitas Udara dan Suhu Kota Bandung," *Jurnal Telematika*, vol. 15, no. 1, pp. 13–18, 2020.
- [4] Y. A. Susetyo, H. A. Parhusip, S. Trihandaru, and B. Susanto, "LSTM-IOT (LSTM-based IoT) UNTUK MENGATASI KEHILANGAN DATA AKIBAT KEGAGALAN KONEKSI ," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 12, no. 1, pp. 175–186, Feb. 2025.
- [5] T. Lattifia, P. W. Buana, and N. K. D. Rusjyanthi, "Model Prediksi Cuaca Menggunakan Metode LSTM," *Jurnal Ilmiah Teknologi dan Komputer*, vol. 3, pp. 1–6, Apr. 2022.
- [6] M. F. Rizkilloh and S. Widiyanesti, "Prediksi Harga Cryptocurrency Menggunakan Algoritma Long Short Term Memory (LSTM)," *JURNAL RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 1, pp. 25–31, Feb. 2022.
- [7] G. Budiprasetyo, M. Hani'ah, and D. Z. Aflah, "Prediksi Harga Saham Syariah Menggunakan Algoritma Long Short-Term Memory (LSTM)," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 8, pp. 165–172, 2023.
- [8] D. R. Alghifari, M. Edi, and L. Firmansyah, "Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia," *Jurnal Manajemen Informatika (JAMIKA)*, vol. 12, no. 2, pp. 89–99, Oct. 2022.

- [9] M. A. Q. Dani, C. A. Pratama, I. Raihan, and I. Kharisudin, "Pemodelan Runtun Waktu Harga Nikel dengan Algoritma LSTM dan GRU," *PRISMA, Prosiding Seminar Nasional Matematika*, vol. 8, pp. 392–398, 2025.
- [10] A. F. Hanif, T. B. Sasongko, and A. D. Laksito, "Perbandingan Kinerja LSTM, Bi-LSTM, dan GRU pada Klasifikasi Judul Berita Clickbait," *Indonesian Journal of Computer Science*, vol. 12, no. 4, pp. 2136–2150, 2023.
- [11] N. Yulis, M. A. Anhar, and A. S. Rombe, "Analisis Perbandingan Peramalan Penggunaan Bahan Baku Menggunakan Metode Weighted Moving Average (WMA) dan Evaluasi dengan Mean Absolute Error (MAE)," *PROSIDING SEMINAR ILMIAH SISTEM INFORMASI DAN TEKNOLOGI INFORMASI*, vol. 14, no. 1, pp. 57–64, 2025.
- [12] F. K. T. Putri and A. D. Wowor, "Implementasi Algoritma Long Short-Term Memory dalam Prediksi Konsentrasi Gas Metana (CH<sub>4</sub>) di Kota Salatiga," *Jurnal JTik (Jurnal Teknologi Informasi dan Komunikasi)*, vol. 8, no. 2, pp. 419–426, 2024.
- [13] N. O. Idris and F. Pontooyo, "Evaluasi Model Machine Learning untuk Prediksi Harga Mobil dengan Perbandingan Ensemble dan Regresi Linear," *Jurnal Ilmu Komputer dan Sistem Informasi (JIRSI)*, vol. 4, no. 1, pp. 129–143, Jan. 2025.