

LSTM-IOT (LSTM-based IoT) UNTUK MENGATASI KEHILANGAN DATA AKIBAT KEGAGALAN KONEKSI

Yosia Adi Susetyo¹, Hanna Arini Parhusip^{*2}, Suryasatriya Trihandaru³, Bambang Susanto⁴

^{1,2,3,4}Universitas Kristen Satya Wacana, Salatiga
Email: ¹adi.yosia1995@gmail.com, ²hanna.parhusip@uksw.edu,
³suryasatriya@uksw.edu, ⁴bambang.susanto@uksw.edu
*Penulis Korespondensi

(Naskah masuk: 31 Agustus 2024, diterima untuk diterbitkan: 11 Februari 2025)

Abstrak

Masalah dalam industri terkait kehilangan data suhu dan kelembaban sering terjadi akibat gangguan perangkat atau hilangnya koneksi. Data ini penting untuk menentukan kelayakan produk yang akan didistribusikan. Untuk mengatasi permasalahan tersebut, dikembangkan inovasi *LSTM-IOT*, yaitu perangkat *IoT* yang terintegrasi dengan model *Long Short-Term Memory (LSTM)* dalam arsitektur *Environment Intelligence*. Arsitektur ini telah dioptimalkan melalui eksperimen menggunakan berbagai jenis *optimizer*, seperti *Adam*, *RMSprop*, *AdaGrad*, *SGD*, *Nadam*, dan *Adadelta*. Dari hasil optimasi, kombinasi *Nadam Optimizer* dengan arsitektur terpilih menunjukkan kinerja unggul dengan nilai *Mean Square Error (MSE)* sebesar $5,844 \times 10^{-5}$, *Mean Absolute Error (MAE)* sebesar 0,005971, dan *Root Mean Square Error (RMSE)* sebesar 0,007645. Arsitektur *Environment Intelligence* versi (a) dengan *Nadam Optimizer* terbukti paling efektif dalam memproses data sensor, sehingga dipilih untuk integrasi dengan perangkat *LSTM-IOT*. Implementasi *LSTM-IOT* dalam skenario dunia nyata dilakukan pada wadah web lokal yang memungkinkan akses *real-time* ke data suhu dan kelembaban di berbagai lokasi. Halaman web berbasis *Streamlit* ini menampilkan visualisasi data, performa *LSTM*, dan hasil prediksi. Uji fungsional menunjukkan bahwa *LSTM-IOT* memenuhi kebutuhan perusahaan, termasuk penyimpanan data dalam *database* internal serta prediksi kondisi lingkungan hingga 150 menit ke depan. Dengan fitur prediksi dan pemantauan yang canggih, perangkat ini memberikan solusi efisien dan bernilai tinggi bagi perusahaan dalam memantau kondisi lingkungan secara akurat dan proaktif.

Kata kunci: *LSTM*, *internet of thing*, *environment intelligence*, *nadam optimizer*, *data sensor*.

LSTM-IOT (LSTM-BASED IOT) TO ADDRESS DATA LOSS DUE TO CONNECTION FAILURES

Abstract

Problems in the industry related to temperature and humidity data loss are often caused by device interference or loss of connection. This data is important to determine the feasibility of the product to be distributed. To overcome these problems, an *LSTM-IOT* innovation was developed, namely an *IoT* device that is integrated with the *Long Short-Term Memory (LSTM)* model in the *Environment Intelligence* architecture. This architecture has been optimized through experiments using different types of optimizers, such as *Adam*, *RMSprop*, *AdaGrad*, *SGD*, *Nadam*, and *Adadelta*. From the optimization results, the combination of *Nadam Optimizer* with the selected architecture shows superior performance with a mean square error (*MSE*) value of 5.844×10^{-5} , a mean absolute error (*MAE*) of 0.005971, and a root mean square error (*RMSE*) of 0.007645. The *Environment Intelligence* architecture version (a) with *Nadam Optimizer* proved to be the most effective in processing sensor data, so it was chosen for integration with *LSTM-IOT* devices. The implementation of *LSTM-IOT* in real-world scenarios is carried out on a local web container that allows real-time access to temperature and humidity data in various locations. This *Streamlit*-based webpage displays data visualizations, *LSTM* performance, and prediction results. Functional tests show that *LSTM-IOT* meets the needs of the company, including data storage in an internal database and prediction of environmental conditions for up to the next 150 minutes. With advanced prediction and monitoring features, these devices provide efficient and high-value solutions for companies to monitor environmental conditions accurately and proactively.

Keywords: *LSTM*; *internet of thing*; *environment intelligence*; *nadam optimizer*; *sensor data*.

1. PENDAHULUAN

Sistem pantauan suhu dan kelembaban memegang peranan penting di berbagai industri, terutama dalam bidang makanan, minuman, kosmetik, jamu, dan farmasi. Suhu dan kelembaban yang dipantau secara setiap saat memiliki kepentingan dalam mencegah cacat produk, memastikan ketelusuran kondisi lingkungan jika terjadi masalah pada produk, dan memenuhi regulasi yang ditetapkan oleh divisi riset dan pengembangan produk (R&D) (Susilawati, Suseno and Rozikin, 2020). Mengabaikan pantauan lingkungan suhu dan kelembaban dapat menyebabkan berbagai permasalahan serius seperti pertumbuhan bakteri dalam produk, produk yang tidak matang sempurna, kerusakan alat, masa simpan produk yang tidak optimal, dan gangguan pada proses sterilisasi yang memungkinkan bakteri tertentu tetap dapat hidup pada suhu dan kelembaban yang tidak terukur secara tepat (Purnowo, Setiawan and Yusmaniar, 2024). Proses pemantauan suhu dan kelembaban menjadi sangat krusial dalam proses produksi dan dilakukan secara ketat dan berkelanjutan. Selama proses produksi, suhu dan kelembaban harus dipantau apakah sudah sesuai dengan standar SOP yang ditetapkan oleh unit R&D.

Selama ini, pemantauan dilakukan dengan perangkat *IoT* komersial yang memiliki harga beli yang cukup mahal. Perangkat mampu mencatat suhu dan kelembaban setiap saat, dan data disimpan di *cloud* dengan biaya sewa tahunan. Namun, perangkat ini memiliki kelemahan, terutama ketika terjadi kegagalan koneksi atau pemadaman listrik, yang tidak diketahui oleh pengguna menyebabkan data perekaman hilang dan tidak tercatat. Data tersebut sangat penting untuk menentukan apakah produk layak untuk didistribusikan ke konsumen atau tidak. Mengingat pentingnya setiap pencatatan suhu dan kelembaban pada unit produksi maka dalam penelitian ini, akan menawarkan metode pembuatan perangkat *IoT* dengan bahan produksi yang murah, terjangkau dan mudah ditemukan di *marketplace*. Bahan perangkat *IoT* menggunakan *mikrokontroler ESP 32*, sensor *DHT 22*, *LCD I2C*, baterai isi ulang 6000 *mAh*, dan *hard cover* mika. *ESP32* adalah *mikrokontroler* buatan *Espressif Systems* yang populer untuk aplikasi *IoT* memiliki fitur yang canggih dan hemat biaya. *Mikrokontroler* ditenagai oleh prosesor *dual-core Xtensa LX6 12* hingga 240 *MHz*, *ESP32* mendukung *Wi-Fi* dan *Bluetooth*, menjadikannya ideal untuk komunikasi nirkabel. Dengan 520 *KB SRAM* dan beragam *GPIO pin*, *ESP32* mendukung fungsi seperti *PWM*, *ADC*, dan komunikasi serial (Baso, Manjang and Suyuti, 2024; Purwanto et al., 2024). Selanjutnya *DHT22* merupakan sensor digital yang dirancang untuk mengukur suhu dan kelembaban udara dengan akurasi tinggi. Sensor ini sering digunakan dalam berbagai aplikasi pemantauan lingkungan dan proyek *IoT* karena kemampuannya untuk menyediakan data yang stabil dan andal dalam berbagai kondisi. *DHT22*

mampu mengukur suhu dalam rentang -40°C hingga 80°C dengan akurasi $\pm 0,5^{\circ}\text{C}$ dan kelembaban dalam rentang 0% hingga 100% RH dengan akurasi $\pm 2\%$ RH. Data yang dihasilkan oleh sensor ini dikirimkan dalam bentuk sinyal digital, sehingga mudah diintegrasikan dengan mikrokontroler seperti *ESP32* untuk keperluan pemrosesan dan transmisi data lebih lanjut (Muhammad Syukri Sulieman et al., 2024; Rizal et al., 2024). Bahan-bahan tersebut akan dirakit dan diintegrasikan sehingga menghasilkan perangkat *LSTM-IOT* yang mampu mendeteksi suhu dan kelembaban lingkungan.

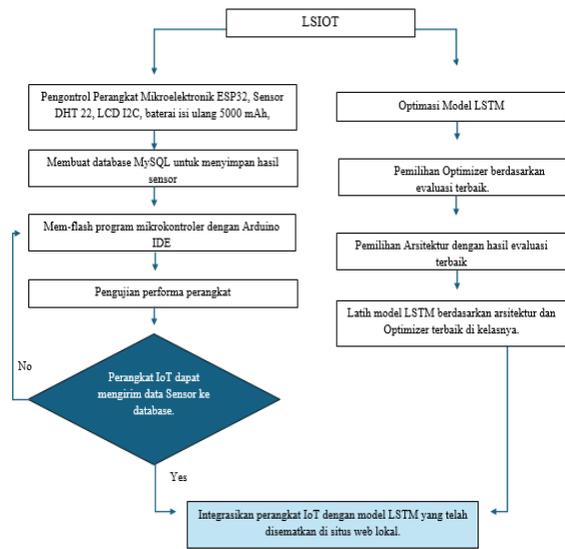
Selain itu, perangkat *IoT* terintegrasi dengan perangkat lunak berbasis situs web lokal menggunakan kerangka kerja *Streamlit*. Di situs web, model pembelajaran mesin *LSTM* dengan arsitektur yang disebut *Ei (Environment Intelligence)* akan disematkan dengan fitur deteksi suhu dan kelembaban. Beberapa penelitian yang telah dilakukan sebelumnya terkait *LSTM* dan perangkat *IoT* disebutkan sebagai berikut : Manzhu melakukan penelitian sebelumnya terkait integrasi *LSTM* dengan perangkat *IoT*, mengintegrasikan *IoT* dengan jaringan *LSTM* untuk memprediksi suhu udara di *New York City*, dan memperoleh nilai *RMSE* dalam kisaran 2,71 hingga 4,02 (Yu et al., 2021). Yalamanchi, melakukan pengumpulan data *IoT* dari *dataset Daphnet* digunakan untuk klasifikasi otomatis penyakit Alzheimer menggunakan pendekatan *deep learning* dengan model *LSTM* memperoleh nilai akurasi 95,68% (Yalamanchi, Venkateshkar and Kumar, 2023). Parhusip berhasil mengintegrasikan perangkat *IoT* dengan model kecerdasan buatan. Algoritma yang digunakan untuk memprediksi data sensor adalah algoritma *K-Means* dan algoritma *Support Vector Regression (SVR)* (Parhusip et al., 2022). Studi lain yang dilakukan oleh Shengwei Li menggunakan sensor *CO2* dan model *LSTM* untuk memperkirakan nilai emisi *CO2* diperoleh dengan rentang nilai 0,986 hingga 0,990 dan *RMSE* dalam kisaran nilai 0,165 hingga 0,167 (Li, Tong and Haroon, 2024). Selanjutnya, Xin Wang memprediksi suhu oli mesin diesel pada lokomotif kereta api menggunakan model metode data *fusion* berdasarkan informasi terkompresi yang dapat membuat prediksi yang sangat akurat dengan nilai *mean square error (MSE)*, *mean absolute error (MAE)*, dan *mean absolute percentage error (MAPE)* sebesar 0,335, 0,369, dan 0,586% (Cheng and Yue, 2024).

Berdasarkan penelitian sebelumnya, belum banyak dilakukan optimasi model *LSTM* berdasarkan arsitektur *neural network* dan *optimizer* yang terintegrasi dengan peperangan *IoT*, bersama dengan implementasi di situs web lokal; hal ini merupakan kebaruan dari penelitian ini. Model *LSTM* akan dibangun dari nilai historis suhu dan kelembaban kurang lebih 400.000 titik data *input* dari sensor yang digunakan sebagai modal untuk membangun model. Selanjutnya dilakukan optimasi pembuatan model *LSTM* dengan berbagai jenis *optimizer* seperti *Adam*

Optimizer, RMSprop Optimizer, AdaGrad Optimizer, SGD Optimizer, Nadam Optimizer, dan Adadelta Optimizer. Dari enam *optimizer*, evaluasi model dilakukan dengan mengukur nilai *MSE (mean squared error)*, *RMSE (root mean squared error)*, dan *MAE (mean absolute error)*. Model *LSTM* yang paling optimal akan dipilih dan diimplementasikan di *website*. Hasil dari proses implementasi ini memungkinkan *website* tidak hanya menampilkan catatan suhu dan kelembaban secara *real-time* tetapi juga dapat memprediksi suhu dan kelembaban untuk beberapa periode ke depan sehingga nilai sensor yang hilang karena perangkat *IoT* rusak atau terputus koneksi dapat diatasi di mana nilai pemantauan suhu dan kelembaban dapat diperoleh melalui hasil prediksi yang dibuat oleh model *LSTM* yang ditampilkan pada situs web. Dengan menggunakan pendekatan ini, sistem pemantauan suhu dan kelembaban akan lebih andal dan ekonomis. Penggunaan mikrokontroler *ESP 32* dan sensor *DHT 22*, serta integrasi dengan model *LSTM*, memastikan data yang akurat dan prediktif, yang penting untuk menjaga kualitas produksi di perusahaan. Selain itu, biaya operasional dapat ditekan karena tidak bergantung pada perangkat *IoT* komersial yang mahal, dan ada biaya penyimpanan data server eksternal berbayar yang berkelanjutan.

2. METODE PENELITIAN

Metode penelitian dilakukan melalui beberapa tahap sebagai berikut: perakitan perangkat *LSTM-IOT*, optimasi model *LSTM* dalam menangani data suhu dan kelembaban, implementasi model dalam situs web lokal menggunakan *framework Streamlit*, evaluasi kinerja sistem berupa performa sistem dalam memprediksi suhu dan kelembaban, serta koneksi situs web lokal dengan perangkat *IoT* secara umum ditunjukkan pada Gambar 1.



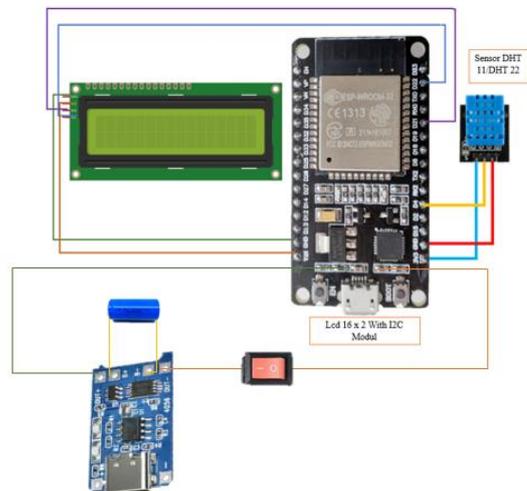
Gambar 1. Alur proses penelitian dalam mengembangkan *LSTM-IOT*

Gambar 1. menguraikan proses yang komprehensif bagaimana mengintegrasikan perangkat *IoT* dengan

model *LSTM* yang dioptimalkan dalam suatu sistem prediktif. Proses meliputi pengaturan perangkat keras dan pemrograman perangkat mikrokontroler, serta pengembangan dan optimalisasi model *LSTM*, sehingga sistem dapat mengumpulkan data sensor, memprosesnya menggunakan model pembelajaran mesin, dan menerapkan hasilnya untuk aplikasi praktis di situs web.

2.1. Perakitan Perangkat IOT

Perakitan perangkat *IoT* suhu dan kelembaban dilakukan dengan menyiapkan alat dan bahan yang diperlukan diantaranya-Nya mikrokontroler *ESP 32*, sensor suhu dan kelembaban *DHT 22*, *LCD 16x2* dengan modul *I2C*, baterai isi ulang *6000 mAh, 3.7V*, saklar dan akrilik (Mohammed Saad Ashraf Alrubaye et al., 2024; Ramakrishnaiah et al., 2024). Perangkat *IoT* dirakit sesuai dengan Gambar 2a. Hasil rakitan dilakukan *flashing* program dengan *Arduino IDE*. Semua perangkat yang telah berhasil terkoneksi dengan Mikrokontroler *ESP32* dilakukan uji fungsi seperti tampilan *LCD*, kinerja sensor suhu dan kelembaban, kinerja baterai dalam melakukan isi ulang dan suplai daya dirakit menjadi suatu produk bernama *LSTM-IOT* seperti Gambar 2a dan Gambar 2b.



Gambar 2a. Skema rangkaian elektronik perangkat *LSTM-IOT*



Gambar 2b. Produk jadi rakitan perangkat *LSTM-IOT*

2.2. Pembuatan data base

Basis data yang digunakan dalam penelitian ini dibangun menggunakan platform *MySQL* dan disusun dalam struktur tabel data. Basis data ini berfungsi sebagai media penyimpanan untuk semua nilai yang diperoleh dari sensor, yang kemudian disimpan secara terorganisir dalam tabel-tabel tersebut (Rawat, Purnama and Mulyati, 2021; Wahyudi et al., 2022). Keberadaan basis data ini sangat penting dalam mendukung pengembangan dan penerapan model *Long Short-Term Memory (LSTM)* yang dirancang. Struktur basis data yang digunakan dapat ditunjukkan pada Gambar 3.

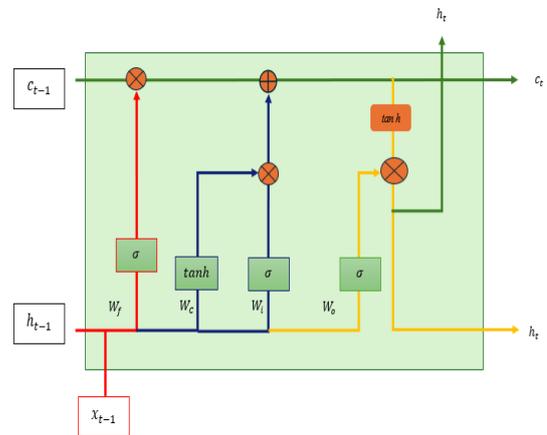
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 401480 entries, 0 to 401479
Data columns (total 5 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   id           401480 non-null  int64
1   suhu        401480 non-null  float64
2   kelembaban  401480 non-null  float64
3   waktu       401480 non-null  object
4   tanggal     401480 non-null  object
dtypes: float64(2), int64(1), object(2)
memory usage: 15.3+ MB
```

Gambar 3. Struktur database pantauan suhu dan kelembaban

2.3. Optimasi model LSTM

2.3.1. Sistem kerja LSTM

LSTM (Long Short-Term Memory) adalah jenis jaringan saraf tiruan (*Artificial Neural Network* atau *ANN*) yang dirancang khusus untuk menangani data berurutan atau *time series*. *LSTM* merupakan varian dari *Recurrent Neural Networks (RNN)* (Taib et al., 2025). *LSTM* diciptakan untuk mengatasi keterbatasan *RNN* dalam mengingat informasi jangka panjang. Dalam *RNN*, ketika menggunakan data yang sangat besar, gradien cenderung mengecil (*vanishing gradient*) atau membesar (*exploding gradient*) selama proses pelatihan. Hal ini menyebabkan *RNN* sulit belajar dari data jangka panjang. *LSTM* adalah model jaringan saraf yang dalam, dan merupakan perluasan dari jaringan saraf berulang (*RNN*) dengan arsitektur yang ditingkatkan. Model ini menggunakan tiga jenis mekanisme gerbang: *input gate*, *forget gate*, dan gerbang *output gate*. Ketiga gerbang ini memungkinkan *LSTM* untuk mengatur aliran informasi secara efektif, sehingga mampu mengingat dan memproses informasi dalam jangka waktu yang lebih panjang (Ali et al., 2024; Cheng and Yue, 2024). Dalam satu lapisan model *LSTM*, mekanisme gerbang bekerja bersama untuk mengelola informasi yang masuk, dilupakan, dan keluar dari sel memori (Dinsa, Das and Abebe, 2024). Sistem kerja dari *LSTM* diilustrasikan pada Gambar 4.



Gambar 4. Struktur sel LSTM

Gambar 4 menunjukkan struktur dari sel *Long Short-Term Memory (LSTM)*. Berdasarkan Gambar 3 diperkenalkan notasi sebagai berikut: x_t : input saat ini; h_{t-1} : keadaan tersembunyi sebelumnya; c_{t-1} : keadaan sel sebelumnya; f_t : output forget gate; i_t : output input gate; \tilde{c}_t : keadaan sel kandidat; c_t : keadaan sel yang diperbarui; o_t : output gate output dan h_t : keadaan tersembunyi yang diperbarui. Mekanisme kerja *LSTM* terdiri dari:

Input Gate (i_t): Gerbang input berfungsi dalam menentukan seberapa banyak informasi baru dari input yang saat ini x_t dan keadaan tersembunyi sebelumnya h_{t-1} yang harus ditambahkan ke dalam sel. Dalam gerbang input menggunakan fungsi aktivasi sigmoid (σ) untuk menghasilkan nilai antara 0 dan 1. Input pada gerbang ini dirumuskan dengan

$$W_i \times [h_{t-1}, x_t] + b_i \quad (1)$$

Forget Gate (f_t): berfungsi untuk menentukan seberapa banyak keadaan sel sebelumnya c_{t-1} yang harus diteruskan. Pada gerbang ini menggunakan aktivasi sigmoid (σ) untuk memutuskan bagian mana dari informasi yang akan disimpan atau dilupakan. Persamaan matematisnya adalah:

$$W_f \times [h_{t-1}, x_t] + b_f \quad (2)$$

Cell State Update (\tilde{c}_t): merupakan kandidat sel baru yang dihitung dengan input saat ini dan keadaan tersembunyi sebelumnya. Dengan fungsi aktivasi yang digunakan adalah *tanh* dengan hasil yang diperoleh adalah nilai pada rentang -1 dan 1 dengan input yang dituliskan dengan:

$$W_c \times [h_{t-1}, x_t] + b_c \quad (3)$$

Output gate (o_t): berfungsi dalam menentukan bagian mana dari sel yang harus dijadikan output sebagai keadaan tersembunyi h_t . Aktivasi yang digunakan adalah *sigmoid* dengan input yang dapat dituliskan dengan :

$$W_o \times [h_{t-1}, x_t] + b_o \quad (4)$$

Selanjutnya akan dijelaskan bagaimana fungsi aktivasi pada sistem kerja dalam sel *LSTM*:

Aktivasi *forget gate* mengacu pada persamaan (2), aktivasi pada *forget gate* yang dapat dituliskan

$$f_t = \sigma(W_f \times [h_{t-1}, x_t] + b_f) \quad (5)$$

aktivasi *forget gate* menentukan bagian mana dari keadaan sel sebelum C_{t-1} yang di lupakan.

Aktivasi *input gate*, mengacu pada persamaan (1) aktivasi pada *input gate* dapat dituliskan :

$$it = \sigma(W_i \times [h_{t-1}, x_t] + b_i) \quad (6)$$

input gate berfungsi untuk mengontrol seberapa banyak informasi baru yang ditambahkan ke keadaan sel.

Perhitungan status sel baru, mengacu pada persamaan (3) maka ditulis kan persamaan dengan aktivasi *tanh* sebagai berikut

$$\tilde{c}_t = \tanh(W_c \times [h_{t-1}, x_t] + b_c) \quad (7)$$

Update Cell State: keadaan sel diperbarui dengan menggabungkan keadaan sel lama yang dimodifikasi oleh gerbang lupa dengan keadaan sel kandidat baru yang diskalakan oleh gerbang *input*. *update cell state* ditulis dengan persamaan:

$$c_t = f_t \times C_{t-1} + i_t \times \tilde{c}_t \quad (8)$$

Aktivasi *output gate*, mengacu pada persamaan (4) maka aktivasi pada *output gate* dapat ditulis

$$o_t = \sigma(W_o \times [h_{t-1}, x_t] + b_o) \quad (9)$$

pada *output gate* memiliki peranan dalam menentukan apa yang harus dikeluarkan berdasarkan keadaan sel yang diperbaharui.

Hidden state update, keadaan tersembunyi dihitung ketika menggunakan gerbang keluaran dan keadaan sel yang diperbarui. Selanjutnya dituliskan dengan persamaan

$$h_t = o_t \tanh(c_t) \quad (10)$$

keadaan tersembunyi ini merupakan representasi dari informasi yang diproses oleh *LSTM* dan akan diteruskan ke langkah waktu berikutnya, serta digunakan untuk menghasilkan *output* pada langkah saat ini.

2.3.2. Arsitektur LSTM

LSTM memiliki mekanisme yang mampu menyimpan, memperbarui, dan memproses informasi secara efisien dalam rangkaian data yang panjang. Sel *LSTM* memungkinkan jaringan untuk belajar dan mempertahankan ketergantungan jangka panjang dengan mengatur aliran informasi melalui setiap gerbang dengan hati-hati. Dalam pembelajaran mendalam *LSTM* di organisasikan dalam bentuk lapisan yang terdiri dari beberapa sel *LSTM* yang saling bekerja secara paralel. Jumlah *LSTM* dalam suatu lapisan bergantung oleh ukuran dari *hidden state* atau *hidden units*. Model *deep learning* yang dibangun terdiri dari beberapa *layer LSTM* hal ini bergantung pada kompleksitas pada masalah yang dipecahkan, Setiap *layer* akan memproses *output* dari *layer* sebelumnya sehingga memungkinkan model untuk menangkap fitur yang semakin kompleks dari

data *time series* (Kholifatullah and Prihanto, 2023). Dalam model *deep learning* terdapat jenis dari susunan *LSTM* yaitu *single layer* dan *multi layer*. *Single layer* merupakan struktur *LSTM* yang paling sederhana dengan sejumlah sel yang sama dengan ukuran *hidden state* dan *Multi-layer LSTM (Stacked LSTM)* merupakan *LSTM* yang tersusun dari beberapa lapisan ditumpuk satu di atas yang lain (Salman et al., 2018). *Output* dari satu *layer LSTM* menjadi *input* bagi *layer* berikutnya yang memungkinkan model untuk mempelajari fitur yang lebih abstrak dan kompleks. Dalam penelitian ini untuk membangun model yang mampu memprediksi suhu dan kelembaban digunakan arsitektur *deep learning* yang diberi nama *Ei (Environment Intelligence)*. Terdapat dua versi struktur *Environment Intelligence* yang digunakan *Ei* versi (a) yang ditunjukkan pada Tabel 1 dan *Ei* versi (b) yang ditunjukkan pada tabel 2.

Tabel 1. Arsitektur *LSTM Environment intelligence* versi a.

Layer (type)	Output shape	Param #	Activation
<i>Lstm_20 (LSTM)</i>	(None, 50, 128)	67,072	<i>ReLU</i>
<i>Lstm_21 (LSTM)</i>	(None, 50, 64)	49,408	<i>ReLU</i>
<i>Lstm_22 (LSTM)</i>	(None, 50, 32)	12,416	<i>ReLU</i>
<i>Lstm_23 (LSTM)</i>	(None, 50,16)	3, 136	<i>tanh</i>
<i>Lstm_24 (LSTM)</i>	(None, 8)	800	<i>tanh</i>
<i>dense_4 (Dense)</i>	(None,2)	18	

Tabel 2. Arsitektur *LSTM Environment intelligence* versi b.

Layer (type)	Output shape	Param #	Activation
<i>lstm_10 (LSTM)</i>	(None, 50, 128)	67072	<i>ReLU</i>
<i>dropout_4 (Dropout)</i>	(None, 50, 128)	0	<i>ReLU</i>
<i>batch_normalization_4 (BatchNormalization)</i>	(None, 50, 128)	512	<i>ReLU</i>
<i>lstm_11 (LSTM)</i>	(None, 50, 64)	49408	<i>ReLU</i>
<i>dropout_5 (Dropout)</i>	(None, 50, 64)	0	<i>ReLU</i>
<i>batch_normalization_5 (BatchNormalization)</i>	(None, 50, 64)	256	<i>ReLU</i>
<i>lstm_12 (LSTM)</i>	(None, 50, 32)	12416	<i>ReLU</i>
<i>dropout_6 (Dropout)</i>	(None, 50, 32)	0	<i>ReLU</i>
<i>batch_normalization_6 (BatchNormalization)</i>	(None, 50, 32)	128	<i>ReLU</i>
<i>lstm_13 (LSTM)</i>	(None, 50, 16)	3136	<i>tanh</i>
<i>dropout_7 (Dropout)</i>	(None, 50, 16)	0	<i>tanh</i>
<i>batch_normalization_7 (BatchNormalization)</i>	(None, 50, 16)	64	<i>tanh</i>
<i>lstm_14 (LSTM)</i>	(None, 50, 8)	800	<i>tanh</i>
<i>dense_2 (Dense)</i>	(None, 2)	18	

Tabel 1 merupakan model *LSTM Environment intelligence* versi (a) yang terdiri dari beberapa lapisan bertingkat, dimulai dengan *LSTM_20* memiliki 128 unit sel dan menghasilkan urutan dengan panjang 50 langkah waktu serta vektor *output* berdimensi 128. Lapisan berikutnya, *LSTM_21*, mengurangi dimensi *output* menjadi 64, diikuti oleh *LSTM_22* dengan 32 unit sel, dan *LSTM_23* dengan 16 unit sel, yang secara bertahap mengurangi dimensi *output* tiap langkah waktu. Pada *LSTM_24*, keluaran

terakhir tidak lagi berupa urutan waktu dan hanya terdiri dari 8 dimensi. Akhirnya, lapisan *Dense* terakhir mengubah vektor 8 dimensi menjadi 2 dimensi, yang digunakan untuk regresi akhir dalam model. Tabel 2 menampilkan model *LSTM Environment intelligence* versi (b) dengan lima lapisan *Long Short-Term Memory (LSTM)* bertingkat yang secara bertahap mengurangi ukuran representasi *sekuensial*, sehingga mempermudah pemetaan dari *input* ke *output*. Lapisan pertama memiliki 128 unit, kemudian jumlah unit ini berkurang hingga hanya 8 unit di lapisan terakhir.

Pada *Environment intelligence* versi (b) disetiap lapisan *LSTM* disertai dengan lapisan *dropout* dan *batch normalization* untuk mengurangi *overfitting* dan meningkatkan stabilitas selama pelatihan (Verianto, 2024). Pada lapisan *LSTM* terakhir, fungsi aktivasi *tanh* digunakan untuk menjaga *output* dalam rentang -1 hingga 1, yang membantu stabilisasi dan interpretasi data. *Output* dari lapisan terakhir ini diteruskan ke lapisan *Dense* yang memiliki dua unit *output*, yang digunakan untuk regresi dua variabel. Dengan struktur bertingkat model mampu menangkap pola *sekuensial* yang kompleks dalam data sambil tetap mempertahankan kemampuan generalisasi yang baik berkat penggunaan *dropout* dan *batch normalization*.

Dalam membangun Arsitektur *LSTM* kami menggunakan *Activation ReLU* pada lapisan awal dan *tanh* pada lapisan akhir struktur kombinasi *activation* dinilai seimbang dalam penanganan kompleksitas dan stabilitas data. *ReLU* akan bekerja dengan baik pada lapisan awal karena memungkinkan model untuk mempelajari fitur-fitur penting dengan cepat dan mencegah *vanishing gradient*, sehingga pelatihan menjadi lebih stabil dan cepat. Selanjutnya *activation tanh* akan menjaga *output* tetap terkontrol dan stabil, yang berguna untuk memastikan prediksi akhir lebih akurat dan konsisten. Kombinasi ini memungkinkan model mendapatkan manfaat terbaik dari kedua fungsi aktivasi, yaitu konvergensi yang cepat dan stabilitas *output*. Hal ini sejalan pada penelitian sebelumnya yang melakukan peramalan data cuaca ekstrem di Indonesia, menggunakan beberapa *activation* dalam membangun model seperti *tanh*, *ReLU*, *Sigmoid* dan *Gaussian* memperoleh bahwa ditemukan bahwa *activation* terbaik menggunakan *activation ReLU* ditinjau dari nilai evaluasi *RMSE* dan *RME* (Hikmah et al., 2023)

Dataset bahan pembuatan model *LSTM* diambil dari histori rekaman perangkat *IoT* dengan fitur suhu dan kelembaban. Jumlah *time step* adalah 50, menunjukkan bahwa data urutan dari data ke-1 hingga ke-50 merupakan data *independen* (x) dan data ke-51 merupakan data *dependen* (y), berlanjut seterusnya. Data ke-2 hingga ke-51 bertindak sebagai variabel x dan data ke-52 sebagai variabel y. Data

selanjutnya dibagi menjadi data tes dan data pelatihan.

Data pelatihan digunakan untuk melatih model *LSTM* dengan arsitektur *Environment Intelligence (Ei)* versi (a) dan *Environment Intelligence (Ei)* versi (b) seperti yang ditunjukkan pada Tabel 1 dan Tabel 2. Data tes digunakan untuk evaluasi model, dengan nilai evaluasi model diukur menggunakan *MSE (Mean Squared Error)*, *RMSE (Root Mean Squared Error)*, dan *MAE (Mean Absolute Error)* (Teh, Kempa-Liehr and Wang, 2020).

2.3.3. Optimasi Model

Dalam proses *deep learning* pada *LSTM*, *optimizer* memiliki peran penting untuk memperbarui bobot dan bias. *Optimizer* berfungsi untuk mengatur seberapa banyak bobot dan bias jaringan saraf harus disesuaikan berdasarkan gradien dan fungsi *loss*, sehingga dapat meningkatkan akurasi prediksi *LSTM*. Selain itu, *optimizer* juga membantu mengurangi *overfitting*, yaitu situasi di mana model sangat cocok dengan data pelatihan namun buruk dalam memprediksi data baru. *Optimizer* mempercepat konvergensi dengan menggunakan adaptasi *learning rate*, sehingga model dapat lebih cepat mencapai solusi yang optimal. Beberapa *optimizer* terkenal karena kemampuannya dalam menyesuaikan *learning rate* selama pelatihan, yang membuat proses konvergensi lebih stabil dan cepat. Penggunaan *optimizer* yang tepat sangat penting untuk memastikan model *LSTM* tidak hanya belajar dengan cepat, tetapi juga mampu melakukan generalisasi dengan baik terhadap data baru. Terdapat beberapa jenis *optimizer* yang umum digunakan, di antaranya adalah *Adam Optimizer*, *RMSprop Optimizer*, *AdaGrad Optimizer*, *SGD Optimizer*, *Nadam Optimizer*, dan *Adadelata Optimizer* (Dharrao et al., 2024; Hu et al., 2024; Kumar et al., 2024). Proses optimasi dilakukan selama pelatihan model *LSTM* dengan menggunakan beberapa jenis *optimizer* tersebut selanjutnya dilakukan evaluasi dengan mengukur *MSE (Mean Squared Error)*, *RMSE (Root Mean Squared Error)*, dan *MAE (Mean Absolute Error)*. Hasil optimasi dari *optimizer* dengan versi arsitektur terbaik kemudian dipilih untuk membangun model *LSTM* yang diimplementasikan ke dalam *website* pantauan suhu dan kelembaban dilingkungan perusahaan.

2.4. Pembuatan Website yang Diperkaya Dengan Fitur Prediksi Dengan Model LSTM

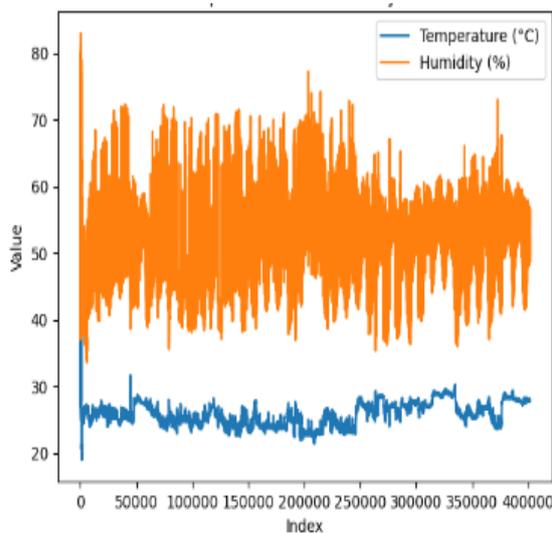
Berdasarkan nilai evaluasi model berupa *MSE (Mean Squared Error)*, *RMSE (Root Mean Squared Error)*, dan *MAE (Mean Absolute Error)*, dipilih arsitektur dan *optimizer* terbaik untuk ditanamkan ke dalam lokal *website* yang dibangun dengan *frame work streamlit* (Nápoles-Duarte et al., 2022; Prasad, Gupta and Ghosh, 2024). *Streamlit* merupakan *frame work* yang *open source* yang digunakan untuk membangun *web* yang interaktif secara mudah dan cepat (Cui et al., 2024; Sufian et al., 2024). Sistem akan mengambil

data rekaman *website* dari *database MySql* diambil 50.000 data terakhir dari rekaman perangkat *LSTM-IOT* yang tersimpan dalam *database MySql* selanjutnya data digunakan untuk memprediksi nilai suhu dan kelembaban sebanyak 150 *input* sensor ke depan. Sistem ini berjalan secara *realtime* sehingga nilai prediksi selalu diperbaharui seiring bertambahnya data baru yang dikirim sensor ke *database MySql*. Hasil prediksi dari model dapat disimpan sehingga pengguna dapat memperoleh data *backup* yang berupa informasi data prediksi. Data prediksi berguna sebagai antisipasi ketika perangkat *IOT* mengalami kegagalan koneksi atau kerusakan yang tidak diketahui oleh pengguna. Pengujian dilakukan dengan melihat sejauh mana data prediksi dapat digunakan. *Website* juga memiliki beberapa fitur berupa tampilan suhu dan kelembaban secara *real-time*, tampilan suhu dan kelembaban di beberapa lokasi pantauan yang dilakukan secara simultan.

3. HASIL DAN DISKUSI

3.1. Optimasi Model

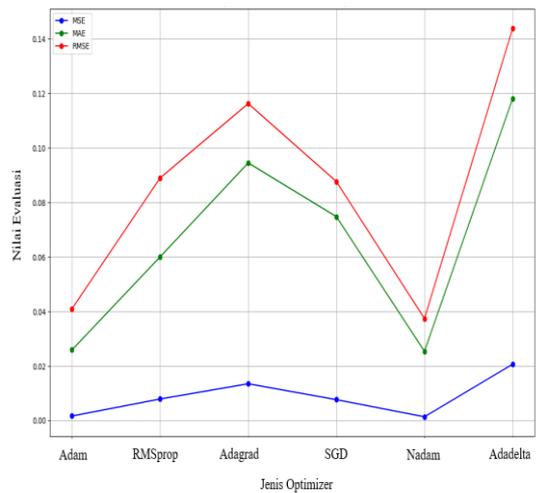
Hasil pengukuran suhu dan kelembaban dengan *LSTM-IOT* ditampilkan pada Gambar 5.



Gambar 5. Plot suhu dan kelembaban

Gambar 5 menunjukkan nilai sensor dari suhu dan kelembaban yang direkam kurang lebih sebanyak 400.000 data pada lingkungan kerja. Dari data tersebut dibangun model *LSTM* dengan berbagai jenis *optimizer* seperti yang telah dijelaskan sebelumnya. Diperoleh hasil evaluasi *optimizer* yang ditunjukkan pada Gambar 6.

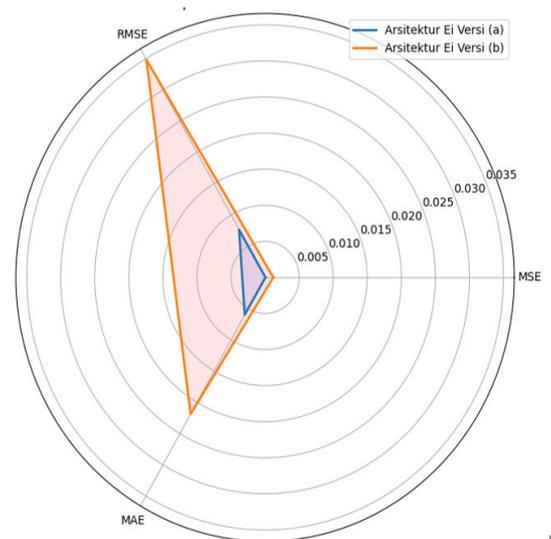
Gambar 6 Menunjukkan grafik yang membandingkan kinerja berbagai *optimizer* dengan menggunakan tiga metrik: *Mean Squared Error (MSE)*, *Mean Absolute Error (MAE)*, dan *Root Mean Squared Error (RMSE)*. Dari grafik tersebut, diketahui bahwa *optimizer* *Nadam* memberikan kinerja terbaik di antara semua *optimizer*, dengan nilai *MSE*, *MAE*, dan *RMSE* terendah.



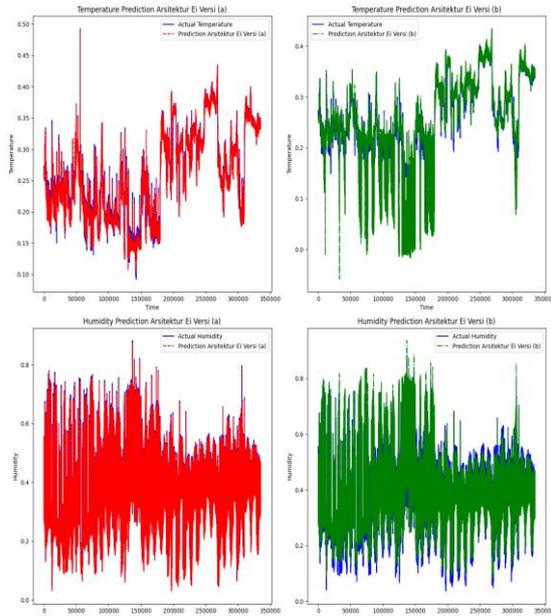
Gambar 6. Hasil evaluasi *MSE*, *MAE*, dan *RMSE* pada setiap jenis *Optimizer*

Sebaliknya, *optimizer* *AdaGrad* dan *Adadelta* menunjukkan nilai *error* yang lebih tinggi, mengindikasikan kinerja yang lebih buruk dibandingkan dengan *optimizer* lainnya. *Optimizer* *Adam* dan *SGD* memiliki nilai *error* yang sedang, menjadikannya pilihan alternatif, tergantung pada kebutuhan spesifik dan *trade-off* yang diinginkan. Grafik berguna dalam pemilihan *optimizer* yang paling sesuai untuk model *machine learning* berdasarkan metrik *error* yang dihasilkan. Dari hasil evaluasi dipilih *optimizer* *Nadam* yang selanjutnya diterapkan pada model arsitektur *LSTM Environment Intelligence (Ei)* versi (a) dan *Environment Intelligence (Ei)* versi (b) diperoleh nilai evaluasi pada model arsitektur pada Gambar 7. Model arsitektur digunakan untuk memprediksi data sensor diperoleh perbandingan hasil prediksi di antara dua arsitektur yang ditunjukkan pada Gambar 7.

Perbandingan Matriks Evaluasi Antara Kedua Arsitektur LSTM



Gambar 7. Diagram laba-laba nilai evaluasi dari setiap arsitektur



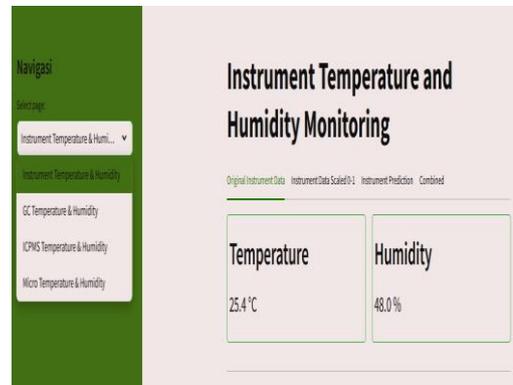
Gambar 8. Perbandingan kinerja model LSTM arsitektur *Environment Intelligence (Ei)* versi (a) dan *Environment Intelligence (Ei)* versi (b) dalam melakukan prediksi data suhu dan kelembaban

Gambar 7 menunjukkan perbandingan kinerja dua arsitektur, yaitu "*Arsitektur Ei Versi (a)*" dan "*Arsitektur Ei Versi (b)*" berdasarkan tiga metrik evaluasi: *MSE (Mean Squared Error)*, *RMSE (Root Mean Squared Error)*, dan *MAE (Mean Absolute Error)*. Dari grafik tersebut, terlihat bahwa "*Arsitektur Ei Versi (a)*" memiliki performa yang lebih baik dibandingkan "*Arsitektur Ei Versi (b)*" karena nilai *MSE*, *RMSE*, dan *MAE* lebih kecil, yang ditunjukkan oleh area biru yang lebih dekat ke pusat grafik. Sebaliknya, "*Arsitektur Ei Versi (b)*" memiliki kesalahan yang lebih tinggi, terutama pada metrik *RMSE* dan *MAE*, yang ditunjukkan oleh area oranye yang lebih besar dan lebih jauh dari pusat. Secara keseluruhan, "*Arsitektur Ei Versi (a)*" lebih akurat dan andal dibandingkan "*Arsitektur Ei Versi (b)*". Perbandingan nilai evaluasi *Arsitektur Ei Versi (a)* dan (b) dinyatakan sebagai berikut : *MSE* *Arsitektur Ei Versi (a)* sebesar 5.844×10^{-5} dan *Arsitektur Ei Versi (b)* sebesar 0.0012141; Nilai *MAE* *Arsitektur Ei Versi (a)* sebesar 0.005971 dan *Arsitektur Ei Versi (b)* sebesar 0.02187; *RMSE* *Arsitektur Ei Versi (a)* sebesar 0.007645 dan *Arsitektur Ei Versi (b)* sebesar 0.03484. Gambar 8 menunjukkan bahwa *Arsitektur Ei Versi (a)* menunjukkan kinerja yang lebih baik dibandingkan dengan *Arsitektur Ei Versi (b)* dalam hal kecocokan prediksi dengan data asli. Grafik hasil prediksi suhu dan kelembaban pada *Arsitektur Ei Versi (a)* memperlihatkan pola yang lebih mendekati dan mengikuti data aktual, dengan deviasi yang lebih kecil dibandingkan dengan *Arsitektur Ei Versi (b)*. Sebaliknya, *Arsitektur Ei Versi (b)* cenderung menghasilkan prediksi yang lebih fluktuatif dan kurang stabil, yang menyebabkan perbedaan yang lebih mencolok antara prediksi dan data sebenarnya. Hal ini menunjukkan bahwa *Arsitektur Ei Versi (a)*

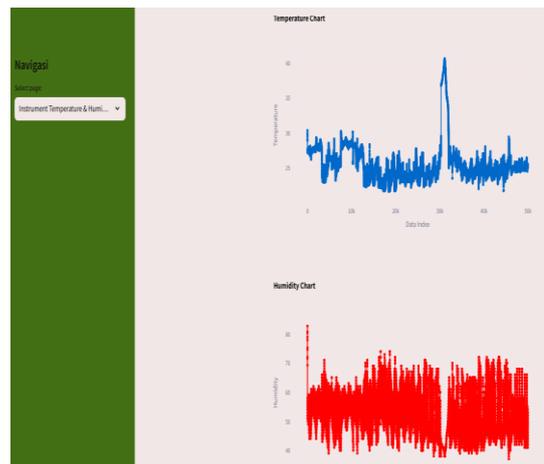
memiliki kemampuan generalisasi yang lebih baik terhadap data lingkungan, sehingga lebih tepat untuk digunakan dalam model prediksi suhu dan kelembaban dalam konteks ini. Kesimpulan ini mengindikasikan bahwa parameter dan struktur yang digunakan dalam *Arsitektur Ei Versi (a)* lebih efektif dalam menangkap pola dari data sensor yang ada.

3.2. Implementasi Model pada lokal *website*

Model *LSTM* dan perangkat *IoT* berhasil untuk diintegrasikan, perangkat *IoT* akan membaca parameter suhu dan kelembaban sesuai dengan jenis sensor yang dipasang. Selanjutnya data sensor akan dikirim ke dalam *database*. *Website* akan mengambil 5000 data terakhir dan digunakan untuk memprediksi nilai sensor pada 150 langkah ke depan dalam satuan menit. Hasil lokal *website* yang dibangun dengan *frame work streamlit* sebagaimana ditunjukkan pada Gambar (9) hingga Gambar (14).



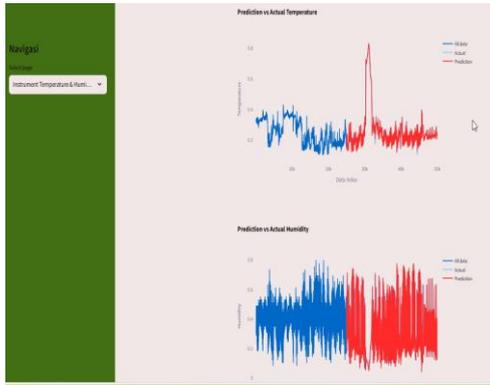
Gambar 9. Tampilan sensor LSTM-IOT secara realtime



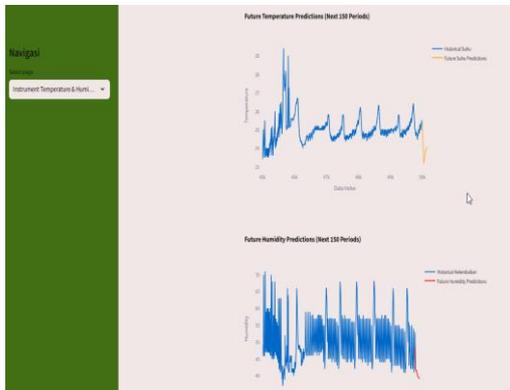
Gambar 10. Tampilan histori pencatatan suhu dan kelembaban

Gambar 9-14 menunjukkan data rekaman Perangkat *LSTM-IOT* berhasil diintegrasikan dengan lokal *website*. *Website* mampu menampilkan secara visual data suhu kelembaban terkini, data historis suhu kelembaban pada periode sebelumnya, data prediksi nilai suhu dan kelembaban yang merupakan *output* dari model *LSTM* yang ditanam dalam lokal *web*. Uji

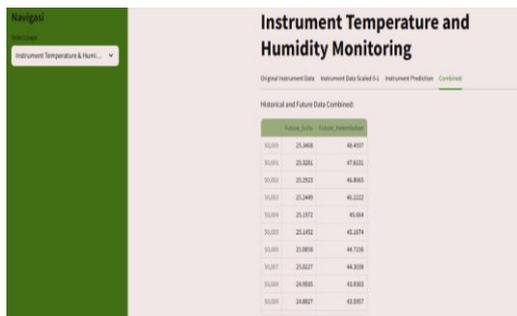
lanjutan dilakukan dengan uji fungsi perangkat yang disesuaikan dengan kondisi di lapangan.



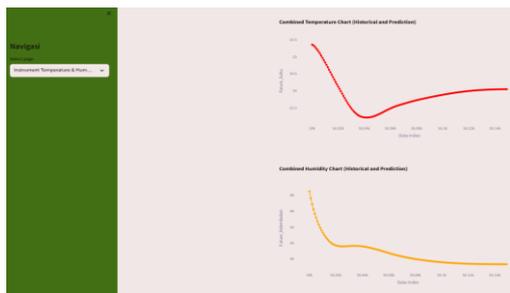
Gambar 11. Tampilan kemampuan model dalam memprediksi data historis



Gambar 12. Tampilan kemampuan model dalam memprediksi 150 langkah ke depan



Gambar 13. Tampilan data set hasil prediksi dalam 150 langkah ke depan



Gambar 14. Pola 150 data prediksi suhu dan kelembaban

Uji fungsi bertujuan untuk menilai apakah perangkat *LSTM-IOT* mampu menjawab kebutuhan mendasar pengguna (Gantini, Djajalaksana and Yefta, 2018). Hasil pengujian dapat ditunjukkan pada Tabel 3.

Tabel 3. Tes fungsi perangkat LSTM-IOT

No	Parameter Pengujian	Perangkat IoT Komersial	Perangkat LSTM-IOT
1	Perangkat <i>IoT</i> Dapat mendeteksi suhu dan kelembaban lingkungan	Ya	Ya
2	Perangkat <i>IoT</i> dapat mengirim data ke server internal <i>MySQL</i>	Hanya bisa ke server eksternal dengan penyimpanan berbayar	Ya
3	<i>Website</i> mampu melakukan prediksi nilai sensor suhu dan kelembaban di 150 menit ke depan	Tidak	Ya
4	Sistem mampu menangani kegagalan koneksi jaringan dan melanjutkan pengiriman data setelah terhubung kembali:	Ya	Ya
5	Perangkat <i>IoT</i> mampu bertahan dengan daya baterai selama minimal 24 jam:	Ya	Ya
6	Pengguna dapat mengakses dan mengunduh riwayat data sensor dalam format <i>csv</i> dan <i>excel</i> :	Ya	Ya

Perbandingan antara perangkat *IoT* komersial dan perangkat *LSTM-IOT* menunjukkan beberapa keunggulan signifikan pada perangkat *LSTM-IOT*. Kedua perangkat sama-sama mampu mendeteksi suhu dan kelembaban lingkungan, serta bertahan dengan daya baterai minimal 24 jam. Selain itu, keduanya mampu menangani kegagalan koneksi jaringan dengan melanjutkan pengiriman data setelah koneksi pulih, dan memungkinkan pengguna untuk mengakses serta mengunduh riwayat data sensor dalam format *CSV* dan *Excel*. Namun, perangkat *LSTM-IOT* unggul dalam beberapa aspek penting. Pertama, *LSTM-IOT* dapat mengirim data langsung ke server internal berbasis *MySQL*, sementara perangkat *IoT* komersial hanya dapat mengirim data ke server eksternal dengan penyimpanan berbayar, yang membatasi fleksibilitas dan meningkatkan biaya operasional. Kedua, perangkat *LSTM-IOT* juga memiliki kemampuan prediksi nilai sensor untuk 150 pembacaan ke depan, fitur yang tidak dimiliki oleh perangkat *IoT* komersial. Keunggulan-keunggulan ini menjadikan perangkat *LSTM-IOT* lebih unggul dalam

pengelolaan data yang lebih mendalam, analisis prediktif, dan penghematan biaya operasional.

4. KESIMPULAN

Penelitian ini berisi solusi untuk masalah kehilangan data pada sensor suhu dan kelembaban akibat kerusakan perangkat atau kehilangan koneksi pada perangkat. Padahal, data yang hilang sangat penting untuk menentukan apakah hasil produksi dapat didistribusikan atau tidak. Untuk mengatasi masalah tersebut, serangkaian sistem yang disebut *LSTM-IOT* merupakan integrasi dari sistem *IoT* dan model LSTM yang terintegrasi dengan web, di mana model LSTM telah disematkan di web untuk dapat memprediksi suhu dan kelembaban untuk beberapa langkah selanjutnya. *LSTM* yang ditanamkan adalah model yang paling optimal dalam hal evaluasi yang diperoleh dari struktur jaringan *deep learning* dan pengoptimalan yang dipilih. Perangkat *LSTM-IOT* yang berhasil dibangun menunjukkan keunggulan dalam berbagai aspek, termasuk efisiensi biaya, fleksibilitas dalam manajemen data, dan kemampuan prediktif yang unggul. Perangkat ini terintegrasi ke dalam *website* yang dilengkapi dengan model *LSTM* yang telah dioptimalkan menggunakan beberapa arsitektur, yaitu *Environment Intelligence* versi (a) dan *Environment Intelligence* versi (b), serta berbagai *optimizer* seperti *Adam*, *RMSprop*, *AdaGrad*, *SGD*, *Nadam*, dan *Adadelta*. Hasil optimal dicapai melalui kombinasi *Nadam Optimizer* dengan arsitektur versi *Environment Intelligence* (a), yang memberikan kinerja prediksi yang sangat baik. Hal ini dibuktikan dengan nilai evaluasi: kesalahan rata-rata kuadrat (*MSE*) $5,844 \times 10^{-5}$, kesalahan absolut rata-rata (*MAE*) 0,005971, dan kesalahan kuadrat rata-rata akar (*RMSE*) lokal 0.007645. Situs *web* yang terintegrasi dengan perangkat *LSTM-IOT* ini dapat menampilkan suhu dan kelembaban secara *real-time*, memberikan grafik data historis, dan menampilkan hasil prediksi suhu dan kelembaban dalam bentuk kumpulan data dan grafik. Pengujian lebih lanjut mengungkapkan bahwa perangkat *LSTM-IOT* tidak hanya dapat menggantikan perangkat *IoT* komersial yang ada tetapi juga menawarkan kemampuan unggul dengan biaya yang lebih hemat biaya, menjadikannya solusi yang lebih efektif dan ekonomis untuk implementasi *IoT* di berbagai lingkungan.

5. UCAPAN TERIMA KASIH

Penelitian ini didanai oleh Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi dengan nomor hibah 001/LL6/PB/AL.04/2024,027/SPK-PFR/RIK/6/2024. Penulis menyatakan bahwa tidak ada potensi konflik kepentingan terkait dengan publikasi karya ini. Selain itu, semua aspek etika, termasuk plagiarisme, persetujuan yang diinformasikan, pelanggaran, fabrikasi data, pemalsuan, dan prosedur publikasi, telah ditangani di bawah standar etika yang berlaku.

6. DAFTAR PUSTAKA

- ALI, M., SARWAR, T., MUBARAK, N.M., KARRI, R.R., GHALIB, L., BIBI, A. AND MAZARI, S.A., 2024. Prediction of CO₂ solubility in Ionic liquids for CO₂ capture using deep learning models. *Scientific Reports*, [online] 14(1), pp.1–19. <https://doi.org/10.1038/s41598-024-65499-y>.
- BASO, M., MANJANG, S. AND SUYUTI, A., 2024. The Intelligent kWh Export-Import Utilizing Classification Models for Efficiency in Hybrid PLTS. *Journal of Applied Data Sciences*, 5(2), pp.668–678. <https://doi.org/10.47738/jads.v5i2.244>.
- CHENG, C. AND YUE, W., 2024. Construction of a Green and Low-Carbon Travel Order Prediction Model Based on Shared Bicycle Big Data. *International Journal of Computational Intelligence Systems*, [online] 17(1). <https://doi.org/10.1007/s44196-024-00519-3>.
- CUI, Y., SHI, X., QIN, Y., WANG, Q., CAO, X., CHE, X., PAN, Y., WANG, B., LEI, M. AND LIU, Y., 2024. Establishment and validation of an interactive artificial intelligence platform to predict postoperative ambulatory status for patients with metastatic spinal disease: a multicenter analysis. *International journal of surgery (London, England)*, 110(5), pp.2738–2756. <https://doi.org/10.1097/JS9.0000000000001169>.
- DHARRAO, D., MR, A., MITAL, R., VENGALI, A., PANGAVHANE, M., RAJPUT, S. AND BONGALE, A.M., 2024. An efficient method for disaster tweets classification using gradient-based optimized convolutional neural networks with BERT embeddings. *MethodsX*, [online] 13(July), p.102843. <https://doi.org/10.1016/j.mex.2024.102843>.
- DINSA, E.F., DAS, M. & ABEBE, T.U., 2024. AI-based disease category prediction model using symptoms from low-resource Ethiopian language: Afaan Oromo text. *Scientific Reports*, [online] 14(1), pp.1–15. <https://doi.org/10.1038/s41598-024-62278-7>.
- GANTINI, T., DJAJALAKSANA, Y.M. & YEFTA, S.K., 2018. Pengujian Perangkat Lunak. *Journal Universitas STEKOM*, [online] 4, pp.1–4. Available at: <<https://journal.stekom.ac.id/>>.
- HIKMAH, H., ASRIRAWAN, A., APRIYANTO, A. & NILAWATI, N., 2023. Peramalan Data Cuaca Ekstrim Indonesia Menggunakan Model ARIMA dan Recurrent Neural Network. *Jambura Journal of Mathematics*,

- 5(1).
<https://doi.org/10.34312/jjom.v5i1.17496>.
- HU, H., LU, H., SHI, R., FAN, X. & DENG, Z., 2024. A novel fault diagnosis method for key transmission sections based on Nadam-optimized GRU neural network. *Electric Power Systems Research*, [online] 233(May), p.110522. <https://doi.org/10.1016/j.epsr.2024.110522>.
- KHOLIFATULLAH, B.A.H. & PRIHANTO, A., 2023. Penerapan Metode Long Short Term Memory Untuk Klasifikasi Pada Hate Speech. *Journal of Informatics and Computer Science (JINACS)*, 04, pp.292–297. <https://doi.org/10.26740/jinacs.v4n03.p292-297>.
- KUMAR, Y., GARG, P., MOUDGIL, M.R., SINGH, R., WOŹNIAK, M., SHAFI, J. & IJAZ, M.F., 2024. Enhancing parasitic organism detection in microscopy images through deep learning and fine-tuned optimizer. *Scientific Reports*, [online] 14(1), pp.1–29. <https://doi.org/10.1038/s41598-024-56323-8>.
- LI, S., TONG, Z. & HAROON, M., 2024. Estimation of transport CO2 emissions using machine learning algorithm. *Transportation Research Part D*, [online] 133(June), p.104276. <https://doi.org/10.1016/j.trd.2024.104276>.
- MOHAMMED SAAD ASHRAF ALRUBAYE, AZIZAN AS'ARRY, MUHAMMED AMIN AZMAN, MOHD ZUHRI MOHAMED YUSOFF, KHAIRIL ANAS MD REZALI & ALI ZOLFAGHARIAN, 2024. Monitoring and Prediction of Air Quality System using Internet of Things (IoT). *Journal of Advanced Research in Applied Sciences and Engineering Technology*, 48(1), pp.61–76. <https://doi.org/10.37934/arasets.48.1.6176>.
- MUHAMMAD SYUKRI SULIAMAN, ADAM SAMSUDIN, MUHAMAD FALIHAN BAHARI, NOR HAFIZAH HUSSIN, NUR AZURA NOOR AZHUAN, NURUL HAJAR MOHD YUSSOFF, NORHAYATI ROSLI & IRIANTO IRIANTO, 2024. Development of Automated Exhaust Fan for Modern Kitchen with IoT Notification System. *Journal of Advanced Research in Applied Mechanics*, 120(1), pp.99–109. <https://doi.org/10.37934/aram.120.1.99109>.
- NÁPOLES-DUARTE, J.M., BISWAS, A., PARKER, M.I., PALOMARES-BAEZ, J.P., CHÁVEZ-ROJO, M.A. & RODRÍGUEZ-VALDEZ, L.M., 2022. Stmol: A component for building interactive molecular visualizations within streamlit web-applications. *Frontiers in Molecular Biosciences*, 9(September), pp.1–10. <https://doi.org/10.3389/fmolb.2022.990846>.
- PARHUSIP, H.A., TRIHANDARU, S., RUMAKSARI, A.N., PUSPITASARI, M.D., HARYADI, A.H. & SANTOSA, P.P., 2022. Integrated Sensors into Artificial Intelligence Mining (AI-Mining) Data Acquisition of Environmental Features. *International Interdisciplinary Humanitarian Conference for Sustainability, IIHC 2022 - Proceedings*, pp.532–539. <https://doi.org/10.1109/IIHC55949.2022.10060158>.
- PRASAD, S., GUPTA, H. & GHOSH, A., 2024. Leveraging the Potential of Large Language Models. *Informatica (Slovenia)*, 48(8), pp.1–16. <https://doi.org/10.31449/inf.v48i8.5635>.
- PURNOWO, D., SETIAWAN, A. & YUSMANIAR, Y., 2024. Pengaruh Faktor Suhu dan Kelembaban pada Lingkungan Kerja terhadap Pertumbuhan dan Perkembangan Mikroba. *JRSKT - Jurnal Riset Sains dan Kimia Terapan*, 9(2), pp.45–54. <https://doi.org/10.21009/jrskt.092.01>.
- PURWANTO, H., HERMAWAN, M., NAINGGOLAN, C. & WIBOWO, B., 2024. Design of an Automatic Temperature Recorder for Fishery Vessels Using Internet of Things Technology. *Journal of Applied Engineering and Technological Science*, 5(2), pp.1020–1050. <https://doi.org/10.37385/jaets.v5i2.4887>.
- RAMAKRISHNAIAH, V.K.H., LAKSHMAPPA, H., GURURAJ, B., MUNIYAPPA, R., SIDDARAMAIAH, P.G. AND BYLAMURTHY, N.H., 2024. A novel smart irrigation framework with timing allocation using solenoid valves and Arduino microcontroller. *International Journal of Reconfigurable and Embedded Systems (IJRES)*, 13(3), p.758. <https://doi.org/10.11591/ijres.v13.i3.pp758-766>.
- RAWAT, B., PURNAMA, S. AND MULYATI, 2021. MySQL Database Management System (DBMS) On FTP Site LAPAN Bandung. *International Journal of Cyber and IT Service Management (IJCITSM)*, [online] 1(2), pp.173–179. <https://doi.org/10.34306/ijcitsm.v1i1.47>.
- RIZAL, H.M., WARNI, E., ANGRIAWAN, R., HARIADI, M., ARIF, Y.M. & MAULINA, D. 2024. Design of Flood Early Detection Based on the Internet of Things and Decision Support System. *Ingenierie des Systemes d'Information*, 29(3), pp.1183–1193. <https://doi.org/10.18280/isi.290335>.
- SALMAN, A.G., HERYADI, Y., ABDURAHMAN,

- E. AND SUPARTA, W., 2018. Single Layer & Multi-layer Long Short-Term Memory (LSTM) Model with Intermediate Variables for Weather Forecasting. *Procedia Computer Science*, [online] 135, pp.89–98. <https://doi.org/10.1016/j.procs.2018.08.153>.
- SUFIAN, M.A., HAMZI, W., ZAMAN, S., ALSADDER, L., HAMZI, B., VARADARAJAN, J. & AZAD, M.A.K., 2024. Enhancing Clinical Validation for Early Cardiovascular Disease Prediction through Simulation, AI, and Web Technology. *Diagnostics*, 14(12), pp.1–44. <https://doi.org/10.3390/diagnostics14121308>.
- SUSILAWATI, S., SUSENO, S. & ROZIKIN, C., 2020. Sistem Monitoring Suhu Dan Kelembaban Ruang Produksi Berbasis Wireless Sensor Network Pada Pt. Xxx Manufacturing Services Indonesia. *JUST IT: Jurnal Sistem Informasi, Teknologi Informasi dan Komputer*, 10(2), p.136. <https://doi.org/10.24853/justit.10.2.136-143>.
- TAIB, S.A.T., ABU, N., SENAWI, A. & GO, C.K., 2025. The Implementation of Long-Short Term Memory for Tourism Industry in Malaysia. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, 46(2), pp.90–97. <https://doi.org/10.37934/araset.46.2.9097>.
- TEH, H.Y., KEMPA-LIEHR, A.W. & WANG, K.I.K., 2020. Sensor data quality: a systematic review. *Journal of Big Data*, [online] 7(1), pp.1–49. <https://doi.org/10.1186/s40537-020-0285-1>.
- VERIANTO, E., 2024. Mencegah Overfitting Pada Model Prediksi. *Jurnal Sistem Informasi dan Sistem Komputer*, 9(2), pp.195–204.
- WAHYUDI, J., ASBARI, M., SASONO, I., PRAMONO, T. & NOVITASARI, D., 2022. Database management in MySQL. *Edumaspul - Jurnal Pendidikan*, 6(2), pp.2413–2417.
- YALAMANCHILI, A., VENKATASEKHAR, D. & KUMAR, G.V., 2023. Attn _ CNN _ LSTM : IoT-Based Automated Alzheimer ' s Disease Classification using Deep Learning Approach. *International Journal of Engineering Trends and Technology*, 71(7), pp.132–146.
- YU, M., XU, F., HU, W., SUN, J. & CERVONE, G., 2021. Using Long Short-Term Memory (LSTM) and Internet of Things (IoT) for Localized Surface Temperature Forecasting in an Urban Environment. *IEEE Access*, 9, pp.137406–137418. <https://doi.org/10.1109/ACCESS.2021.3116809>.