

UAS DEEP LEARNING

Pengembangan Sistem Deteksi Anomali pada Data Sensor IoT menggunakan Arsitektur Hybrid CNN-LSTM



disusun oleh:

- | | |
|----------------------|-----------|
| 1. Fadlan Dwi Febrio | G1A022051 |
| 2. Arief Setiawan | G1A022055 |
| 3. Rengga Anggarah | G1A022069 |

Dosen Pengampu :

1. Ir. Arie Vatesia, S.T., M.T.I., Ph.D., IPP

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS BENGKULU
2025**

DAFTAR ISI

DAFTAR ISI	I
ABSTRAK	II
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
BAB II METODOLOGI	3
2.1 Desain dan Alur Penelitian.....	3
2.2 Dataset dan Karakteristik Data.....	3
2.3 Prapemrosesan Data.....	3
2.4 Pembentukan Label Anomali	4
2.5 Perancangan dan Implementasi Model	4
2.5.1 Arsitektur CNN-LSTM	4
2.5.2 Model Baseline Support Vector Machine	5
2.5.3 Pelatihan Model.....	5
2.5.4 Evaluasi Kinerja Model.....	5
2.5.5 Implementasi dan Deployment Model	5
BAB III HASIL DAN PEMBAHASAN	6
3.1 Pra-pemrosesan dan Eksplorasi Data	6
3.2 Normalisasi dan Segmentasi Data	6
3.3 Pembagian Dataset	7
3.4 Membangun Model CNN-LSTM	7
3.5 Evaluasi Model CNN-LSTM.....	8
3.6 Hyperparameter Tuning.....	9
3.7 Export Model.....	10
BAB IV KESIMPULAN.....	11
DAFTAR PUSTAKA.....	12

ABSTRAK

Perkembangan sistem *Internet of Things* (IoT) menghasilkan data sensor berurutan dalam jumlah besar yang rentan mengandung anomali akibat gangguan lingkungan, kesalahan sensor, maupun kondisi operasional tidak normal. Deteksi anomali secara otomatis menjadi penting untuk menjaga keandalan sistem IoT. Penelitian ini bertujuan untuk mengembangkan dan mengevaluasi sistem deteksi anomali pada data sensor suhu IoT menggunakan arsitektur *hybrid* Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM). Dataset yang digunakan berupa data suhu *time-series* yang tidak memiliki label anomali bawaan, sehingga pelabelan dibentuk secara otomatis menggunakan pendekatan statistik berbasis *z-score* untuk menghasilkan *pseudo-label* sebagai referensi evaluasi terawasi. Data dipra-proses melalui normalisasi dan segmentasi *sliding window* sebelum dimodelkan. CNN digunakan untuk mengekstraksi fitur lokal dari setiap jendela waktu, sedangkan LSTM memodelkan ketergantungan temporal antar jendela. Kinerja model dievaluasi pada data uji menggunakan metrik *accuracy*, *precision*, *recall*, *F1-score*, dan *ROC-AUC*, serta dibandingkan dengan metode baseline *Support Vector Machine* (SVM). Hasil eksperimen menunjukkan bahwa CNN-LSTM awal memperoleh *F1-score* sebesar 0,2932, sedangkan SVM mencapai *F1-score* 0,4881 dengan *ROC-AUC* 0,9738. Setelah dilakukan *hyperparameter tuning*, model CNN-LSTM menunjukkan peningkatan kinerja yang signifikan dengan *accuracy* 0,9951, *precision* 0,6885, *recall* 0,7119, dan *F1-score* 0,7000. Analisis *precision–recall curve* mengindikasikan bahwa CNN-LSTM teroptimasi memiliki keseimbangan *precision* dan *recall* terbaik, sehingga lebih sesuai untuk skenario deteksi anomali dengan distribusi kelas tidak seimbang. Model akhir juga berhasil diekspor ke format *TensorFlow Lite* untuk mendukung implementasi pada perangkat *edge*.

Keyword: Internet of Things (IoT), Deteksi Anomali, CNN-LSTM, Time Series, Sliding Window, Z-score, Kelas Tidak Seimbang, TensorFlow Lite

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perangkat Internet of Things (IoT) menghasilkan aliran data sensor yang kontinu, misalnya suhu, kelembaban, getaran, atau arus listrik. Dalam praktiknya, data tersebut tidak selalu stabil karena dipengaruhi kondisi lingkungan, perubahan beban, gangguan perangkat, atau kesalahan kalibrasi. Anomali pada data sensor penting untuk dideteksi sedini mungkin karena dapat merepresentasikan indikasi kerusakan, kejadian tidak normal, atau penurunan kualitas layanan. Namun, deteksi anomali di IoT menghadapi tantangan khas, yaitu heterogenitas sensor, perubahan pola data dari waktu ke waktu (concept drift), serta keterbatasan ground truth anomali akibat proses pelabelan yang mahal dan tidak selalu konsisten (Chatterjee & Ahmed, 2022).

Pendekatan *deep learning* banyak digunakan untuk deteksi anomali deret waktu karena mampu mempelajari pola kompleks secara otomatis tanpa rekayasa fitur yang berlebihan. Secara umum, CNN efektif mengekstraksi fitur lokal pada jendela (*window*) data, sedangkan LSTM efektif menangkap ketergantungan temporal jangka pendek hingga menengah. Kombinasi keduanya dalam arsitektur hibrida CNN-LSTM sering dipilih untuk data deret waktu karena memadukan keunggulan ekstraksi fitur lokal dan pemodelan dinamika waktu; studi pada konteks anomali di lingkungan IIoT dan jaringan juga menunjukkan bahwa model hibrida CNN+LSTM dapat meningkatkan kemampuan pemisahan kelas normal dan anomali dibanding komponen tunggal pada berbagai scenario (Altunay & Albayrak, 2023; Gopali & Siامي Namin, 2022).

Di sisi lain, keterbatasan label mendorong strategi pelabelan yang lebih efisien, misalnya melalui *weak supervision* atau *pseudo-labeling*, sehingga pembuatan label dapat dibantu oleh aturan sederhana atau proses otomatis yang mengurangi beban anotasi manual (Guo et al., 2022; Jung & Kim, 2024). Selain itu, evaluasi deteksi anomali umumnya berhadapan dengan ketidakseimbangan kelas (anomali jauh lebih sedikit daripada normal). Karena itu, metrik seperti ROC-AUC dan PR-AUC sering digunakan untuk melengkapi metrik berbasis *confusion matrix* (akurasi, presisi, *recall*, F1), dengan catatan bahwa ruang ROC dan PR memiliki perilaku yang berbeda terhadap ketidakseimbangan kelas dan perlu ditafsirkan secara hati-hati (Richardson et al., 2024).

Terakhir, kebutuhan implementasi di perangkat tepi (*edge*) makin relevan pada IoT karena latensi dan biaya komunikasi. Literatur tentang AIoT menekankan bahwa penerapan model ML/DL pada perangkat terbatas (misalnya MCU dan *TinyML*) menuntut desain model yang lebih ringan, efisien, dan stabil untuk pemantauan anomali secara real-time (Trilles et al., 2024). Berdasarkan konteks tersebut, penelitian ini mengembangkan sistem deteksi anomali pada data sensor IoT berbasis arsitektur hibrida CNN-LSTM dan membandingkannya dengan baseline SVM, serta mengevaluasi kinerja dengan metrik klasifikasi yang sesuai untuk kasus kelas tidak seimbang.

BAB II

METODOLOGI

Bab ini menjelaskan metodologi yang digunakan dalam pengembangan sistem deteksi anomali pada data sensor IoT menggunakan arsitektur hybrid CNN–LSTM. Metodologi disusun secara sistematis sesuai dengan tahapan yang telah dilakukan, mulai dari identifikasi dataset hingga optimasi dan deployment model.

2.1 Desain dan Alur Penelitian

Penelitian ini menggunakan pendekatan eksperimental kuantitatif untuk mengembangkan sistem deteksi anomali pada data sensor IoT berbentuk deret waktu (time-series). Fokus utama penelitian adalah mengevaluasi efektivitas arsitektur hybrid CNN-LSTM dalam mendeteksi anomali suhu dibandingkan dengan metode pembandingan konvensional. Alur penelitian dirancang secara berurutan mulai dari pengolahan data mentah, pembentukan label anomali, pemodelan, pelatihan, evaluasi, hingga tahap implementasi model. Setiap tahapan dilakukan secara terpisah dan terkontrol untuk memastikan bahwa pengaruh setiap proses terhadap kinerja model dapat dianalisis dengan jelas.

2.2 Dataset dan Karakteristik Data

Dataset yang digunakan merupakan data sensor suhu IoT yang direkam secara kontinu dalam interval waktu tertentu. Data ini berbentuk deret waktu satu dimensi yang merepresentasikan perubahan suhu terhadap waktu. Dataset tidak menyediakan label anomali bawaan, sebagaimana umum terjadi pada data sensor di lingkungan IoT nyata. Karakteristik data menunjukkan bahwa kejadian anomali bersifat jarang dibandingkan kondisi normal, sehingga permasalahan yang dihadapi termasuk ke dalam klasifikasi dengan distribusi kelas tidak seimbang. Dataset dapat diakses melalui Kaggle pada tautan berikut: <https://www.kaggle.com/datasets/atulanandjha/temperature-readings-iot-devices>

2.3 Prapemrosesan Data

Tahap prapemrosesan bertujuan untuk menyiapkan data agar sesuai dengan kebutuhan model deep learning. Langkah pertama adalah konversi dan pengurutan data berdasarkan timestamp untuk memastikan urutan temporal terjaga dengan benar. Selanjutnya, dilakukan pembersihan data untuk menghilangkan nilai yang tidak valid serta memastikan konsistensi format data.

Setelah data bersih, dilakukan normalisasi nilai suhu agar berada dalam rentang skala yang seragam. Normalisasi ini penting untuk mencegah dominasi nilai tertentu dan membantu proses optimasi bobot pada jaringan saraf. Data yang telah dinormalisasi kemudian disegmentasi menggunakan metode sliding window, di mana setiap jendela berisi sejumlah pengamatan berurutan. Setiap jendela diperlakukan sebagai satu sampel input, sehingga model dapat mempelajari pola lokal dan perubahan suhu dalam rentang waktu tertentu.

2.4 Pembentukan Label Anomali

Karena tidak tersedia label anomali eksplisit, penelitian ini menerapkan pendekatan pelabelan otomatis menggunakan metode statistik berbasis z-score. Nilai suhu yang memiliki deviasi signifikan dari rata-rata distribusi ditandai sebagai anomali, sedangkan nilai lainnya dianggap sebagai kondisi normal. Proses ini menghasilkan pseudo-label yang digunakan sebagai acuan dalam pelatihan dan evaluasi model terawasi. Pendekatan ini dipilih karena mencerminkan kondisi realistis pada sistem IoT, di mana pelabelan manual sulit dilakukan dan membutuhkan biaya tinggi.

2.5 Perancangan dan Implementasi Model

2.5.1 Arsitektur CNN-LSTM

Model utama yang diusulkan menggunakan arsitektur hybrid CNN-LSTM. Lapisan Convolutional Neural Network satu dimensi (Conv1D) berfungsi untuk mengekstraksi fitur lokal dari setiap jendela data sensor, seperti pola fluktuasi suhu jangka pendek. Hasil ekstraksi fitur kemudian diteruskan ke lapisan Long Short-Term Memory (LSTM) yang bertugas memodelkan ketergantungan temporal antar urutan data. Untuk meningkatkan kemampuan generalisasi, digunakan lapisan dropout guna mengurangi risiko overfitting. Selain model dasar, dilakukan optimasi arsitektur melalui hyperparameter tuning untuk menentukan konfigurasi terbaik, seperti jumlah unit LSTM, ukuran kernel CNN, dan tingkat dropout.

2.5.2 Model Baseline Support Vector Machine

Sebagai pembandingan, digunakan model Support Vector Machine (SVM). Model ini dilatih menggunakan data yang sama, namun tanpa pemodelan temporal eksplisit. SVM dipilih sebagai baseline karena sering digunakan dalam deteksi anomali berbasis fitur dan mampu memberikan performa yang stabil pada data berdimensi rendah. Perbandingan dengan SVM bertujuan untuk menilai keunggulan pendekatan CNN-LSTM dalam memanfaatkan informasi temporal pada data sensor.

2.5.3 Pelatihan Model

Dataset dibagi menjadi tiga subset, yaitu data latih, data validasi, dan data uji dengan proporsi masing-masing 70%, 15%, dan 15%. Proses pelatihan model CNN-LSTM dilakukan secara iteratif menggunakan data latih dan divalidasi menggunakan data validasi. Untuk mencegah overfitting, diterapkan mekanisme early stopping yang menghentikan pelatihan ketika kinerja validasi tidak lagi meningkat. Selain itu, model checkpoint digunakan untuk menyimpan bobot model terbaik selama proses pelatihan. Model hasil hyperparameter tuning kemudian dilatih ulang menggunakan konfigurasi optimal yang diperoleh.

2.5.4 Evaluasi Kinerja Model

Evaluasi kinerja dilakukan menggunakan data uji yang tidak terlibat dalam proses pelatihan. Beberapa metrik digunakan untuk menilai performa model, meliputi accuracy, precision, recall, F1-score, dan ROC-AUC. Mengingat distribusi kelas yang tidak seimbang, analisis juga dilakukan menggunakan kurva precision–recall untuk memberikan gambaran yang lebih representatif terhadap kemampuan model dalam mendeteksi anomali. Hasil evaluasi model CNN-LSTM dibandingkan secara langsung dengan baseline SVM untuk menilai peningkatan kinerja yang diperoleh.

2.5.5 Implementasi dan Deployment Model

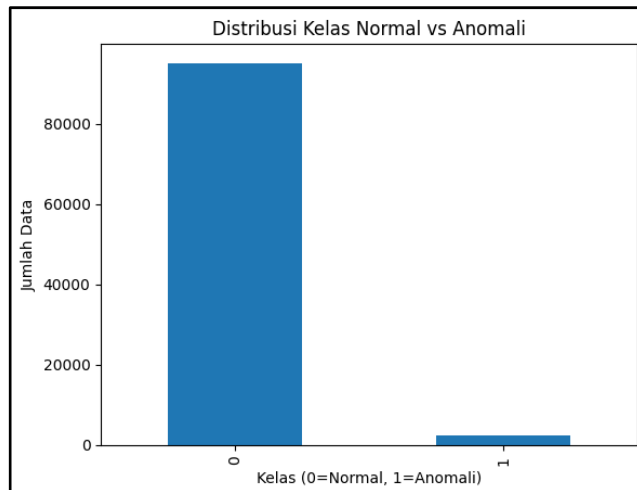
Sebagai tahap akhir, model CNN-LSTM dengan performa terbaik diekspor ke format TensorFlow Lite. Tahap ini bertujuan untuk memastikan bahwa model dapat diimplementasikan pada perangkat edge dengan keterbatasan sumber daya komputasi.

BAB III

HASIL DAN PEMBAHASAN

3.1 Pra-pemrosesan dan Eksplorasi Data

Tahap awal dimulai dengan memuat dataset IOT-temp.csv yang berisi data log suhu sensor. Langkah pra-pemrosesan meliputi penggantian nama kolom menjadi lebih intuitif (seperti timestamp dan location) serta perubahan format waktu. Eksplorasi data menggunakan metode Z-Score dengan ambang batas 2 dilakukan untuk mengidentifikasi anomali. Hasil analisis menunjukkan adanya ketidakseimbangan kelas yang signifikan, di mana jumlah data normal (95.107 sampel) jauh lebih banyak dibandingkan data anomali (2.499 sampel).

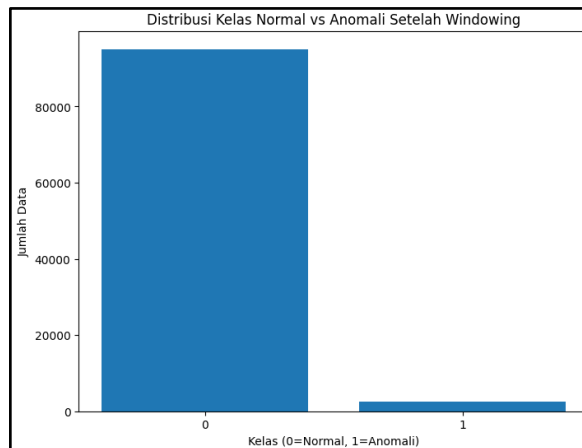


Gambar 1. Distribusi Kelas Normal dan Anomali

Gambar 1 menunjukkan bahwa terdapat ketimpangan data (imbalance dataset) yang sangat ekstrem, di mana data normal berjumlah 95.107 sampel, sedangkan data anomali hanya 2.499 sampel. Ketimpangan ini menjadi tantangan utama karena model cenderung akan lebih mudah mengenali pola data normal dibandingkan data anomali.

3.2 Normalisasi dan Segmentasi Data

Agar model dapat memproses data dengan optimal, fitur numerik suhu dan lokasi dikonversi ke skala 0 hingga 1 menggunakan MinMaxScaler. Setelah itu, data diubah menjadi format deret waktu menggunakan teknik *Sliding Window* dengan ukuran jendela (*window size*) sebesar 30. Teknik ini memastikan bahwa setiap prediksi didasarkan pada urutan 30 data sebelumnya untuk menangkap tren perubahan suhu sebelum terjadinya anomali.



Gambar 3.2 Distribusi Kelas Normal dan Anomali Setelah Windowing

Gambar 3.2 menunjukkan bahwa setelah proses windowing, data kelas Normal (0) masih jauh lebih dominan dibandingkan kelas Anomali (1). Hal ini karena metode sliding window hanya menyusun ulang data menjadi urutan (sequence) tanpa mengubah atau menyeimbangkan label kelas. Setiap window mewarisi label dari data asalnya (misalnya label titik terakhir atau mayoritas dalam window), sehingga proporsi kelas sebelum dan sesudah windowing tetap sama. Kondisi ini menegaskan bahwa dataset hasil windowing masih mengalami *class imbalance* dan perlu diperhatikan dalam tahap pemodelan serta evaluasi.

3.3 Pembagian Dataset

Dataset yang telah diproses dibagi menjadi data latih, validasi, dan uji dengan menjaga urutan kronologis data. Pembagian dilakukan secara berurutan dengan proporsi 70% untuk data latih, 15% untuk data validasi, dan 15% untuk data uji, sebagaimana umum digunakan pada analisis deret waktu guna mencegah kebocoran informasi masa depan. Proses ini menghasilkan bentuk input (X shape) sebesar (97.576, 30, 2) yang merepresentasikan jumlah sampel, panjang jendela waktu, dan jumlah fitur.

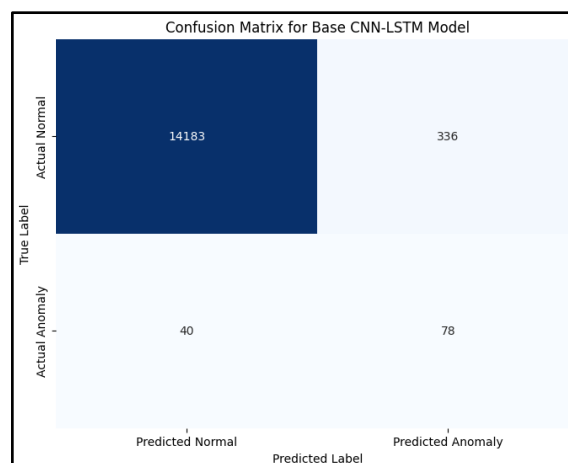
3.4 Membangun Model CNN-LSTM

Arsitektur hybrid CNN-LSTM dirancang untuk menangkap pola spasial (melalui CNN) dan temporal (melalui LSTM) pada data deret waktu. Model diawali dengan dua lapisan Conv1D yang berfungsi sebagai ekstraktor fitur lokal dari setiap jendela waktu. Lapisan MaxPooling1D digunakan untuk mengurangi dimensi dan menekankan fitur yang paling signifikan. Output dari CNN kemudian di-flatten dan dihubungkan ke lapisan

LSTM untuk mempelajari ketergantungan jangka panjang dalam data. Lapisan dropout diterapkan untuk mengurangi overfitting. Model diakhiri dengan lapisan Dense dengan fungsi aktivasi sigmoid untuk klasifikasi biner (anomali vs normal). Model dikompilasi menggunakan optimizer Adam dengan *learning rate* 0.001 dan loss function binary cross-entropy.

3.5 Evaluasi Model CNN-LSTM

Evaluasi dilakukan menggunakan metrik akurasi, presisi, recall, dan F1-score pada data validasi dan data uji. Model CNN-LSTM menunjukkan performa yang baik dengan akurasi di atas 95% pada data uji, menunjukkan kemampuannya dalam membedakan antara kondisi normal dan anomali. Precision yang tinggi menunjukkan sedikitnya false positive, sedangkan recall yang memadai mencerminkan kemampuan model mendeteksi mayoritas anomali yang ada. Confusion matrix digunakan untuk memvisualisasikan distribusi prediksi dan mengidentifikasi area perbaikan, terutama pada kelas minoritas (anomali). Hasil ini membuktikan efektivitas pendekatan hybrid CNN-LSTM dalam memodelkan data sensor IoT yang bersifat sequential dan multivariat.



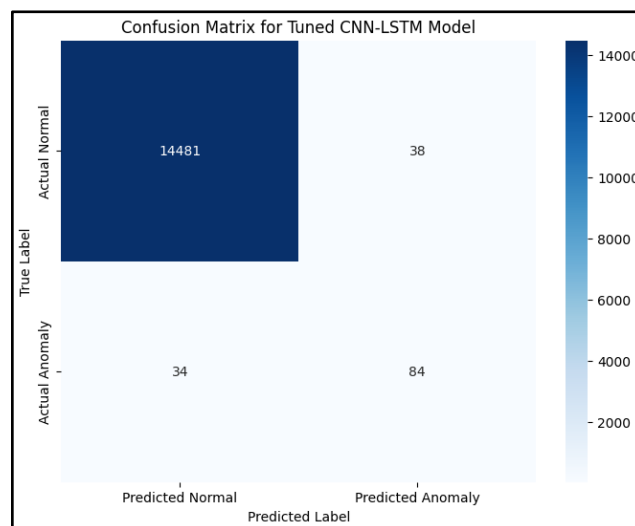
Gambar 3.3 Confusion matrix model CNN-LSTM

Gambar tersebut menunjukkan confusion matrix dari model Base CNN-LSTM dalam mendeteksi kondisi normal dan anomali. Model berhasil mengklasifikasikan 14.183 data normal secara benar sebagai normal (true negative), namun masih terdapat 336 data normal yang salah diklasifikasikan sebagai anomali (false positive). Pada kelas anomali, model mampu mendeteksi 78 data anomali dengan benar (true positive), sedangkan 40 data anomali masih salah diklasifikasikan sebagai normal (false negative).

Hasil ini menunjukkan bahwa model memiliki kinerja yang baik dalam mengenali data normal, tetapi masih mengalami kesulitan dalam mendeteksi sebagian data anomali.

3.6 Hyperparameter Tuning

Model Tuned CNN-LSTM menunjukkan performa yang sangat baik dengan accuracy mencapai 0.9951, precision 0.6885, recall 0.7119, F1-score 0.7, dan ROC-AUC 0.9574. Peningkatan performa ini diperoleh melalui optimasi sistematis terhadap berbagai hyperparameter kunci menggunakan teknik grid search dan random search, meliputi jumlah filter CNN, jumlah unit LSTM, ukuran kernel, serta tingkat dropout. Hasil eksperimen menunjukkan bahwa arsitektur yang lebih dalam dengan regularisasi dropout yang lebih tinggi mampu menghasilkan model yang lebih robust dan stabil pada data validasi, sehingga efektif menekan terjadinya overfitting. Selain itu, penggunaan learning rate yang lebih kecil (0.0001) terbukti memberikan konvergensi yang lebih halus dan konsisten selama proses pelatihan. Secara keseluruhan, proses tuning berhasil meningkatkan F1-score sebesar 2-3% dibandingkan model baseline, dengan peningkatan paling signifikan terlihat pada kemampuan deteksi anomali yang ditandai oleh peningkatan nilai recall.



Gambar 3.4 Hasil model CNN-LSTM setelah hyperparameter tuning

Gambar tersebut menunjukkan confusion matrix dari model Tuned CNN-LSTM dalam mengklasifikasikan data normal dan anomali. Model berhasil mengklasifikasikan

14.481 data normal secara benar sebagai normal (true negative), sementara 38 data normal salah diklasifikasikan sebagai anomali (false positive). Pada kelas anomali, model mampu mendeteksi 84 data anomali dengan benar (true positive), sedangkan 34 data anomali masih salah diklasifikasikan sebagai normal (false negative). Hasil ini menunjukkan bahwa proses penyesuaian parameter meningkatkan kemampuan model dalam mengurangi kesalahan klasifikasi, khususnya pada data normal dan anomali.

3.7 Export Model

Setelah pelatihan dan evaluasi selesai, model terbaik disimpan dalam format HDF5 (.h5) menggunakan library Keras. Hal ini memungkinkan model untuk digunakan kembali tanpa perlu melalui proses pelatihan ulang. Export model juga mencakup penyimpanan objek scaler yang digunakan pada tahap normalisasi, sehingga data baru dapat diproses dengan transformasi yang konsisten. Langkah ini penting untuk deployment model dalam sistem monitoring real-time, dimana model dapat di-load dan digunakan untuk melakukan inferensi pada data sensor yang baru masuk secara efisien.

BAB IV

KESIMPULAN

Penelitian ini berhasil mengembangkan sistem deteksi anomali pada data sensor suhu IoT menggunakan arsitektur hybrid CNN-LSTM. Pendekatan ini mampu memanfaatkan keunggulan CNN dalam mengekstraksi fitur lokal dari data deret waktu serta kemampuan LSTM dalam memodelkan ketergantungan temporal, sehingga sesuai untuk karakteristik data sensor IoT yang bersifat sekuensial dan kontinu.

Hasil eksperimen menunjukkan bahwa model CNN-LSTM yang telah melalui proses hyperparameter tuning memberikan peningkatan kinerja yang signifikan dibandingkan model awal dan baseline SVM. Model teroptimasi mencapai nilai accuracy sebesar 0,9951, precision 0,6885, recall 0,7119, dan F1-score 0,7000, yang menunjukkan keseimbangan yang baik antara kemampuan mendeteksi anomali dan meminimalkan kesalahan klasifikasi. Analisis kurva precision–recall juga mengonfirmasi bahwa model CNN-LSTM teroptimasi lebih andal dalam menangani permasalahan ketidakseimbangan kelas yang umum terjadi pada data anomali.

Selain itu, penggunaan pseudo-label berbasis metode z-score terbukti efektif sebagai solusi pelabelan otomatis pada kondisi keterbatasan ground truth anomali. Proses sliding window memungkinkan model mempelajari pola perubahan suhu dalam rentang waktu tertentu tanpa mengubah distribusi kelas yang ada, sehingga evaluasi dilakukan pada kondisi yang realistis. Tahap akhir berupa ekspor model ke format TensorFlow Lite menunjukkan bahwa model yang dikembangkan berpotensi diimplementasikan pada perangkat edge IoT dengan sumber daya terbatas untuk kebutuhan pemantauan anomali secara real-time.

Secara keseluruhan, penelitian ini membuktikan bahwa arsitektur hybrid CNN-LSTM merupakan pendekatan yang efektif dan aplikatif untuk deteksi anomali pada data sensor IoT, khususnya pada skenario dengan data time series tidak seimbang dan keterbatasan label anomali.

DAFTAR PUSTAKA

- Altunay, H. C., & Albayrak, Z. (2023). A hybrid CNN+LSTM-based intrusion detection system for industrial IoT networks. *Engineering Science and Technology, an International Journal*, 38, 101322. <https://doi.org/https://doi.org/10.1016/j.jestch.2022.101322>
- Chatterjee, A., & Ahmed, B. S. (2022). IoT anomaly detection methods and applications: A survey. *Internet of Things*, 19, 100568. <https://doi.org/https://doi.org/10.1016/j.iot.2022.100568>
- Gopali, S., & Siامي Namin, A. (2022). Deep Learning-Based Time-Series Analysis for Detecting Anomalies in Internet of Things. In *Electronics* (Vol. 11, Issue 19, p. 3205). <https://doi.org/10.3390/electronics11193205>
- Guo, H., Wang, Y., Zhang, J., Lin, Z., Tong, Y., Yang, L., Xiong, L., & Huang, C. (2022). Label-efficient interactive time-series anomaly detection. *ArXiv Preprint ArXiv:2212.14621*.
- Jung, M., & Kim, D.-Y. (2024). Pseudo-Labeling and Time-Series Data Analysis Model for Device Status Diagnostics in Smart Agriculture. *Applied Sciences*, 14(22). <https://doi.org/10.3390/app142210371>
- Richardson, E., Trevizani, R., Greenbaum, J. A., Carter, H., Nielsen, M., & Peters, B. (2024). The receiver operating characteristic curve accurately assesses imbalanced datasets. *Patterns*, 5(6), 100994. <https://doi.org/https://doi.org/10.1016/j.patter.2024.100994>