

# Laporan AI Lanjut Kelompok 8.pdf

*by Akun Cek*

---

**Submission date:** 04-Jan-2026 05:12PM (UTC+0900)

**Submission ID:** 2852545042

**File name:** Laporan\_AI\_Lanjut\_Kelompok\_8.pdf (440.64K)

**Word count:** 1456

**Character count:** 9247

# Laporan Final Project Kecerdasan Buatan (Lanjutan)

*Penerapan Deep Learning untuk Anomaly Detection pada  
Predictive Maintenance Menggunakan LSTM dan Transformer*



Kelompok 08

Anggota Kelompok:

1.23.11.5701 | Alief El Faqih Yulizar

2.23.11.5718 | Husni Nur Dzaki

3.23.11.5693 | Alim Dama Wicaksono

## 1. Latar Belakang

Di dunia industri manufaktur, kelancaran pada lini produksi bukan sekadar target, melainkan fondasi utama sebuah bisnis. Tantangan paling berat sering kali muncul dari arah yang sulit diprediksi, seperti kerusakan mesin secara mendadak. Insiden ini bukan hanya gangguan teknis yang sepele, melainkan krusial bagi profit perusahaan karena memicu henti produksi (downtime) yang merugikan. Kondisi inilah yang mendorong urgensi penerapan Predictive Maintenance. Strategi ini mengubah pandangan perawatan mesin, dari yang semula memperbaiki saat rusak menjadi proaktif dengan mengandalkan langkah antisipasi dini.

Guna menjawab tantangan tersebut, pendekatan Anomaly Detection menjadi pilihan paling rasional. Sering kali, pengembang AI kesulitan karena data kerusakan mesin di lapangan sangat langka. Oleh karena itu, daripada memaksa model mempelajari kegagalan yang minim datanya, kami melatih sistem untuk memahami kondisi "Normal". Idennya adalah membangun pemahaman mendalam pada AI tentang pola kerja standar mesin, sehingga setiap penyimpangan atau perilaku aneh yang keluar dari jalur normal akan langsung terdeteksi sebagai sinyal awal potensi kerusakan [3], [5].

Dalam riset ini, kami memanfaatkan kecanggihan Deep Learning berbasis sequence untuk membedah data sensor yang rumit dan dinamis. Kami mengomparasikan dua arsitektur raksasa: LSTM Autoencoder [1], yang sudah teruji keandalannya dalam mengolah data runtun waktu, serta Transformer Autoencoder [2], yang memiliki keunggulan pada mekanisme atensi (attention mechanism) untuk menangkap hubungan antar-data secara lebih tajam. Fokus utama eksperimen ini adalah menguji efektivitas keduanya dalam mengendus anomali, bahkan tanpa perlu menyertakan satu pun data kegagalan selama proses pelatihan model.

## 2. Metode

### 2.1 Alur Pengerjaan (*Project Flow*)

Alur pengerjaan proyek ini dirancang secara sistematis mulai dari pemahaman data hingga evaluasi model:

1. **EDA:** Mengidentifikasi sekilas persebaran dan pola data.
2. **Feature Selection:** Menggunakan metode *Backward Elimination* berbasis OLS (*Ordinary Least Squares*) untuk menyeleksi sensor mana yang paling signifikan secara statistik, sehingga model hanya fokus pada fitur yang relevan.

3. **Sequence Generation:** Karena data sensor bersifat runtun waktu, kami menerapkan teknik *Sliding Window*. Teknik ini mengubah data menjadi potongan-potongan urutan (*sequence*) sehingga model dapat mempelajari pola historis, bukan hanya data sesaat.
4. **Pelatihan (Training):** Melatih model Autoencoder hanya menggunakan data normal untuk meminimalisir *reconstruction error*.
5. **Evaluasi:** Menguji model menggunakan data campuran (normal dan *failure*) untuk melihat seberapa akurat model menandai data *failure* sebagai anomali.

## 2.2 Algoritma: Autoencoder (LSTM & Transformer)

Konsep utama yang digunakan adalah *Autoencoder*, di mana model bertugas mengompresi data input (Encoder) lalu menyusunnya kembali (Decoder). Jika model dilatih dengan data normal, maka ia akan kesulitan menyusun ulang data kerusakan (error rekonstruksi tinggi). Error inilah yang menjadi indikator anomali. Dua arsitektur yang diuji adalah:

- **LSTM Autoencoder:** Menggunakan struktur Encoder-Decoder berbasis LSTM [4] dengan lapisan *bottleneck*. Model ini dipilih karena kemampuannya mengingat dependensi jangka panjang pada data sensor.
- **Transformer Autoencoder:** Memanfaatkan *Positional Encoding* dan *Multi-Head Self-Attention* [2]. Berbeda dengan LSTM yang memproses urutan secara linear, Transformer mampu melihat hubungan global antar *time step* secara simultan, yang diharapkan dapat menangkap pola anomali yang lebih halus [5].

## 3. Dataset

Dataset yang digunakan dalam project ini kami ambil dari kaggle. Dataset ini dipilih karena merepresentasikan kondisi nyata suatu sensor dari mesin dengan karakteristik sebagai berikut:

- Data Sensor: Terdiri dari 5 parameter utama yaitu suhu udara, suhu proses, kecepatan rotasi, torsi, dan tingkat keausan alat (tool wear).
- Data Record kerusakan: 9661 Mesin normal, dan 339 Mesin rusak
- Data Jenis Kerusakan : 5 jenis kerusakan yaitu TWF,HDF,PWF, OSF dan RNF
- Volume Data: Sekitar 10.000 data pengamatan.

## 4. Hasil Pengujian

### 4.1 Skenario Pengujian

- **Dataset**

- Dataset: AI4I 2020 Predictive Maintenance Dataset
- Data normal digunakan **sepenuhnya untuk pelatihan**
- Data failure **tidak digunakan saat training**, hanya digunakan pada tahap pengujian
- Data jenis Kerusakan yang lain tidak digunakan juga karena jumlahnya sangat sedikit.

- **Pembagian Data (Time-Ordered Split)**

- 60% data normal → Training
- 20% data normal → Validation
- 20% data normal + seluruh data failure → Testing

Pendekatan ini dipilih untuk **menghindari data leakage** dan mensimulasikan kondisi industri nyata, di mana model hanya memiliki akses ke data normal saat fase pembelajaran.

- **Konfigurasi Model**

- LSTM Autoencoder**

- Arsitektur:
  - Encoder: LSTM (32 unit)
  - Bottleneck: representasi laten tunggal
  - Decoder: LSTM (32 unit)
- Epoch maksimum: 50
- Batch size: 32
- Early Stopping:
  - Monitor: validation loss
  - Patience: 5 epoch

- Transformer Autoencoder**

- Arsitektur:

- Positional Encoding
  - Encoder Transformer Blocks (Multi-Head Attention + Feed Forward)
  - Bottleneck Dense Layer (compression ratio 0.5)
  - Decoder Transformer Blocks
- Epoch maksimum: 50
- Batch size: 32
- Early Stopping:
  - Monitor: validation loss
  - Patience: 5 epoch

- **Metode Evaluasi**

Evaluasi model dilakukan menggunakan pendekatan **anomaly detection berbasis reconstruction error**, dengan langkah sebagai berikut:

1. Model dilatih hanya menggunakan data normal
2. Reconstruction error dihitung menggunakan Mean Squared Error
3. Threshold ditentukan dari distribusi reconstruction error data training (persentil ke-90)
4. Data dengan error di atas threshold diklasifikasikan sebagai anomali
5. Performa diukur menggunakan:
  - Confusion Matrix
  - <sup>1</sup> Precision
  - Recall
  - F1-score

## 4.2 Hasil Pengujian

### LSTM Autoencoder

Berdasarkan pengujian pada data test (normal unseen + failure), diperoleh hasil sebagai berikut:

```
*** LSTM Autoencoder - Anomaly Detection Report:
      precision    recall  f1-score   support

      0.0         0.99      0.93      0.96      1931
      1.0         0.71      0.95      0.81       329

   accuracy              0.94      2260
  macro avg              0.85      0.94      0.89      2260
 weighted avg              0.95      0.94      0.94      2260

Confusion Matrix:
[[1800  131]
 [   15  314]]
```

Confusion Matrix menunjukkan bahwa sebagian besar kondisi failure berhasil terdeteksi, meskipun terdapat sejumlah false positive pada data normal.

**Transformer Autoencoder**

Hasil pengujian Transformer Autoencoder menunjukkan performa yang sebanding, dengan karakteristik sebagai berikut:

```
*** Transformer Autoencoder - Evaluation
      precision    recall  f1-score   support

      0.0         0.95      0.98      0.96      1931
      1.0         0.85      0.70      0.77       329

   accuracy              0.94      2260
  macro avg              0.90      0.84      0.87      2260
 weighted avg              0.94      0.94      0.94      2260

Confusion Matrix:
[[1892   39]
 [  100  229]]
```

- Deteksi anomali lebih sensitif terhadap perubahan pola global
- Reconstruction error pada kondisi drift meningkat lebih cepat dibandingkan LSTM
- Recall failure tinggi, namun precision cenderung sedikit lebih rendah akibat sensitivitas model terhadap fluktuasi kecil

## 5. Analisa Hasil

### 5.1 Efektivitas Deteksi Anomali

Kedua model terbukti sukses menerapkan konsep *Anomaly Detection*. Tanpa pernah melihat data kerusakan sebelumnya, model mampu memberikan sinyal *error* yang tinggi saat disodorkan data *failure*. Ini membuktikan bahwa pendekatan *unsupervised* pada data normal sangat efektif untuk mengatasi kelangkaan data kerusakan.

### 5.2 Karakteristik Model:

- **LSTM Autoencoder:** Menunjukkan performa yang sangat stabil. Model ini menghasilkan nilai Recall yang tinggi pada kelas failure, yang berarti risiko kerusakan yang "lolos" dari deteksi sangat kecil. LSTM juga cenderung lebih tahan terhadap noise fluktuasi kecil [7].
- **Transformer Autoencoder:** responsif terhadap perubahan pola global berkat mekanisme Self-Attention. Meskipun sensitif/dapat mendeteksi drift jangka panjang, model ini punya sedikit kelemahan yaitu menghasilkan False Positive yang lebih banyak dibanding LSTM di kondisi perubahan ringan.

### 5.3 Early Warning

Simulasi pada skenario *NORMAL – DRIFT – FAILURE* menunjukkan bahwa grafik *error* rekonstruksi meningkat secara bertahap sebelum kegagalan total terjadi. Hal ini mengindikasikan bahwa sistem ini potensial digunakan sebagai alat peringatan dini di industri [8].

## 6. Kesimpulan

Dari hasil model yang sudah kita latih, kedua model ini berhasil mempelajari data sensor dan bisa membedakan kondisi mesin yang normal dan mesin yang abnormal. Namun, dari hasil evaluasi, model LSTM lebih stabil dalam mempelajari pola berurutan dari data sensor, sedangkan Transformer lebih unggul jika digunakan untuk menangkap pola hubungan global antar fitur dengan mekanisme self-attention



## 7. Referensi

- [1] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- [2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need. *arXiv*. <https://arxiv.org/abs/1706.03762>
- [3] Bruneo, D., & De Vita, F. (2019, June). On the Use of LSTM Networks for Predictive Maintenance in Smart Industries. In *2019 IEEE International Conference on Smart Computing (SMARTCOMP)* (pp. 241–248). IEEE. <https://doi.org/10.1109/SMARTCOMP.2019.00059>
- [4] Mitiche, I., McGrail, T., Boreham, P., Nesbitt, A., & Morison, G. (2021). Data-Driven Anomaly Detection in High-Voltage Transformer Bushings with LSTM Auto-Encoder. *Sensors*, 21(21), 7426. <https://doi.org/10.3390/s21217426>
- [5] Tuli, S., Casale, G., & Jennings, N. R. (2022). TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data. *arXiv preprint arXiv:2201.07284*. <https://doi.org/10.48550/arXiv.2201.07284>
- [6] Yan, X., He, C., Yin, J., Zhang, C., & Li, H. (2023). Voltage Transformer Anomaly Detection Using LSTM Based Autoencoder. *2023 5th International Conference on Data-Driven Optimization of Complex Systems (DOCS)*, 1–6. <https://doi.org/10.1109/DOCS60977.2023.10294839>
- [7] Chen, Y., Liu, X., Fan, W., Duan, N., & Zhou, K. (2024). A Deep-LSTM-Based Fault Detection Method for Railway Vehicle Suspensions. *Machines*, 12(2), 116. <https://doi.org/10.3390/machines12020116>
- [8] Rahman, M., Hossain, M. S., Rozario, U., Roy, S., Mridha, M. F., & Dey, N. (2025). MultiSenseNet: Multi-Modal Deep Learning for Machine Failure Risk Prediction. *IEEE Access*, 13, 120404–120420. <https://doi.org/10.1109/ACCESS.2025.3524673>

## **8. Kontribusi & distribusi anggota kelompok**

- Alief El Faqih Yulizar (23.11.5701):
  - Membantu penyusunan laporan
  - Memastikan susunan pada paragraf dan kesesuaian bahasa
- Husni Nur Dzaki (23.11.5718):
  - Implementasi seluruh kode program (*coding*)
  - Melakukan eksplorasi dan pengumpulan dataset dari sumber open source (Kaggle)
  - Memastikan kesesuaian isi laporan dengan project
- Alim Dama Wicaksono:
  - Membuat laporan awal keseluruhan
  - Menyusun dan merapikan format laporan
  - Pencarian referensi jurnal terkait.

# Laporan AI Lanjut Kelompok 8.pdf

## ORIGINALITY REPORT

1 %

SIMILARITY INDEX

1 %

INTERNET SOURCES

0 %

PUBLICATIONS

0 %

STUDENT PAPERS

## PRIMARY SOURCES

1

[openlibrarypublications.telkomuniversity.ac.id](http://openlibrarypublications.telkomuniversity.ac.id)

Internet Source

1 %

Exclude quotes On

Exclude matches Off

Exclude bibliography On