

# IMPLEMENTASI ALGORITMA PERAMALAN PADA SISTEM MONITORING KAPASITAS SAMPAH BERBASIS IOT SEBAGAI UPAYA PENGELOLAAN TIMBULAN SAMPAH

Annisa Putri Setianingrum<sup>1\*)</sup>, Enda Wista Sinuraya<sup>2</sup> dan Yosua Alvin Adi Soetrisno<sup>3</sup>

<sup>123</sup>Program Studi Sarjana Departemen Teknik Elektro, Universitas Diponegoro  
Jl. Prof. Sudarto, SH, Kampus UNDIP Tembalang, Semarang 50257, Indonesia

<sup>\*)</sup>Penulis korespondensi, E-mail: [annisaps@students.undip.ac.id](mailto:annisaps@students.undip.ac.id)

## Abstrak

Prinsip pembangunan berkelanjutan menjadi strategi dan acuan dalam segala aspek kehidupan berwawasan lingkungan. Melalui program kompetisi *Green Campus*, universitas membantu menjalankan misi konservasi lingkungan hidup. Poin penilaian dalam kompetisi tersebut memengaruhi kebijakan di Universitas Diponegoro, contohnya pengelolaan sampah. Namun kebijakan pengelolaan sampah hanya mencakup kegiatan pengolahan di hulu dan hilir saja. Padahal proses pengelolaan sampah terdiri dari penanganan, pengolahan, pembuangan atau daur ulang. Penelitian mengusulkan sistem monitoring berbasis IoT dilengkapi fitur prediksi guna mengatasi keterbatasan pengelolaan sampah konvensional dalam memantau dan memprediksi kondisi tempat sampah. Arsitektur yang dibangun di dua *node* mengukur tingkat tidak terisinya tempat sampah, memprosesnya dan mengirimkan ke *database*. Data ketinggian sampah kemudian dikomputasi menggunakan Jaringan Saraf Tiruan metode Levenberg Marquardt untuk memprediksi ketinggian sampah dalam 2 jam. Hasil pengujian komunikasi IoT menunjukkan rata-rata *delay* pengiriman data ke *database* sebesar 2,886s dan 2,912s dengan 0% packet loss. Koefisien korelasi yang dihasilkan pada proses pelatihan model Levenberg Marquardt yaitu 0,925 dan 0,965. Perbandingan data prediksi dengan data aktual pembacaan sensor menunjukkan bahwa model ini memiliki tingkat rata-rata *error* kuadrat senilai 8,96cm dan 13,04cm.

*Kata kunci: green campus, pengelolaan sampah, IoT, Jaringan Saraf Tiruan*

## Abstract

The principle of sustainable development becomes a strategy and reference in all aspects of life with an environmental perspective. University helps carry out environmental conservation missions through the Green Campus competition program. The competition's scoring points influence Diponegoro University policies, for example, waste management. However, the waste management policy only covers upstream and downstream processing activities. In contrast, the waste management process consists of handling, processing, disposal, or recycling. The study proposes an IoT-based monitoring system equipped with a prediction feature to overcome the limitations of conventional waste management in monitoring and predicting trash bin conditions. The architecture built on two nodes measures the unfilled level of the bin, processes, and sends it to the database. The data then computed using Levenberg Marquardt's Artificial Neural Network to predict the height of the garbage in 2 hours. The results of the IoT communication show that the average delay in sending data to the database is 2,886s and 2,912s, with 0% packet loss. The correlation coefficients generated in the Levenberg Marquardt model training process are 0,925 and 0,965. A comparison of the prediction data with the actual sensor reading data shows that this model has a MSE rate of 8,96cm and 13,04cm.

*Keywords: green campus, waste management, IoT, Artificial Neural Network*

## 1. Pendahuluan

Prinsip pembangunan berkelanjutan kini menjadi model, strategi, dan acuan dalam segala aspek kehidupan berwawasan lingkungan. Melalui program kompetisi *Green Campus*, universitas membantu menjalankan misi konservasi lingkungan hidup. Secara umum, *Green*

*Campus* akan dinilai dan diperingkat menggunakan parameter yang berdasarkan pada pilar lingkungan, sosial dan ekonomi, meliputi penataan dan infrastruktur, energi dan perubahan iklim, pengelolaan limbah, penggunaan air, transportasi, dan pendidikan lingkungan [1].

Poin penilaian dalam kompetisi tersebut kini memengaruhi kebijakan yang diterapkan di universitas. Seperti yang tertuang di dalam Surat Edaran no. 27 Tahun 2019 tentang Pengelolaan Sampah di Universitas Diponegoro [2]. Namun, kebijakan tersebut hanya mencakup kegiatan pengolahan di hulu dan hilir saja. Padahal pada proses pengelolaan sampah terdiri dari rangkaian kegiatan yang berkaitan dengan penanganan, pengolahan, pembuangan atau daur ulang bahan limbah. Semuanya memiliki peranan yang penting dalam memastikan bahwa limbah yang berasal dari suatu lokasi dibawa keluar, diolah, dibuang dengan cara yang aman dan benar.

Rangkaian proses pengumpulan dan pengelolaan sampah dimulai dari tempat sampah yang diletakkan di beberapa titik. Fungsi tempat sampah sendiri sebagai wadah untuk mengumpulkan sampah yang diproduksi manusia, menjaga kebersihan dan keindahan lingkungan, serta mengelola lingkungan sehat dan higienis. Namun sering kali fungsi tersebut tidak terpenuhi akibat beberapa faktor membuat petugas kebersihan kesulitan mengerjakan tugasnya. Faktor tersebut seperti, tidak adanya kesadaran membuang sampah pada tempatnya, letak tempat sampah yang salah, tempat sampah terlalu penuh hingga sampah berceceran, tidak adanya sistem pemantauan keadaan tempat sampah secara aktual, atau bahkan peramalan ketinggian sampah di masa depan.

Cubillos, M. [3] menyatakan bahwa proyeksi yang akurat dari jumlah sampah penting untuk keberhasilan perencanaan sistem pengelolaan sampah yang efisien. Estimasi timbulan sampah di masa depan menjadi dasar dalam pengembangan berkelanjutan infrastruktur pengelolaan sampah. Prediksi yang tidak tepat dapat menyebabkan masalah yang meluas, seperti infrastruktur pembuangan limbah yang tidak memadai atau berlebihan [4]. Hal ini perlu dipertimbangkan sebab pengelolaan sampah yang baik, mulai dari sistem pengumpulan hingga pembuangan, mampu melindungi kesehatan manusia, lingkungan dan melestarikan sumber daya alam [5].

Berbagai penelitian telah memberikan solusi untuk mengurangi dampak luapan limbah dengan memanfaatkan berbagai teknologi. Terdapat penelitian yang mengembangkan IoT berbasis LoRa untuk monitoring tempat sampah[6]. Teknologi tong sampah dengan lengan mekanik yang akan mendeteksi keberadaan sampah di luar tempat sampah[7], bahkan truk pengangkut sampah berbasis *line follower* [8] telah dikembangkan sebagai upaya penanganan timbulan sampah.

Berdasarkan permasalahan di atas maka akan dibangun sebuah sistem yang dapat digunakan sebagai sarana pemantauan dan prediksi ketinggian sampah. Pemantauan akan memberi kemampuan visibilitas sembari mengumpulkan data historis, sedangkan peramalan memberikan dasar untuk implementasi, perbaikan, dan optimalisasi operasi pengelolaan sampah. Fitur peramalan

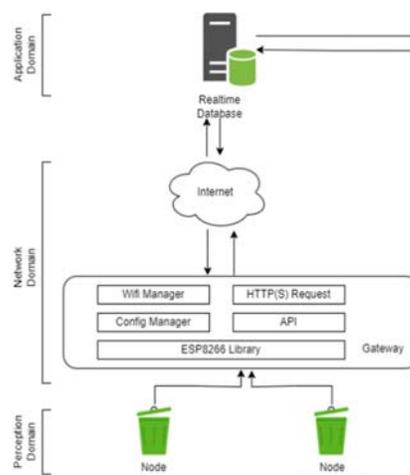
akan dikolaborasikan dalam konsep IoT sehingga didapatkan sebuah sistem *monitoring* muatan sampah secara *realtime*.

Tujuan penelitian ini adalah membangun komunikasi data berbasis IoT dan memodelkan data menggunakan Jaringan Saraf Tiruan (JST) untuk memprediksi ketinggian sampah. Fitur ini merupakan inisialisasi awal dari sistem keseluruhan sebelum ditampilkan kepada pengguna melalui aplikasi. Data masukan berupa ketinggian sampah yang ditangkap oleh sensor HC-SR04, kemudian dikirimkan oleh ESP8266 menuju Firebase untuk disimpan. Selanjutnya dilakukan proses pelatihan terhadap data menggunakan algoritma Levenberg-Marquardt untuk meramalkan ketinggian sampah 2 jam berikutnya. Sebelum sampah penuh dan tercecer, petugas kebersihan dan *stakeholder* dapat menjadikan data prediksi sebagai acuan dalam pengambilan keputusan, kebijakan, dan membantu mereduksi timbulan sampah.

## 2. Metode

### 2.1 Desain Sistem

Perancangan arsitektur sistem komunikasi data diadopsi dari standar IEEE SA-2413-2019 *Standard for an Architectural Framework for The Internet of Things* [9]. Standar tersebut membagi arsitektur umum IoT kedalam beberapa domain, yaitu *domain perception*, *domain network*, *domain application* seperti pada Gambar 1.



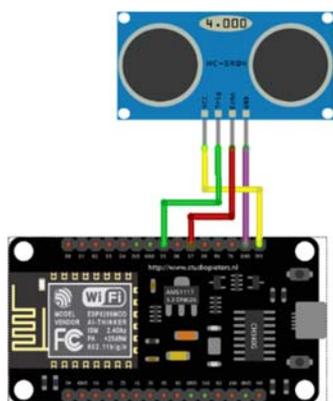
Gambar 1. Skema Jaringan Sistem Monitoring Ketinggian Sampah

*Domain perception* merupakan *layer* pengambilan data, yang mengubah besaran fisis menjadi sinyal listrik. Pada penelitian ini, digunakan sensor ultrasonik HC-SR04 sebagai masukan yang akan mendeteksi ketinggian sampah di tempat sampah. Perubahan protokol dilakukan oleh NodeMCU ESP8266 yang berperan sebagai IoT *Gateway* dalam merutekan data antara perangkat IoT dengan *cloud server*. ESP8266 memungkinkan pengguna terhubung ke

internet melalui jaringan Wifi dan mengirimkan data menuju Firebase *realtime database* dengan mengimplementasikan protokol HTTP/HTTPS. *Application domain* akan bertanggung jawab untuk melakukan pemrosesan data dan penyedia layanan, yang dalam hal ini data yang berasal dari *database* akan di tampilkan melalui antarmuka Android.

### 2.1 Perancangan Perangkat Keras

Komponen penyusun perangkat keras pada sistem ini adalah Sensor Ultrasonik HC-SR04 sebagai sumber informasi utama sebagai pemulai proses analisis informasi. NodeMCU ESP8266 berperan sebagai modul wifi untuk komunikasi data dan akan mengirimkan data secara berkala. Perancangan dapat dilihat pada Gambar 2.



Gambar 2. Wiring Perancangan Perangkat Keras

Kedua komponen tersebut diletakkan pada bagian tutup tempat sampah, berukuran 20L. Sistem ini menggunakan catu daya PLN sebagai *supply* listrik.

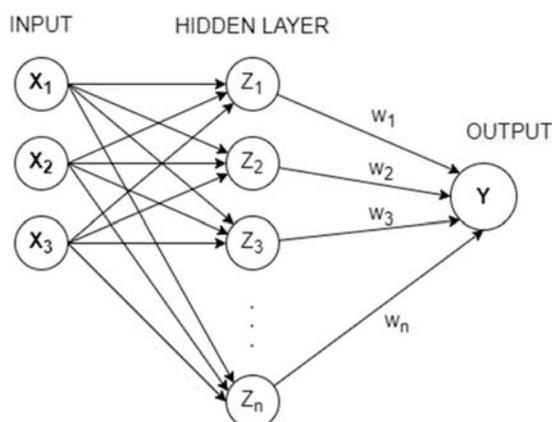
### 2.2 Perancangan JST

Sistem peramalan pada penelitian ini mengimplementasikan metode Jaringan Saraf Tiruan dengan algoritma Levenberg-Marquardt. Proses pelatihan dan pengujian dikerjakan menggunakan perangkat lunak Matlab. Kemudian konversi pelatihan menjadi bentuk persamaan matematis yang akan diimplementasikan dalam mikrokontroler. Diagram proses dijelaskan pada Gambar 3.



Gambar 3. Diagram Proses Perancangan model JST

Arsitektur metode algoritma Levenberg-Marquardt dapat dijabarkan seperti pada Gambar 4.



Gambar 4. Arsitektur Jaringan Levenberg-Marquardt untuk Prediksi Ketinggian Sampah

Arsitektur jaringan Levenberg-Marquardt di atas digunakan untuk memprediksi ketinggian sampah dalam waktu setelah 2 jam. Struktur tersebut dapat diuraikan sebagai berikut :

1. *Input* : 3 unit data yang disimbolkan dengan  $X_1$ ,  $X_2$ , dan  $X_3$  yang merepresentasikan data ketinggian sampah tiap 30 menit.
2. *Hidden layer* : disimbolkan dengan  $Z_1$ ,  $Z_2$ ,  $Z_3$ , ...  $Z_n$  berfungsi sebagai penghubung antara unit masukan dan unit keluaran dinyatakan dalam bobot dan fungsi aktivasi. Pada tugas akhir ini digunakan 3 neuron untuk *node* Dekanat dan 6 neuron untuk *node* Teknik Elektro.
3. *Output* : 1 unit data prediksi yang disimbolkan dengan  $Y$  hasil dari bobot *hidden layer* yang diteruskan menuju lapisan keluaran.

Berikut merupakan proses *training* Jaringan Saraf Tiruan dengan algoritma Levenberg Marquardt.

**Langkah 1** : Inisialisasi parameter (*epoch*, parameter marquardt, faktor tau, target MSE, neuron *hidden layer*).

**Langkah 2** : Perhitungan umpan maju terhadap unit masukan yang menerima sinyal *input*. Dilakukan dengan menjumlahkan input dengan bobot 1 dan bias 1 untuk diteruskan ke *hidden layer* dengan bantuan fungsi aktivasi.

**Langkah 3** : Tiap nilai *output* dijumlahkan dengan nilai bobot 2 dan bias 2. Kemudian keseluruhan unit dikirim ke lapisan atasnya dengan bantuan fungsi aktivasi.

**Langkah 4** : Mencari besar *error* dan MSE dari lapisan *output*. merupakan tahap akhir dari perhitungan umpan maju. Kemudian bandingkan nilai MSE dengan target *error*. Jika MSE masih lebih besar daripada target *error* yang ditetapkan di awal proses *training*, lanjut ke langkah 5.

**Langkah 5** : Permulaan tahap perhitungan umpan mundur. Menghitung informasi *error* tiap unit keluaran, kemudian hasilnya akan dikirim ke *hidden layer* menuju *output* untuk proses koreksi bias dan bobot dengan bantuan fungsi aktivasi.

**Langkah 6 :** Menghitung informasi *error* tiap unit *hidden layer*, kemudian hasilnya akan dikirim ke lapisan *input* menuju *hidden layer* untuk proses koreksi bias dan bobot disana.

**Langkah 7 :** Menyusun matriks Jacobian  $J(x)$  yang tersusun berdasarkan nilai koreksi bobot dan bias di seluruh jaringan. Matriks ini berguna untuk mendapatkan koreksi bobot dan bias baru di *hidden layer*, baik menuju *input* atau menuju *output*.

**Langkah 8 :** Mencari nilai bobot dan bias baru di unit *input* dan *ouput* mencari selisih koreksi bobot dan bias lama dengan yang baru.

**Langkah 9 :** Pelatihan akan berhenti jika  $error = target$  error atau jika kondisi sudah terpenuhi atau  $epoch \geq epoch$  maksimal

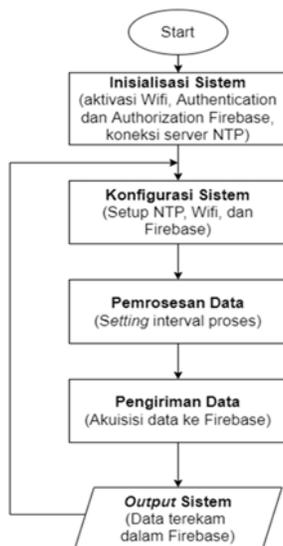
## 2.1 Perancangan Perangkat Lunak

### 2.1.1 Perancangan Program Sensor Ultrasonik HC-SR04

Pada perancangan program sensor ultrasonik ini dilakukan dengan menggunakan *software compiler* Arduino IDE berbasis bahasa C++. Perancangan perangkat lunak untuk sensor ultrasonik HC-SR04 terbagi menjadi tiga bagian yaitu: inisialisasi sistem, pemrosesan data, dan *output* sistem. Pada tahap inisialisasi merupakan pendeklarasian *library* dan *setup*, tahap pemrosesan data dilakukan dengan konversi ke satuan jarak (cm) dan bentuk persentase. Tahap *output* sistem, variabel akan ditampilkan pada serial monitor dan disimpan dalam *database*.

### 2.1.2 Perancangan Program Komunikasi Data NodeMCU ESP8266

Pada perancangan ini Gambar 5 merupakan flowchart dari perancangan.

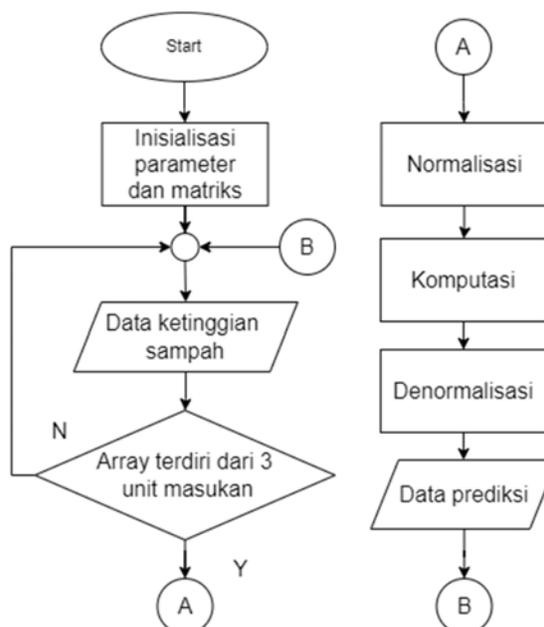


Gambar 5. Flowchart Perancangan Program Komunikasi Data

Bertujuan untuk membangun komunikasi antara ESP8266 yang berperan sebagai IoT *Gateway* dengan Firebase sebagai penyimpanan data. Gambar 5 menunjukkan *flowchart* atau diagram alir dari perancangan program NodeMCU ESP8266.

### 2.1.3 Perancangan Program JST

Dalam perancangan ini, pemrograman dilakukan dengan mengubah persamaan matematis yang didapatkan dari pelatihan software Matlab menjadi bahasa C supaya dapat dikomputasikan bersamaan dengan proses lain. Diagram alir pada Gambar 6 menunjukkan langkah perancangan program jaringan saraf tiruan dalam bahasa C.



Gambar 6. Flowchart Perancangan Program JST

Sehubungan dengan digunakannya satu jenis sensor pada sistem ini, maka harus dilakukan persiapan *dataset* untuk memulai komputasi Jaringan Saraf Tiruan. Hal ini dilakukan karena model ini disusun menggunakan tiga unit masukan untuk memprediksi keluaran pada waktu tertentu. Program akan bekerja dengan memanggil hasil bacaan sensor dengan interval 30 menit, kemudian diletakkan dalam sebuah *array*.

## 3 Hasil dan Pembahasan

### 3.1 Pengujian Komunikasi Data IoT

Merupakan pengujian terhadap sistem komunikasi antara perangkat IoT dengan *cloud server*. Tingkat keberhasilan pengujian pada tahap ini adalah ketika ESP8266, sebagai IoT *Gateway*, dapat mengunggah data ketinggian sampah hasil tangkapan sensor menuju realtime *database*.

### 3.1.1 Pengujian Pengiriman Data Serial

Pengujian ini dilakukan terhadap serial monitor untuk mengetahui proses dan tingkat keberhasilan pertukaran data antar perangkat. Gambar 7 berikut menampilkan serial monitor modul ESP8266.

```
17:12:34.908 -> Jarak: 0.00 cm
17:12:34.995 -> 0.00 %
17:12:34.995 -> Kapasitas sampah: 100.00 %
17:12:34.995 -> 11-8-2022
17:12:44.618 -> Jarak: 4.00 cm
17:12:44.618 -> 8.89 %
17:12:44.618 -> Kapasitas sampah: 91.11 %
17:12:44.665 -> 11-8-2022
17:12:55.839 -> Jarak: 3.00 cm
17:12:55.839 -> 6.67 %
17:12:55.839 -> Kapasitas sampah: 93.33 %
17:12:55.839 -> 11-8-2022
17:13:05.589 -> Jarak: 3.00 cm
```

Gambar 7. Tampilan Serial Monitor

Berdasarkan gambar di atas, maka mikrokontroler berhasil mengolah dan mengubah data dari pin masukan menuju pin keluaran.

### 2.1.1 Pengujian Konektivitas

Pengujian ini dilakukan terhadap kemampuan modul ESP8266 untuk terhubung dengan jaringan wifi maupun Firebase API. Pada bagian pertama akan dilakukan pengujian konektivitas jaringan wifi yang dikerjakan dengan memanfaatkan *library* `ESP8266wifi.h` yang sudah dideklarasikan dalam senarai program. Proses koneksi modul ESP8266 dengan jaringan wifi dapat dilihat pada serial monitor di Gambar 8 berikut.

```
09:56:31.161 -> .....
09:56:36.143 -> Node Teknik Elektro Connect Successfully to IP:
09:56:36.191 -> 10.109.8.53
09:56:36.643 -> [TIME] : Setting time using SNTP
09:56:37.151 ->
09:56:37.151 -> [TIME] : Current time: Sun Aug 28 02:56:37 2022
09:58:10.271 -> .....
09:58:17.952 -> Node Dekanat Lama Connect Successfully to IP:
09:58:17.999 -> 10.109.8.55
09:58:18.422 -> [TIME] : Setting time using SNTP
09:58:18.943 ->
09:58:18.943 -> [TIME] : Current time: Sun Aug 28 02:58:18 2022
```

Gambar 8. Tampilan Serial Monitor Koneksi Jaringan Wifi

Saat modul mencoba terhubung ke jaringan internet, akan muncul tampilan *connecting* yang disimbolkan dengan tanda baca titik (...). Pada serial monitor akan menampilkan alamat IP jika berhasil mendapatkan koneksi internet dari suatu jaringan diikuti dengan keterangan "Node Teknik Elektro/Dekanat Lama Connect Successfully to IP :".

Bagian kedua merupakan pengujian konektivitas dengan API Firebase dengan memanfaatkan *library* `Firebase_ESP_Client` yang sudah dideklarasikan di dalam senarai program. Pada Gambar 9 dibawah ini merupakan kondisi saat proses koneksi Firebase sedang berlangsung.

```
16:07:26.738 ->
16:07:27.711 -> Firebase Client v4.0.0
16:07:27.711 ->
16:07:28.735 -> Token info: type = id token, status = on request
16:07:43.809 -> Token info: type = id token, status = error
16:07:43.847 -> Token error: code: -4, message: connection lost
```

Gambar 9. Serial Monitor Saat Hilang Koneksi

Dapat diketahui bahwa proses *request* token ke server Firebase *error* dikarenakan tidak terdapat koneksi internet. Hal ini membuktikan bahwa proses pertukaran data tidak akan terjadi apabila tidak ada koneksi internet. Selanjutnya Firebase akan terus mencoba dan menyambung kembali jika koneksi terputus. Dan akan berhasil jika token *request* berhasil dan siap untuk digunakan, seperti yang ada pada Gambar 10.

```
16:07:44.310 -> Token info: type = id token, status = on request
16:07:59.991 -> Token info: type = id token, status = ready
```

Gambar 10. Serial Monitor Status Ready

### 2.1.2 Pengujian Unggah Data ke Firebase

Pengujian ini dilakukan terhadap tingkat keberhasilan unggah data ke Firebase realtime database dilihat dari kondisi *delay* dan *packet loss*. Pengujian dilakukan dengan mengirimkan paket data yang memuat hasil tangkapan sensor ultrasonik menuju *database*. Kualitas pengujian didasarkan pada keberhasilan data sensor untuk masuk ke Firebase dan serial monitor menampilkan pesan "Node Teknik elektro/Dekanat Lama berhasil mengirim data". Pengujian ini dilakukan sebanyak 5 percobaan untuk mengetahui konsistensi. Gambar 11 dan Gambar 12 merupakan tampilan serial monitor dan Firebase saat data berhasil diunggah.

```
12:28:07.589 -> Jarak: 7.00 cm
12:28:07.589 -> 15.56 %
12:28:07.589 -> Kapasitas sampah: 84.44 %
12:28:07.589 -> 28-8-2022
12:28:10.446 -> Node Dekanat Lama berhasil mengirim data
2022-8-28:12:28:07
  Jarak: "7"
  PersenIsi: "84.44444"
  Waktu: "28-8-2022/12:28:07"
```

Gambar 11. Serial Monitor dan Firebase Node Dekanat Lama

```
10:56:30.335 -> Jarak: 12.00 cm
10:56:30.335 -> 26.67 %
10:56:30.335 -> Kapasitas sampah: 73.33 %
10:56:30.383 -> 28-8-2022
10:56:33.352 -> Node Teknik Elektro berhasil mengirim data
node Teknik Elektro
  2022-8-28:10:26:30
  2022-8-28:10:56:30
    Jarak: "12"
    PersenIsi: "73.33334"
    Waktu: "28-8-2022/10:56:30"
```

Gambar 12. Serial Monitor Dan Firebase Node Teknik Elektro

Hasil pengujian ditampilkan pada Tabel 1.

**Tabel 1** Pengujian Unggah Data ke Firebase dari Sisi Delay

Percobaan Ke-	Node Dekanat Lama		Node Teknik Elektro	
	Dikirim	Diterima	Dikirim	Diterima
1.	10:28:07.5	10:28:10.4	10:26:30.2	10:26:33.1
	46	92	80	76
2.	10:58:07.5	10:58:10.5	10:56:30.3	10:56:33.3
	78	04	35	52
3.	11:28:07.5	11:28:10.4	11:26:30.4	11:26:33.2
	91	02	07	98
4.	11:58:07.5	11:58:10.4	11:56:30.4	11:56:33.3
	82	70	68	68
5.	12:28:07.5	12:28:10.4	12:26:30.5	12:26:33.4
	89	46	59	15

Dapat dilihat bahwa terdapat waktu tunda antara pengiriman data oleh ESP8266 melalui jaringan wifi hingga data masuk ke database. Jeda tersebut diketahui sebagai *delay* yang merupakan waktu paket dalam melakukan perjalanan [10].

$$delay = waktu\ paket\ dikirim - waktu\ paket\ diterima \quad (1)$$

Apabila dihitung menggunakan rumus *delay* pada Persamaan (3.1) maka dapat diketahui bahwa jaringan komunikasi data sudah baik dengan rata-rata *delay node* Dekanat Lama sebesar 2,912 sekon, dan *node* Teknik Elektro 2,886 sekon. *Delay* tersebut merupakan akibat dari penggunaan fungsi millis dan server NTP dalam program. Millis merupakan fungsi arduino yang mengembalikan nilai waktu dan mengerjakan tugas secara multitugas pada interval pengiriman. Di sisi lain, parameter waktu yang digunakan pada sistem ini berdasarkan server NTP pada interval akuisisi data. Hal tersebutlah yang menyebabkan terjadinya waktu tunda.

Selanjutnya dilakukan peninjauan dari segi *packet loss* pada proses unggah data ke Firebase. *Packet loss* merupakan sebuah kondisi dimana paket gagal ditransmisikan. Berikut merupakan persamaan untuk mengetahui besar *packet loss* [10].

$$packet\ loss = \frac{paket\ dikirim - paket\ diterima}{paket\ dikirim} \times 100\% \quad (2)$$

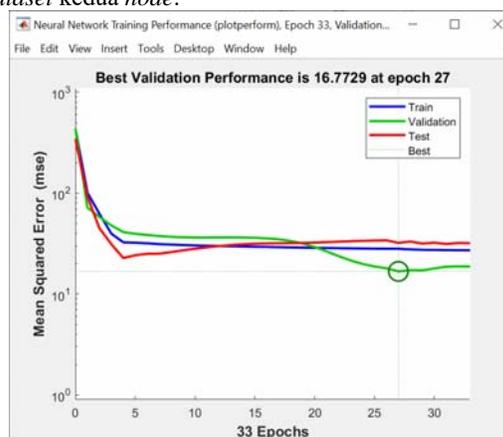
Berdasarkan percobaan pengujian unggah data ke Firebase yang dilakukan sebanyak 5 iterasi di semua node, dapat diketahui bahwa tidak terjadi *packet loss*.

### 2.1 Pengujian Sistem Peramalan Kapasitas Sampah

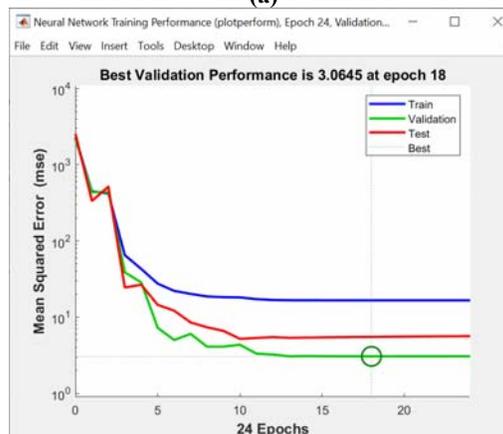
Merupakan pengujian terhadap sistem peramalan yang telah dimodelkan menggunakan Matlab dengan pengimplementasian Jaringan Saraf Tiruan algoritma Levenberg Marquardt. Tingkat keberhasilan pengujian ini ketika hasil prediksi menghasilkan akurasi dan nilai *error* terbaik untuk kedua *node* sehingga dapat digunakan sebagai fitur peramalan dalam aplikasi *monitoring* ketinggian sampah.

### 2.1.1 Pengujian Validasi Data Menggunakan Perangkat Lunak

Pengujian ini dilakukan setelah pelatihan terhadap model Jaringan Saraf Tiruan selesai. Pada Gambar 13 didapatkan kondisi terbaik dari pelatihan yang dilakukan terhadap dataset kedua *node*.



(a)



(b)

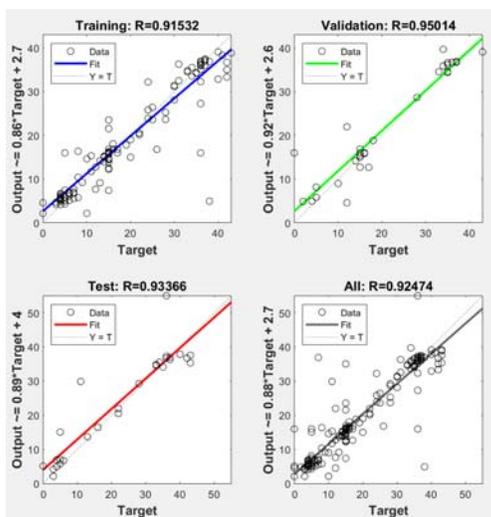
**Gambar 13.** Kurva Performansi Training untuk Node (a) Teknik Elektro (b) Dekanat Lama

Kurva di atas merupakan hasil performansi *training* terbaik untuk *node* Teknik Elektro dan Dekanat Lama. *Training* berhenti setelah performa terbaik diambil dari *epoch* dengan kesalahan validasi terendah. Dapat diketahui bahwa pada *node* Teknik Elektro data *training* terbaik terjadi pada *epoch* ke-27 dengan nilai MSE 16,7729. *Training* terbaik dari *node* Dekanat terjadi pada *epoch* ke-18 dengan MSE sebesar 3,0645.

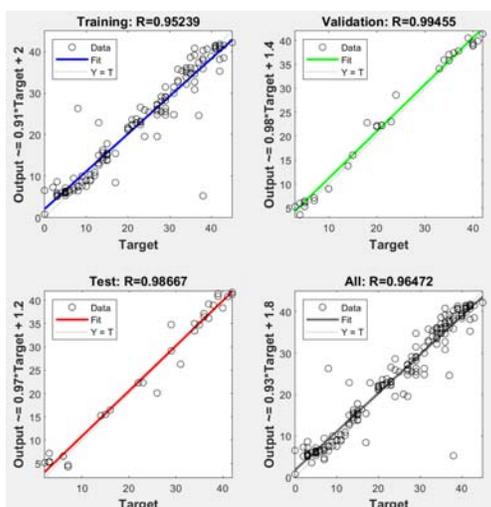
Kualitas hasil pembelajaran dibuktikan dengan plot regresi pada Matlab yang menampilkan hubungan fungsional antara target relatif terhadap *output*. Disajikan grafik untuk setiap hasil pelatihan, validasi dan pengujian dengan simbol lingkaran adalah data input yang telah digunakan

untuk pelatihan. Semakin jauh simbol lingkaran dengan garis regresi, semakin besar *error* yang didapatkan. Sumbu x pada grafik merupakan besaran target dan sumbu y adalah *output* aktual yang dihasilkan oleh jaringan. Adapun nilai *output* pada grafik dimodelkan menggunakan Persamaan (3).

$$output = R * target + bias \quad (3)$$



Gambar 14. Grafik Regresi *Node* Teknik Elektro



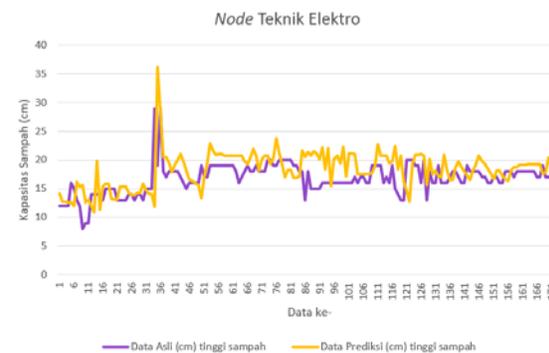
Gambar 15. Grafik Regresi *Node* Dekanat Lama

Dalam kedua *node*, diperoleh kualitas kecocokan yang baik pada kumpulan data. Pada grafik regresi *node* Teknik Elektro nilai R2 sejumlah 0.92474 dan 0,96472 untuk R2 *node* Dekanat Lama. Walaupun proses pelatihan di kedua *node* menggunakan jumlah data yang sama, namun hasil menunjukkan adanya perbedaan. Hal ini disebabkan oleh adanya pengaruh rentang data yang digunakan untuk melatih jaringan. Model prediksi bekerja dengan

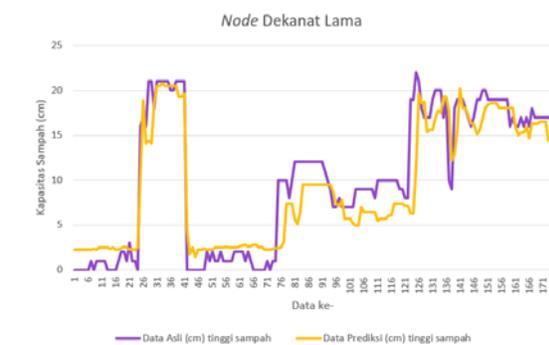
mempelajari pola data historis untuk meramalkan nilai di masa mendatang. Lebih lanjut, dalam menerapkan model JST *dataset* historis akan diproyeksi ke masa depan, sehingga pola data di masa lalu akan sangat berpengaruh pada hasil prediksi. Berdasarkan observasi, pada *node* Teknik Elektro rentang data yang didapatkan lebih bervariasi dibandingkan *node* Dekanat. Hal ini disebabkan oleh tingkat aktivitas yang terjadi disekitar lingkungan observasi. Hasil grafik regresi di masing-masing *node* terdapat pada Gambar 14 dan Gambar 15.

### 2.1.1 Pengujian Prediksi Ketinggian Sampah pada Sistem *Monitoring*

Pengujian ini dilakukan selama 4 hari dengan membandingkan data hasil tangkapan sensor sebenarnya dengan hasil prediksi. Tujuannya untuk mengetahui keakurasian dan performa model prediksi setelah diaplikasikan ke dalam sistem *monitoring*. Data hasil pengujian ditampilkan pada lampiran. Perbandingan ketinggian sampah di *node* Teknik Elektro digambarkan dalam grafik Gambar 16 dan *node* Dekanat Lama pada Gambar 17.



Gambar 16. Perbandingan Data Prediksi dan Data Sensor pada *Node* Teknik Elektro



Gambar 17. Perbandingan Data Prediksi dan Data Sensor pada *Node* Dekanat Lama

Pada Gambar 16 dan Gambar 17 terlihat bahwa perbandingan data prediksi dengan data aktual pada kedua node mempunyai pola yang hampir sama sehingga dapat ditarik kesimpulan bahwa peramalan dengan menggunakan metode Jaringan Saraf Tiruan sesuai dan dapat diterapkan dengan menggunakan data ketinggian sampah tiap 30 menit.

**Tabel 2 Perbandingan Data Prediksi dengan Data Sensor**

Node	Pembandingan	Data Sensor (cm)	Data Prediksi (cm)	Error (cm)	Error kuadrat (cm)
Teknik Elektro	Minimal	8	10,88	2,88	13,04
	Maksimal	29	36,19	7,19	
	Rata-rata	16,86	18,26	2,62	
Dekanat Lama	Minimal	0	1,46	1,46	8,96
	Maksimal	22	20,82	2,82	
	Rata-rata	10,17	9,29	2,17	

Dari Tabel 2 dapat disimpulkan bahwa rata-rata *error* yang dihasilkan pada pengujian *node* Teknik Elektro sebesar 2,62cm dan pada *node* Dekanat Lama sebesar 2,17cm. Tingkat rata-rata *error* kuadrat pada *node* Teknik Elektro adalah 13,04 cm dan pada *node* Dekanat Lama sebesar 8,96 cm. Hasil peramalan kapasitas sampah ini dapat digunakan sebagai acuan penentuan kebijakan dan keputusan bagi *stakeholder*, sehingga dapat membentuk program-program yang dapat mengurangi timbulan sampah secara berkala.

#### 4. Kesimpulan

Telah dibangun komunikasi data *realtime* berbasis IoT pada sistem *monitoring* kapasitas sampah memiliki kualitas layanan yang baik karena didukung oleh jaringan internet yang baik pula. Hal tersebut ditunjukkan dari hasil pengujian rata-rata delay pada node Teknik Elektro untuk mengirim data ke database yaitu sebesar 2,886 sekon dan node Dekanat Lama sebesar 2,912 sekon. Dan persentase keberhasilan pengiriman paket sebesar 100%. Sementara itu, model prediksi juga sudah berhasil dibangun melalui proses training dan testing untuk setiap node pemasangan. Node Teknik Elektro memiliki nilai koefisien korelasi sebesar 0,92474, sedangkan *Node* Dekanat Lama memiliki nilai koefisien korelasi sebesar 0,96472. Tingkat rata-rata *error* model prediksi saat digunakan pada sistem digunakan sebesar 2,62 cm pada *node* Teknik Elektro dan 2,17 cm untuk *node* Dekanat Lama. Serta tingkat rata-rata *error* kuadrat pada *node* Teknik Elektro adalah 13,04 cm dan untuk *node* Dekanat Lama sebesar 8,96 cm.

#### Referensi

- [1] B. Ridhosari, A. Rahman, Carbon footprint assessment at Universitas Pertamina from the scope of electricity, transportation, and waste generation: toward a green campus and promotion of environmental sustainability, *Journal of Cleaner Production* (2019), <https://doi.org/10.1016/j.jclepro.2019.119172>
- [2] Team UI GreenMetric Ranking. 2016. Guideline of UI GreenMetric World University Ranking 2016. [Online]. Tersedia: <http://greenmetric.ui.ac.id/>
- [3] M. Cubillos, Multi-site household waste generation forecasting using a deep learning approach. *Waste Management*. Volume 115. 2020. Pages 8-14. ISSN 0956-053X. 10.1016/j.wasman.2020.06.046.
- [4] M. Abbasi, & A. El Hanandeh, (2016). Forecasting municipal solid waste generation using artificial intelligence modelling approaches. *Waste Management*, 56, 13–22. 10.1016/j.wasman.2016.05.018.
- [5] L. Giusti, (2009). A Review of Waste Management Practices and Their Impact on Human Health. *Waste Management* (New York, N.Y.). 29. 2227-39. 10.1016/j.wasman.2009.03.028.
- [6] A. S. Bharadwaj, R. Rego and A. Chowdhury, "IoT based solid waste management system: A conceptual approach with an architectural solution as a smart city application," 2016 IEEE Annual India Conference (INDICON), 2016, pp. 1-6, doi: 10.1109/INDICON.2016.7839147
- [7] T. S. Vasagade, S. S. Tamboli and A. D. Shinde, "Dynamic solid waste collection and management system based on sensors, elevator and GSM," 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT), 2017, pp. 263-267, doi: 10.1109/ICICCT.2017.7975200.
- [8] A. A. Jaid Jim, et al, "A Noble Proposal for Internet of Garbage Bins (IoGB)," *Smart Cities*, vol. 2, no. 2, pp. 214–229, Jun. 2019, doi: 10.3390/smartcities2020014. [Online]. Tersedia : <https://www.mdpi.com/2624-6511/2/2/14>. (Diakses 29 Juli 2022).
- [9] IEEE Standard for an Architectural Framework for the Internet of Things (IoT). (n.d.). doi:10.1109/ieeestd.2020.9032420. 2020
- [10] W. Sugeng, J. E. Istiyanto, K. Mustofa, dan A. Ashari, "The Impact of QoS Changes Towards Network Performance," *Int. J. Comput. Networks Commun. Secur.*, vol. 3, no. 2, pp. 48–53, 2015, [Online]. Tersedia:[http://www.ijcnscs.org/published/volume3/issue2/p5\\_3-2.pdf](http://www.ijcnscs.org/published/volume3/issue2/p5_3-2.pdf).