

APLIKASI WIRELESS SENSOR NETWORK UNTUK SISTEM MONITORING DAN KLASIFIKASI KUALITAS UDARA

Tri Fidrian Arya, Mahar Faiqurahman, dan Yufis Azhar

Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Malang (UMM)
Jln. Raya Tlogomas No. 246, Malang, 65144, Indonesia

E-mail: trifidrianarya@gmail.com, mahar@umm.ac.id, yufis@umm.ac.id

Abstract

Indonesia is one of the countries investing in the industrial sector. Industries affect the environment, notably air quality. If the air pollution expelled by industrial chimneys could not be processed properly, it would adversely affect human health. Currently, monitoring the air quality in some areas only uses one tool. This is deemed inadequate to describe the condition of air quality in a given area. On the other hand, installing more than one tool would be very expensive. This research applied the concept of wireless sensor network (WSN) for air quality monitoring by installing more than one sensor nodes in a particular location and one sink that acts to collect data from the sensor nodes then sends them to the server. Air quality data obtained by the sensor nodes were then classified using the classification method in data mining that is *k*-nearest neighbor (K-NN). Prior to classification using K-NN, data normalization was performed, which generates a decimal scaling with good performance for air quality data. The *k* values used for the K-NN classification are 5. The accuracy of the system is 94.28%, the precision is 85.16% and the recall is 93.35%.

Keywords: *Wireless Sensor Network, Data Mining, Classification, K-NN, Air Quality.*

Abstrak

Indonesia merupakan salah satu negara yang bergerak di sektor industri. Perindustrian memberikan dampak pada lingkungan hidup, termasuk kualitas udara. Jika polusi udara yang dikeluarkan dari cerobong asap kawasan industri tidak diproses dengan baik, polusi tersebut akan berdampak buruk pada kesehatan manusia. Saat ini, pemantauan kualitas udara terhadap suatu cakupan lokasi tertentu hanya menggunakan satu alat saja. Hal ini dirasa kurang cukup untuk menggambarkan kondisi kualitas udara yang ada pada suatu cakupan lokasi tersebut. Di sisi lain, instalasi lebih dari satu alat akan membutuhkan biaya yang besar. Pada penelitian ini, diaplikasikan konsep *wireless sensor network* (WSN) untuk pemantauan kualitas udara dengan pemasangan lebih dari satu perangkat *node* sensor pada lokasi tertentu dan satu *sink* yang bertindak untuk mengumpulkan data dari *node* sensor lalu mengirimkannya ke *server*. Data kualitas udara yang didapatkan oleh *node* sensor kemudian diklasifikasikan menggunakan metode klasifikasi pada data mining, yaitu *k*-nearest neighbor (K-NN). Sebelum dilakukan klasifikasi menggunakan K-NN, dilakukan normalisasi data yang menghasilkan *decimal scaling* dengan performa yang baik untuk data kualitas udara. Nilai *k* yang digunakan untuk klasifikasi K-NN yaitu 5. Didapatkan tingkat akurasi yang dihasilkan oleh sistem sebesar 94,28%, presisi sebesar 85,16%, dan *recall* sebesar 93,35%.

Kata Kunci: *Wireless Sensor Network, Data Mining, Klasifikasi, K-NN, Kualitas Udara.*

1. Pendahuluan

Indonesia merupakan salah satu negara yang memiliki perkembangan yang pesat pada sektor industri, hal tersebut tentunya berpengaruh pada lingkungan hidup termasuk kualitas udara [1]. Menurut Supraptini [2], polusi udara yang dikeluarkan dari cerobong asap kawasan industri apabila tidak dikelola dengan baik akan berdampak buruk pada kesehatan manusia, di antaranya dapat berpengaruh terhadap status faal paru-paru, perubahan respon kekebalan tubuh, bahkan

menyebabkan kematian pada makhluk hidup. Telah dijelaskan dalam peraturan pemerintah pentingnya udara bagi kehidupan, sehingga perlu dilestarikan serta dijaga fungsinya melalui pengendalian pencemaran udara [3].

Oleh karena itu, *monitoring* tingkat polusi udara menjadi suatu hal yang *urgent* dilakukan. Di Indonesia, *monitoring* kualitas udara saat ini hanya menggunakan satu alat saja dan dalam cakupan lokasi tertentu, sehingga kurang dapat menggambarkan kondisi kualitas udara yang ada. Diperlukan lebih dari satu alat untuk secara efektif

melakukan pemantauan dalam cakupan lokasi tersebut. Untuk menanggulangi hal ini, dapat mengaplikasikan *Wireless Sensor Network* (WSN). WSN adalah teknologi yang melakukan proses *sensing*, kontrol, dan komunikasi untuk *monitoring* lingkungan secara pengukuran fisik [4]. WSN terdiri dari beberapa *node sensor*, serta *base station* yang diimplementasikan dan dapat saling mengirimkan data menggunakan jaringan nirkabel. Setiap *node sensor* menggunakan modul jaringan nirkabel dalam komunikasinya dan mudah diterapkan pada lingkungan [5].

Terdapat beberapa penelitian yang telah mengkaji pengaplikasian *Wireless Sensor Network* (WSN). Pada penelitian yang telah dilakukan oleh Iwan et al [6], sistem *monitoring* kualitas udara dibangun menggunakan WSN, dengan parameter yang digunakan yaitu suhu, kelembaban, karbondioksida, dan karbonmonoksida. Pada penelitian ini [6], data yang didapatkan tidak dilakukan pengolahan secara lebih lanjut untuk menghasilkan informasi klasifikasi kualitas udara. Penelitian yang dilakukan oleh Xu dan Liu [7], yaitu melakukan *monitoring* kualitas air dengan menggunakan parameter kelarutan oksigen, tekanan air, PH, dan temperatur dengan mengaplikasikan WSN. Penelitian aplikasi WSN dengan judul “*Wireless Sensor Network System Design using Raspberry Pi and Arduino for Environmental Monitoring Applications*” [8] mengaplikasikan modul arduino dan Xbee untuk komunikasi nirkabel, hanya saja modul Xbee tersebut memiliki biaya yang cukup tinggi dalam pengadaannya.

Penelitian lain yang mengaplikasikan *wireless sensor network* dan menerapkan konsep klasifikasi *data mining* di antaranya, Laksono et al [9] mengaplikasikan *wireless sensor network* untuk membuat prediksi cuaca dengan menggunakan parameter sensor angin, suhu, dan kelembaban, dengan menggunakan metode C4.5. Pada penelitian Sugiarto dan Sustika [10], dilakukan pengklasifikasian data kualitas udara dengan fitur yang digunakan yaitu suhu, kelembaban, karbondioksida, dan karbon monoksida, kemudian pengklasifikasian dilakukan menggunakan metode *decision tree*. Saoudi et al [11], melakukan penelitian mengaplikasikan WSN untuk mendeteksi kebakaran hutan. *Node sensor* pada penelitian ini [11] melakukan pemantauan suhu, kelembaban, intensitas cahaya, dan intensitas asap, kemudian dilakukan pengklasifikasian data pada *server* untuk mengetahui apakah terjadi kebakaran hutan atau tidak dengan menggunakan *Artificial Neural Network* pada *data mining*.

Pada penelitian ini dibuat aplikasi berbasis *Wireless Sensor Network* (WSN) untuk *monitoring* dan klasifikasi kualitas udara secara *online*, dengan

menggunakan modul LoRa untuk komunikasi antara *node sensor* dan *base station* yang biayanya cukup terjangkau dan jarak komunikasi yang relatif besar pula dibandingkan beberapa modul komunikasi *wireless* yang lain seperti Xbee. Penyebaran *node sensor* dilakukan pada titik-titik lokasi tertentu, misalnya di dekat jalan raya, perumahan, kawasan industri, dan sebagainya dengan parameter yang digunakan di antaranya karbonmonoksida, ozon, dan kadar partikulat di udara. Nilai dari parameter tersebut didapatkan dari proses *sensing* oleh sensor secara berkala.

2. Metode

Dalam penelitian ini akan diaplikasikan konsep *wireless sensor network* untuk sistem *monitoring* dan klasifikasi kualitas udara, di mana perangkat yang dibangun dan digunakan untuk pemantauan tingkat kadar karbonmonoksida (CO), ozon (O₃), dan partikulat (PM10) di udara. Kemudian dari hasil pemantauan (*sensing*) tersebut akan dilakukan klasifikasi kualitas udara lingkungan, apakah baik, sedang, atau tidak sehat. Untuk proses klasifikasi digunakan salah satu teknik yang ada di *data mining* yaitu metode *K-Nearest Neighbor*.

Analisis Data

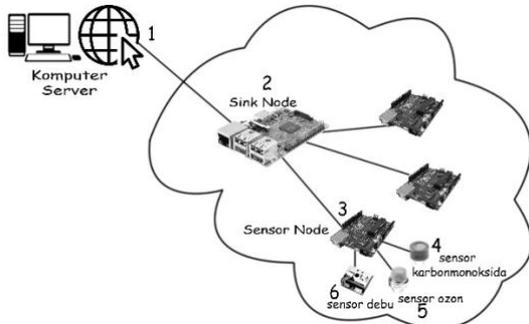
Data yang digunakan dalam penelitian ini merupakan data nilai dari parameter karbon monoksida (CO), ozon (O₃), dan partikulat udara (PM10). Data tersebut digunakan menjadi data latih untuk proses klasifikasi nantinya. Tiga parameter tersebut sesuai dengan keputusan kepala bapedal indek standar pencemaran udara (ISPU) [12]. Data pada penelitian ini didapatkan dari *website* dinas lingkungan hidup dari beberapa daerah di Indonesia. Pendekatan yang dapat digunakan untuk proses klasifikasi digunakan algoritma *K-Nearest Neighbor* (K-NN). Algoritma K-NN dipilih karena data tingkat kadar CO, O₃, dan PM10 yang dipantau berupa angka, dan metode K-NN dalam proses klasifikasi data menggunakan kedekatan jarak antara data uji dengan setiap data yang terdapat pada data latih. Berikut beberapa contoh format data kualitas udara pada Tabel 1.

Data ke	CO (PPM)	O ₃ (PPM)	PM10 (PPM)	Label
1	0,692692	8,175989	45,05	tidak sehat
2	0,83508	8,175989	67,78	tidak sehat
3	0,499352	8,138137	48,36	tidak sehat
4	0,947176	12,58099	29,45	tidak

				sehat
5	0,703906	12,08419	43,7	tidak
6	0,591964	9,898246	51,01	sehat
7	0,88617	9,794153	55,21	sedang
8	0,40288	8,086091	89,22	sedang
9	0,680368	13,57933	32,25	sedang
10	0,520717	3,345153	33,3	sedang
11	1,121146	4,807179	65,4	baik
12	0,373837	4,168429	10,29	baik
13	0,800799	1,47622	17,21	baik
14	0,780917	2,408321	15,86	baik
15	6,600995	99,42201	192,516	baik

Desain Arsitektur Sistem

Pada penelitian ini arsitektur *wireless sensor network* (WSN) yang digunakan terdiri dari *sink node*, *node sensor*, dan komputer *server*. Topologi yang digunakan pada sistem WSN penelitian ini yaitu topologi *star* yang mana setiap *node sensor* langsung terhubung dengan *sink*. Pada Gambar 1 merupakan arsitektur WSN yg diimplementasikan.



Gambar 1. Arsitektur sistem WSN yang diaplikasikan

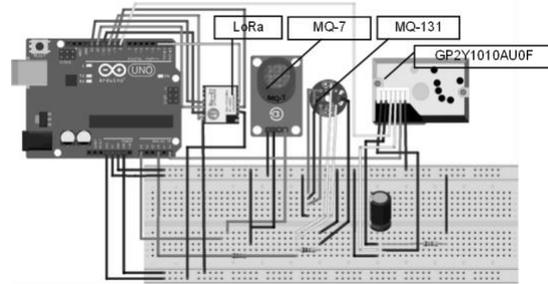
Keterangan :

1. Komputer *server* yang berfungsi untuk *monitoring* dan klasifikasi kualitas udara
2. *Sink* merupakan mini komputer menggunakan perangkat Raspberry Pi 3 Model B dan ditambah Arduino.
3. *Node sensor* merupakan perangkat mikrokontroler menggunakan Arduino Uno.
4. Sensor MQ-7 yang digunakan untuk mengukur tingkat gas karbonmonoksida.
5. Sensor MQ-131 yang digunakan untuk mengukur tingkat gas ozon (O₃).
6. Sensor GP2Y1010AU0F yang digunakan untuk mengukur tingkat partikulat debu diudara.

Error! Reference source not found.Gambar 1 merupakan arsitektur WSN yang diimplementasikan terdiri dari komponen *node sensor*, *sink node*, dan komputer *server*. *Node sensor* yang berfungsi melakukan pengukuran lingkungan terkait kualitas udara. Hasil pengukuran *node sensor* dikirimkan ke *sink*. *Sink node* bertindak untuk mengumpulkan data hasil *sensing* oleh *node sensor* untuk sementara waktu. Setelah data dikumpulkan di *sink*, data tersebut akan dikirimkan ke komputer *server* untuk diolah menghasilkan klasifikasi dan menampilkan kualitas udara.

Node Sensor

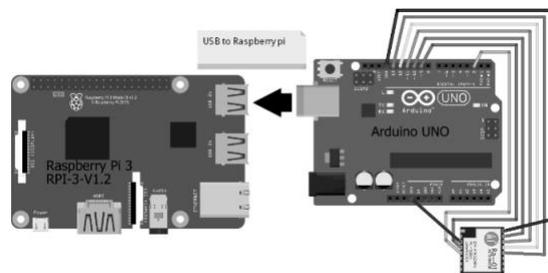
Komponen yang digunakan pada *node sensor* yaitu papan arduino untuk menyimpan kode sumber program, sensor MQ-7 untuk melakukan sensing CO, sensor MQ-131 untuk melakukan sensing O₃, sensor sharp GP2Y1010AU0F untuk melakukan sensing PM10, dan modul LoRa digunakan untuk berkomunikasi dengan *sink*. Gambar 2 merupakan rancangan rangkaian pada *node sensor*.



Gambar 2. Rancangan rangkaian pada *node sensor*

Sink

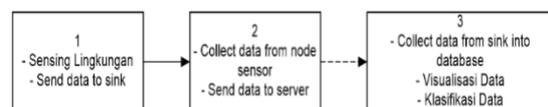
Komponen yang digunakan pada *sink* yaitu papan Raspberry Pi 3 model B yang telah dilengkapi modul wifi pada *boardnya*. Modul wifi tersebut akan digunakan untuk berkomunikasi dengan komputer *server* guna mengirimkan data kualitas udara. Kemudian pada *sink* terdapat Arduino Uno *board* yang dilengkapi dengan modul LoRa terhubung pada port USB Raspberry Pi yang digunakan untuk berkomunikasi guna menerima data dari *node sensor*.



Gambar 3. Rancangan rangkaian pada *sink*

Perancangan Proses

Pada tahapan ini dilakukan perancangan proses untuk setiap komponen yang terdapat pada arsitektur *wireless sensor network* (WSN) yang diaplikasikan pada *node sensor*, *sink*, dan *server* sesuai fungsinya masing-masing. Gambar 4 merupakan gambaran perancangan proses.



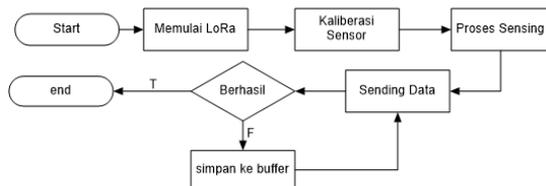
Gambar 4. Rancang rangkaian pada *sink*

Keterangan :

1. Perangkat proses pada *node* sensor (Mikrokontroler).
 2. Perangkat proses pada *sink*.
 3. Perangkat proses pada *server*.
- Jaringan Radio Frekuensi (LoRa).
 → HTTP Request.

Node Sensor

Perangkat lunak pada *node sensor* digunakan untuk melakukan proses *sensing* kondisi udara dengan mengambil nilai dari parameter CO, O₃, dan PM10. Gambar 5 merupakan diagram alur program pada *node sensor* mikrokontroler.

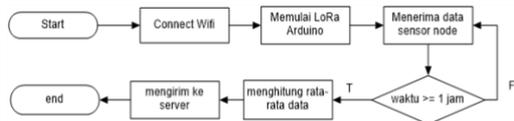


Gambar 5. Diagram alur program *node* sensor

Gambar 5 di atas merupakan proses intruksi program pada *node* sensor yang melakukan proses pengukuran nilai CO, O₃, dan PM10. Proses *sensing* dilakukan selama 1 menit dengan data yang didapatkan ± 40 data, kemudian dihitung nilai rata-ratanya. Nilai dari rata-rata yang didapatkan selama 1 menit tersebut kemudian dikirimkan ke *sink*. Apabila pengiriman yang dilakukan tidak berhasil, data yang tidak dikirimkan tersebut akan disimpan ke dalam *buffer* untuk sementara waktu, dan akan dikirimkan saat proses pengiriman selanjutnya.

Sink

Perangkat lunak pada *sink* digunakan untuk mengumpulkan data dari *node* sensor, kemudian dikirim ke *server*. Gambar 6 merupakan diagram alur perangkat lunak pada *sink*.



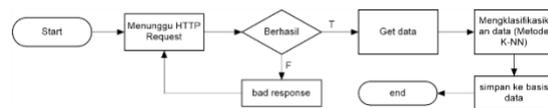
Gambar 6. Diagram alur perangkat lunak pada *sink*

Dari Gambar 6 di atas, *sink* akan menerima data yang dikirimkan oleh *node* sensor, kemudian akan dicatat waktu saat pertama kali menerima data dari *node* sensor. Proses pengiriman data ke *server* dilakukan setiap 1 jam dari saat menerima data pertama kali dari *node* sensor, ini dilakukan sesuai aturan Keputusan Kepala Bapedal No. 107 Tahun 1997. *Node* sensor mengirimkan data ke *sink* setiap 1 menit, maka pada *sink* akan terdapat 60 data tingkat kadar CO, O₃, dan PM10 sebelum data dikirimkan ke *server*. Dari 60 data tersebut

dihitung nilai rata-ratanya, kemudian dikirimkan ke *server*. Pengiriman data ke *server* menggunakan protokol HTTP.

Server

Perangkat lunak pada *server* menerima data kualitas udara dari *sink*. Setelah menerima data dari *sink*, server akan melakukan klasifikasi data yang diterima, lalu memasukan data dan hasil klasifikasinya ke dalam basis data yang tersedia di *server*. Komputer *server* akan menampilkan visualisasi kualitas udara. Gambar 7 adalah alur program dari penerimaan data dari *sink* menggunakan protokol HTTP yang akan diakses oleh *server*.

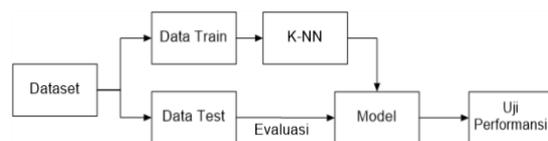


Gambar 7. Diagram alur penerimaan data pada *server*

Berdasarkan Gambar 7 di atas, *server* akan menunggu pesan HTTP yang dikirimkan oleh *sink* yang mengirimkan data kualitas udara yaitu tingkat kadar CO, O₃, PM10. Ketika data diterima oleh *server*, dilakukan pengolahan klasifikasi kualitas udara dengan menggunakan metode K-NN. Setelah dihasilkan label dari data yang dikirimkan oleh *sink*, label disimpan ke dalam basis data yang terdapat pada server.

Pada proses pengklasifikasian data di *server* menggunakan konsep *data mining*, yaitu menggunakan metode *K-Nearest Neighbor* (K-NN). Tahapan normalisasi data dilakukan berupa penyamaan skala atau jarak pada setiap fitur yang ada pada data latih dan data yang akan diklasifikasikan, dikarenakan data yang diterima oleh *server* memiliki skala dari setiap fitur beragam.

Standar jarak (normalisasi) yang digunakan yaitu *z-score*, *min-max*, dan *decimal scaling*. Dari 3 normalisasi tersebut akan dicari manakah yang memiliki tingkat performa untuk klasifikasi data yang baik. **Error! Reference source not found.** merupakan alur pengklasifikasian menggunakan *data mining*.



Gambar 8. Alur klasifikasi pada *data mining*

Algoritma dalam K-NN seperti di bawah ini.

- a. Memasukkan nilai k (jumlah data latih yang terdekat).
- b. Melakukan normalisasi data untuk penyamaan skala dari setiap parameter data.

Penelitian ini menggunakan 3 variasi normalisasi, yaitu:

-Z-score, formula yang digunakan :

$$v' = \frac{v - \mu}{\sigma}$$

Di mana, v' = nilai standarisasi; v = nilai dari atribut; μ = rata-rata; σ = standar deviasi.

-Min-Max, formula dari normalisasi min-max:

$$v' = \frac{v - v_{min}}{v_{max} - v_{min}}$$

-Decimal Scaling, formula decimal scaling:

$$v' = \frac{v}{10^j}$$

Di mana j = nilai *scaling* yang diinginkan

- c. Menghitung jarak terdekat antara data yang akan diklasifikasikan dengan setiap data yang ada pada data latih dengan menggunakan pengukuran *euclidean distance*.

Persamaan *euclidean distance* :

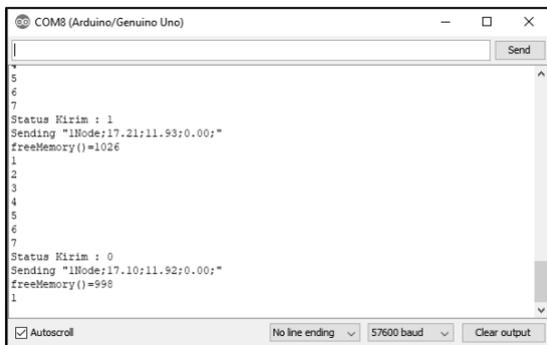
$$D(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

- d. Melakukan pengurutan data yang terdekat dari nilai hasil perhitungan jarak terkecil.
- e. Memilih label dengan frekuensi terbanyak.

3. Hasil dan Analisis

Implementasi Node Sensor

Gambar 9 merupakan *print screen* perangkat lunak pada *node* sensor saat melakukan proses *sensing* dan pengiriman data ke *sink*. Dari gambar di bawah ini dapat dilihat apabila *node* sensor dapat mengirimkan data ke *sink* maka status pengirimannya 1 dan jika tidak maka 0. Selain status pengiriman, *node* sensor akan menampilkan data yang dikirimkan oleh *node* sensor ke *sink*.



Gambar 9. Visualisasi pengiriman data ke *sink*

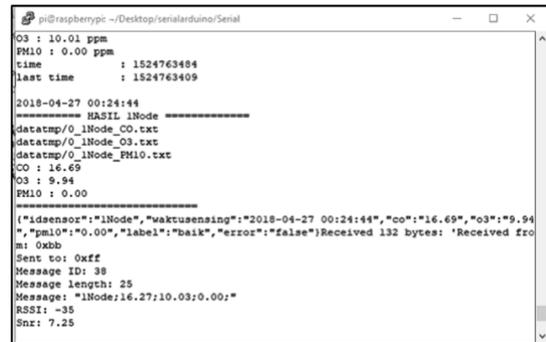
Implementasi Sink

Di bawah ini Gambar 10 merupakan *print screen* perangkat lunak yang terdapat pada *sink*. *Sink* akan menampilkan data yang diterima dari *node* sensor.

Gambar 11 merupakan visualisasi program saat *sink* melakukan pengiriman data ke *server*. Data yang dikirimkan yaitu ID sensor, waktu *sensing*, dan nilai dari paramter kualitas udara. Setelah data berhasil dikirimkan ke *server*, *server* akan memberikan respon JSON data dari *sink*.



Gambar 10. Visualisasi penerimaan data dari *node* sensor di *sink*



Gambar 11. Visualisasi pengiriman data dari *sink* ke *server*

Implementasi Server

Gambar 12 merupakan visualiasi data pemantauan kualitas udara dengan parameter nilai karbonmonoksida (CO), ozon (O₃), partikulat debu (PM10), dan hasil klasifikasi kualitas udaranya yang merupakan hasil pemantauan yang didapatkan oleh *node* sensor.



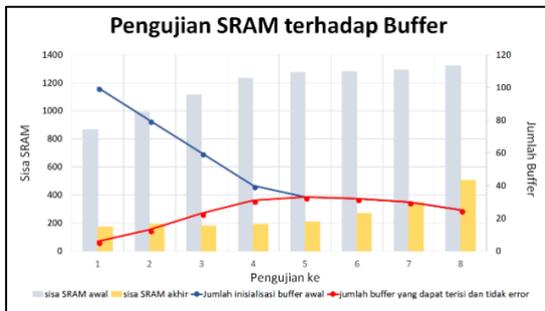
Gambar 12. Visualisasi data pada *server*

Pengujian Penggunaan SRAM dan Flash Memory Sensor Node

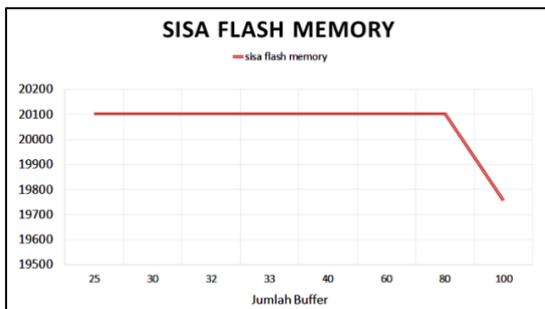
Pengujian ini dilakukan dengan skenario, untuk mengetahui sisa penggunaan SRAM dan flash memory pada arduino (node sensor) terhadap variasi jumlah array buffer yang digunakan untuk menyimpan data ketika node sensor tidak dapat menjangkau atau mengirimkan data ke sink. Pengujian ini dilakukan untuk memaksimalkan penggunaan resource pada arduino. SRAM pada arduino UNO sebesar 2K byte dan flash memory sebesar 32K byte.

TABEL II
PENGUNAAN SRAM DAN FLASH MEMORY

Jumlah inisialisasi buffer awal	sisa SRAM awal (byte)	sisa flash memory (byte)	jumlah buffer yang dapat terisi dan tidak error	sisa SRAM akhir (byte)	keterangan
100	864	19756	6	172	tidak sesuai
80	992	20102	13	192	tidak sesuai
60	1112	20102	23	181	tidak sesuai
40	1232	20102	31	192	tidak sesuai
33	1274	20102	33	208	sesuai
32	1280	20102	32	272	sesuai
30	1292	20102	30	346	kurang memaksimalkan resource
25	1322	20102	25	504	kurang memaksimalkan resources



(a)



(b)

Gambar 13. Penggunaan SRAM dan flash memory

- (a) Grafik sisa SRAM pada arduino berbanding dengan variasi jumlah buffer;
- (b) Grafik sisa flash memory pada arduino berbanding dengan variasi jumlah array buffer

Dari Tabel II dan Gambar 13 di atas dapat dilihat penggunaan variasi jumlah buffer berpengaruh pada sisa SRAM dan flash memory pada arduino. Apabila sisa dari SRAM arduino kurang dari 200 byte, arduino akan mengalami kegagalan sistem (system error). Jumlah buffer yang sesuai untuk memaksimalkan sumber daya SRAM pada arduino yaitu sekitar 33 dan 32 jumlah buffer.

Pengujian Jarak Komunikasi LoRa (SX1278)

Pengujian ini dilakukan untuk mengetahui jangkauan modul LoRa untuk melakukan pengiriman data dari node sensor ke sink. Pengujian ini dilakukan dengan skenario dengan variasi jarak antara node sensor dan sink, dan buffer array yang terisi, jumlah buffer array yang diinisialisasi berjumlah 33 (d disesuaikan dengan pengujian sebelumnya pada Tabel II Penggunaan SRAM dan flash memory).

TABEL III
PENGUKURAN JARAK LORA (SX1278)

Pengujian	Jarak (m)	Maksimal Buffer Terisi	keterangan
1	30	0	stabil
2	60	0	stabil
3	90	0	stabil
4	150	0	stabil
5	200	0	stabil
6	220	10	tidak stabil
7	240	33	gagal terkirim

Pada Tabel III di atas dapat dilihat pada jarak kurang dari 200 meter, pengiriman data dari node sensor ke sink stabil. Kemudian pengujian dari jarak 200 sampai kurang dari 220 meter pengiriman data tidak stabil. Kemudian pada saat jarak lebih dari atau sama dengan 240 meter, pengiriman data tidak dapat dilakukan.

Pengujian Kehilangan Data (Missing Data Handling)

Pengujian ini dilakukan untuk menangani data saat sink tidak dapat dijangkau atau diakses oleh node sensor, yang dapat diakibatkan karena ketidakstabilan jaringan antara node sensor dan sink menggunakan modul LoRa. Skenario pengujiannya, node sensor akan melakukan proses sensing sampai 1 menit (dalam penelitian ini waktu sensing 1 menit). Kemudian jumlah data yang

didapatkan selama 1 menit tersebut akan dicari nilai rata-ratanya (hal ini dilakukan untuk memaksimalkan pengolahan intruksi pada arduino), kemudian dikirimkan ke *sink*. Apabila *sink* tidak dapat diakses oleh *node* sensor, data tersebut akan disimpan ke dalam *buffer*. Hal tersebut dimaksudkan agar data tidak hilang saat *sink* tidak dapat diakses oleh *node sensor*. Jumlah *array* yang digunakan sejumlah 33 *buffer* (d disesuaikan dengan pengujian sebelumnya Tabel II Penggunaan SRAM dan *flash memory*). Apabila data yang terdapat pada *buffer array* telah mencapai 33 data dan data *sensing* oleh *node* sensor selanjutnya dihasilkan saat *sink* masih tidak dapat diakses, data pada *array buffer* yang lama akan diganti dengan data *sensing* yang baru. Saat *node* sensor dapat mengakses *sink*, maka semua *array* pada *buffer* akan langsung dikirimkan oleh *node* sensor ke *sink* secara satu persatu. Pengujian fungsi dari *buffer* dapat dilihat pada Tabel III Pengukuran jarak LoRa (SX1278).

Pengujian nilai k untuk klasifikasi K-NN

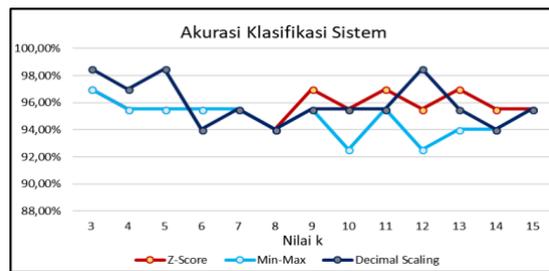
Pengujian nilai k pada K-NN ini dilakukan untuk mengetahui nilai k yang sesuai dengan tingkat akurasi yang tinggi. Pengujian nilai k dilakukan dengan skenario dengan variasi nilai k pada metode K-NN dengan 3 jenis normalisasi. Tabel IV merupakan pengujian nilai k klasifikasi K-NN pada sistem.

Dari Tabel IV Nilai k untuk K-NN dengan variasi normalisasi dan Gambar 14 Akurasi klasifikasi sistem, didapatkan beberapa nilai k untuk normalisasi *z-score* memiliki tingkat akurasi yang tinggi yaitu 3, 9, 11, 13 sebesar 97,01%. Normalisasi *min-max* didapatkan nilai k yang memiliki akurasi paling tinggi, yaitu 3 sebesar 97,01%. Normalisasi *decimal scaling* didapatkan beberapa nilai k dengan tingkat akurasi yang tinggi yaitu 3, 5, dan 12 sebesar 98,51%.

TABEL IV

NILAI K UNTUK K-NN DENGAN VARIASI NORMALISASI

Nilai k	Normalisasi		
	Akurasi Z-Score	Akurasi Min-max	Akurasi Decimal Scaling
3	97,01%	97,01%	98,51%
4	95,52%	95,52%	97,01%
5	95,52%	95,52%	98,51%
6	95,52%	95,52%	94,03%
7	95,52%	95,52%	95,52%
8	94,03%	94,03%	94,03%
9	97,01%	95,52%	95,52%
10	95,52%	92,54%	95,52%
11	97,01%	95,52%	95,52%
12	95,52%	92,54%	98,51%
13	97,01%	94,03%	95,52%
14	95,52%	94,03%	94,03%
15	95,52%	95,52%	95,52%



Gambar 14. Akurasi klasifikasi sistem

Pengujian Akurasi Klasifikasi dengan K-Fold Cross Validation

Pengujian ini dilakukan untuk mengetahui tingkat stabilitas hasil akurasi, presisi, dan *recall* terhadap hasil klasifikasi yang dihasilkan sistem jika diuji dengan data *train* (latih) dan data *test* (uji) yang bervariasi.

Dari Tabel IV pengujian nilai k pada K-NN sebelumnya untuk normalisasi *z-score* akurasi paling tinggi didapatkan dengan menggunakan nilai k untuk klasifikasi K-NN yaitu 11, nilai k untuk normalisasi *min-max* yaitu 3, nilai k untuk normalisasi *decimal scaling* yaitu 5. Tabel V merupakan evaluasi performa sistem klasifikasi dengan menggunakan pengujian *10-fold cross validation*.

TABEL V
EVALUASI PERFORMA KLASIFIKASI 10-FOLD CROSS VALIDATION

fold	Z-Score			Min-Max			Decimal Scaling		
	Akurasi	Presisi	Recall	Akurasi	Presisi	Recall	Akurasi	Presisi	Recall
1	100,00%	100,00%	100,00%	97,30%	97,50%	97,22%	100,00%	100,00%	100,00%
2	94,59%	95,24%	94,44%	94,59%	95,24%	94,44%	97,30%	97,50%	97,22%
3	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
4	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
5	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
6	67,57%	49,36%	63,99%	64,86%	47,27%	60,86%	86,49%	61,33%	84,38%
7	78,38%	74,39%	81,94%	83,78%	77,97%	81,75%	78,38%	51,10%	66,87%

8	89,19%	87,25%	91,79%	83,78%	79,23%	87,62%	89,19%	83,33%	91,79%
9	97,30%	66,67%	97,37%	97,30%	66,67%	97,37%	97,30%	66,67%	97,37%
10	91,18%	88,46%	93,75%	88,24%	85,71%	91,67%	94,12%	91,67%	95,83%
Rata-Rata	91,82%	86,14%	92,33%	90,99%	84,96%	91,09%	94,28%	85,16%	93,35%

Dari Tabel V pengujian 10-fold cross validation yang telah dilakukan didapatkan normalisasi *decimal scaling* yang memiliki tingkat akurasi yang tinggi dan relatif lebih stabil dibandingkan normalisasi *z-score* dan *min-max* dengan tingkat rata-rata akurasi 94,28%, presisinya 85,16% dan *recall*-nya sebesar 93,35%.

4. Kesimpulan

Berdasarkan hasil analisis dan implementasi pada penelitian tentang aplikasi *wireless sensor network* untuk sistem *monitoring* dan klasifikasi kualitas udara ini, dapat disimpulkan beberapa pernyataan seperti di bawah ini.

Aplikasi *wireless sensor network* yang diterapkan sebagai sistem *monitoring* dan klasifikasi kualitas udara dirancang dengan menggunakan sensor MQ-7 untuk mengetahui tingkat kadar CO, sensor MQ-131 untuk mengetahui tingkat kadar O₃, dan sensor GP2Y1010AU0F untuk mengetahui tingkat partikulat debu. Sementara itu, modul LoRa untuk komunikasi *wireless node* sensor ke *sink* dan protokol HTTP untuk pengiriman data dari *sink* ke *server*.

Pengujian jarak pengiriman LoRa dilakukan untuk mengetahui jarak maksimal pengiriman data. Pada penelitian ini didapatkan jarak jangkauan maksimal, yaitu 200 meter. Hasil ini berbeda dengan yang tertera dalam *datasheet*, yang mana jarak jangkauannya dapat sampai 10 km. Untuk menangani *missing data*, digunakan *buffer* untuk menyimpan data sementara saat *node* sensor tidak dapat mengakses *sink* untuk mengirimkan data hasil *sensing*.

Untuk penanganan data saat *node* sensor tidak dapat mengakses *sink* dalam pengiriman data hasil *sensing*, diperlukan *buffer* untuk menampung data tersebut untuk sementara. Hasil pengujian terhadap jumlah *buffer* yang dapat ditampung pada arduino *node* sensor dengan memaksimalkan *resources* yang ada yaitu 33 jumlah *buffer*.

Dari hasil pengujian untuk performa klasifikasi menggunakan *confusion matrix* dan skenario 10-fold cross validation, hasil klasifikasi yang dihasilkan oleh sistem dengan menggunakan algoritma K-Nearest Neighbor (K-NN) adalah nilai akurasi, presisi, dan *recall* yang dipengaruhi oleh jenis normalisasi data yang digunakan. Normalisasi yang memiliki tingkat performa yang lebih baik pada penelitian ini, yaitu normalisasi

decimal scaling dengan nilai k yang digunakan sebesar 5. Nilai akurasi yang didapatkan sebesar **94,28%**, presisinya **85,16%**, dan *recall*-nya sebesar **93,35%**.

Referensi

- [1] T. Astuti, T. Parenta, and H. Paddu, "Peranan Kegiatan Industri Pengolahan Terhadap Pencemaran Lingkungan di Sulawesi Selatan," *Univ. Hasanudin, Makasar*, vol. 3, no. 1, pp. 49–56, 2013.
- [2] S. Supraptini, "Pengaruh Limbah Industri Terhadap Lingkungan Di Indonesia," *Media Penelit. dan Pengemb. Kesehat.*, vol. 12, no. 2 Jun, pp. 10–19, 2002.
- [3] PERATURAN PEMERINTAH REPUBLIK INDONESIA NOMOR 41 TAHUN 1999 TENTANG PENGENDALIAN PENCEMARAN UDARA. 1999.
- [4] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks*. 2007.
- [5] M. Pule, A. Yahya, and J. Chuma, "A Wireless Sensor Network Solution for Monitoring Water Quality in Botswana," *3rd Natl. Found. Sci. Technol. Dev. Conf. Inf. Comput. Sci.*, pp. 12–16, 2016.
- [6] I. Muhammad, B. Sugiarto, and I. Sakti, "Rancang Bangun Sistem Monitoring Kualitas Udara Menggunakan Teknologi Wireless Sensor Network (WSN)," *Inkom*, vol. III, no. 1, pp. 90–96, 2009.
- [7] Y. Xu and F. Liu, "Application of Wireless Sensor Network in Water Quality Monitoring," *22017 IEEE Int. Conf. Comput. Sci. Eng. IEEE Int. Conf. Embed. Ubiquitous Comput.*, pp. 368–371, 2017.
- [8] S. Ferdoush and X. Li, "Wireless sensor network system design using Raspberry Pi and Arduino for environmental monitoring applications," *Procedia Comput. Sci.*, vol. 34, pp. 103–110, 2014.
- [9] H. Budi *et al.*, "APLIKASI WIRELESS SENSOR NETWORK (WSN) SEBAGAI SISTEM," vol. 1, no. 1, pp. 101–110, 2016.
- [10] B. Sugiarto and R. Sustika, "Data Classification for Air Quality on Wireless Sensor Network Monitoring System Using Decision Tree Algorithm," *2016 2nd Int. Conf. Sci. Technol.*, pp. 172–176, 2016.
- [11] M. Saoudi, R. Euler, T. Kechadi, and A. Cuzzocrea, "Energy-Efficient Data Mining

- Techniques for Emergency Detection in Wireless Sensor Networks,” *2016 Intl IEEE Conf. Ubiquitous Intell. Comput. Adv. Trust. Comput. Scalable Comput. Commun. Cloud Big Data Comput. Internet People, Smart World Congr.*, pp. 766–771, 2016.
- [12] *Keputusan Kepala Bapedal No. 107 Tahun 1997 Tentang : Perhitungan Dan Pelaporan Serta Informasi Indeks Standar Pencemar Udara*, no. 107. 1997.
- [13] H. Electronic, “Mq-7 Gas Sensor,” *MQ-7 Datasheet*. 2006.
- [14] Z. W. T. C. Electronics, “Ozone Gas Sensor (Model : MQ131 High Concentration) .” 2014.
- [15] SHARP, “Application note of Sharp dust sensor GP2Y1010AU0F - Sheet No.: OPI3024EN,” 2010.