

Prediksi Waktu Tempuh Bus Trans Metro Bandung dengan *Internet Of Things* dan Metode *Machine Learning*

Arrival Time Prediction Bus Trans Metro Bandung with Internet Of Things and Machine Learning Method

1st Enrico Megantara
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

enricomegantara@student.telkom
university.ac.id

2nd Rendy Munadi
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

rendymunady@telkomuniversity.
ac.id

3rd Sussi
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

sussiss@telkomuniversity.ac.id

Abstrak

Bus Rapid Transit (BRT) diharapkan dapat menekan padatannya lalu lintas Kota Bandung. Namun permasalahan yang dihadapi pihak TMB adalah bus BRT tidak memiliki jalurnya sendiri, yang terjadi adalah bus menggunakan jalur yang sama dengan kendaraan pribadi, jadi penjadwalan waktu kedatangan bus terkadang tidak sesuai estimasi waktu berbeda dari jadwal. Pada tugas akhir ini, peneliti membuat alat menggunakan mikrokomputer raspberry pi untuk tracking data Bus Trans Metro Bandung (TMB) dan *Machine Learning* (ML) untuk memprediksi waktu durasi pada halte Bus TMB yang dilalui. Algoritma ML akan memprediksi waktu durasi Bus TMB dari halte keberangkatan sampai halte tujuan. Pengambilan *dataset* dilakukan setiap waktu untuk pembuatan model. Model ML menggunakan Algoritma *Random Forest* (RF) dan *XGBoost* untuk menganalisa manakah algoritma yang paling efektif untuk memprediksi waktu durasi Bus TMB. Hasil penelitian ini didapatkan model *machine learning* regresi yang terbaik untuk memprediksi waktu kedatangan adalah model *Random Forest* dengan nilai *random state* 102. Model tersebut mendapatkan nilai tingkat akurasi yang tinggi sebesar 98% dan model mendapatkan nilai eror MAE sebesar 0,95, MSE sebesar 33,63, dan RMSE 5,80 yang cenderung lebih rendah dibandingkan dengan nilai eror dari model lain. Sehingga dapat dikatakan bahwa model *Random Forest* dengan *random state* 102 menjadi model yang paling optimal.

Kata Kunci: BRT, Raspberry pi, *Machine Learning*, *Random Forest*, *XGBoost*.

Abstract

Bus Rapid Transit (BRT) is expected to suppress the congested traffic in The City of Bandung. However, the problem faced by

the TMB is that BRT buses do not have their own lanes, what happens is that the buses use the same lanes as private vehicles, so the scheduling of bus arrival times sometimes does not match the estimated time different from the schedule. In this final project, researchers made a tool using raspberry pi microcomputers for tracking Trans Metro Bandung (TMB) and Machine Learning (ML) Bus data to predict the duration time at the TMB Bus stop that was passed. The ML algorithm will predict the duration time of the TMB Bus from the departure stop to the destination stop. Dataset retrieval is done all the time for model creation. The ML model uses the Random Forest (RF) Algorithm and XGBoost to analyze which algorithm is the most effective for predicting the duration time of the TMB Bus. The results of this study obtained the best regression machine learning model to predict arrival time is the Random Forest model with a random state value of 102. The model received a high accuracy rate value of 98% and the model got an MAE error value of 0.95, an MSE of 33.63, and an RMSE of 5.80 which tended to be lower than the error value of other models. So it can be said that the Random Forest model with a random state of 102 is the most optimal model that the Random Forest model with a random state of 102 became the most optimal model.

Keywords: BRT, Raspberry pi, *Machine Learning*, *Random Forest*, *XGBoost*.

I. PENDAHULUAN

Transportasi umum merupakan sarana yang dapat mengurangi jumlah pengguna kendaraan pribadi dan berpengaruh untuk mengurangi kepadatan pengendara [1]. Sulitnya pengembangan transportasi umum di Kota Bandung, memberikan justifikasi bahwa transportasi umum dapat menjadi solusi dimasa yang akan datang. Salah satu cara dari Dinas Perhubungan Kota Bandung

untuk mengurangi kepadatan pengendara yaitu dengan Bus Trans Metro Bandung (TMB). Namun Bus TMB mengalami banyak kendala, seperti bus tidak memiliki jalurnya sendiri dan tingkat layanan yang kurang memadai kepada para pengguna. Sehingga hal tersebut terus dilakukan evaluasi [2].

Waktu tunggu penumpang Bus TMB di halte merupakan salah satu layanan pada sistem bus yang dapat dikembangkan. Pada koridor 2 waktu tunggu bus mendapatkan rata-rata waktu tunggu penumpang 12 menit. Waktu tunggu penumpang pada halte Dengan informasi ini dapat disusun pengembangan optimisasiasasi seperti jumlah bus, jumlah penumpang, dan waktu antara [3]. Dengan pengembangan estimasi waktu tempuh antara halte ke halte memudahkan penumpang untuk mendapatkan informasi tentang durasi perjalanan bus. Sehingga dapat meningkatkan ketertarikan masyarakat untuk menggunakan transportasi umum Bus TMB. Adapun pada penelitian [4] yang mengusulkan tentang pelacakan dan prediksi waktu kedatangan BRT dari *shelter* ke *shelter* lain berbasis *Internet of Things* (IoT). Tetapi penelitian tersebut memiliki tingkat akurasi prediksi kedatangan BRT yang kurang tepat, karena hasil prediksi hanya didapatkan dengan menganalisis data yang diperoleh sebelumnya dari perangkat modul perangkat IoT.

Pada jurnal [5], telah dilakukan metode analisis untuk memperkirakan waktu sampai transit BRT dari *shelter* ke *shelter* lain menggunakan metode model GPS dan menggunakan metode *Delay*, *K-Nearest Neighbors* (KNN), *Kernel Regression* (KR), dan *Long Short Term Memory* (LSTM). Dari ke empat algoritma tersebut, algoritma *deep learning* LSTM memiliki tingkat akurasi yang baik untuk mengembangkan model prediksi waktu kedatangan BRT. Akan tetapi metode LSTM merupakan algoritma *deep learning* yang membutuhkan data yang sangat besar untuk *training*. Sehingga model yang dihasilkan akan jauh lebih besar dan spek komputer yang digunakan harus memiliki TPU/GPU yang tinggi untuk memprosesnya. Pada penelitian ini penulis menyajikan perancangan sistem untuk memprediksi lamanya waktu durasi dari halte awal sampai halte terakhir pada jalur TMB dengan menggunakan 2 metode *machine learning* *Regression* yaitu *bagging Random Forest* (RF) dan *boosting XGBoost*.

II. KAJIAN TEORI DAN METODOLOGI

A. *Internet of Things* (IoT)

Menurut [6] *Internet of Things* merupakan sebuah konsep yang bertujuan untuk memanfaatkan konektivitas internet yang terhubung secara berskala, sehingga memungkinkan kita untuk menghubungkan mesin, peralatan, dan benda fisik menggunakan modul atau sensor jaringan dan aktuator untuk memperoleh data dan mengelola kinerjanya sendiri, sehingga memungkinkan mesin untuk berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperoleh secara independen. Pada Gambar 2.1 terlihat bahwa IoT dapat digunakan pada bidang transportasi. Pada Tugas Akhir ini penulis menggunakan IoT untuk bidang transportasi. IoT

berfungsi untuk mengambil dataset yang selanjutnya diolah pada algoritma *machine learning*.

B. Mikrokomputer

Mikrokomputer Raspberry pi adalah sebuah *hardware* atau platform yang bagus untuk merancang komputasi dengan berbagai tambahan modul komputasi lainnya [7]. Pada tugas akhir ini, penulis menggunakan Raspberry pi versi 3B yang sudah terintegrasi dengan *development board* dan modul *Wi-fi* seperti pada Gambar 2.2, yang berfungsi sebagai pusat untuk pengambilan dataset yang sudah diintegrasikan dengan modul adafruit GPS.

C. Modul GPS

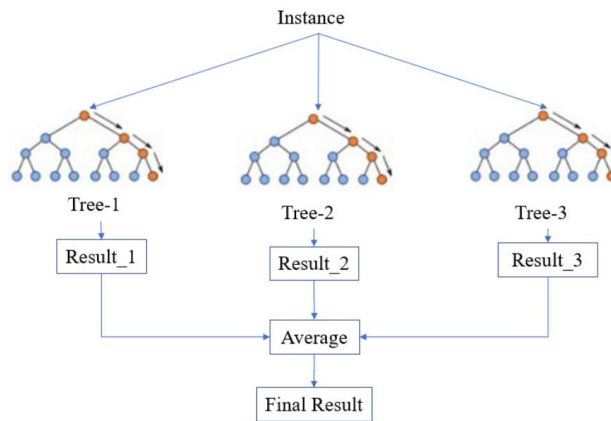
Global Positioning System (GPS) adalah navigasi sistem atau penentuan lokasi. GPS itu sendiri digunakan untuk memperoleh lokasi dan juga dapat berupa kecepatan dan memberikan informasi yang tepat waktu, terus menerus tanpa bergantung pada waktu dan cuaca tertentu. Lokasi di GPS sendiri dinyatakan dalam titik koordinat dimana nantinya koordinat dapat digunakan untuk menampilkan lokasi dimana GPS berada [8].

D. Machine Learning

Machine Learning (ML) adalah metode yang digunakan untuk memberikan komputer sebuah program yang dapat dipelajari dari sejumlah data dengan menggunakan parameter-parameter tertentu. ML pada umumnya terdiri dari 2 tipe, yaitu *Supervised Learning* dan *Unsupervised Learning*. *Supervised learning* algoritma yang belajar dari sekumpulan data dengan berisi berbagai macam input dan output. *Unsupervised Learning* algoritma yang belajar dengan cara membangun pola pengenalan dari kumpulan parameter data yang berisi input tanpa output yang ditetapkan [9][10].

E. Random Forest

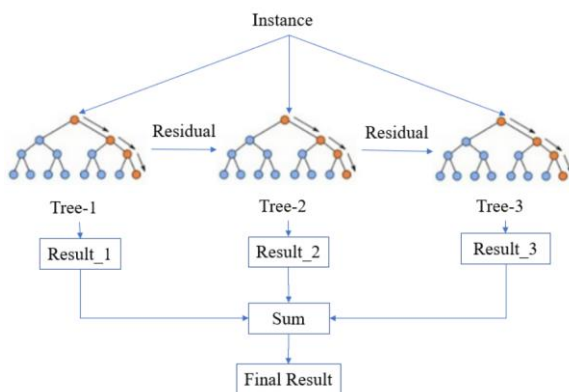
Random Forest (RF) adalah algoritma *machine learning* yang termasuk dalam jenis *supervised learning*. Algoritma RF disusun dari banyaknya algoritma pohon keputusan (*decision tree*). Jumlah banyaknya pohon yang kita gunakan pada RF, akan mempengaruhi ketepatan tingkat akurasi dan *error* atau *loss* yang diperoleh. Setiap jumlah pohon yang digunakan akan melakukan pengambilan sample akurasi. Setiap pohon juga menggunakan sample data yang berbeda. Hasil dari setiap pohon akan diambil rata-rata dan menjadi akurasi yang paling optimal [11].



Gambar 1. Architecture Random Forest.

F. XGBoost

XGBoost adalah algoritma *ensemble machine learning* yang berbasis *decision tree* tapi menggunakan *gradient framework*. XGboost melakukan pengujian data secara paralel, nilai dari pengujian yang tertinggi akan digabungkan setiap *trees* yang digunakan dan menghasilkan nilai akurasi yang paling optimal [12].



Gambar 2. Architecture XGBoost.

G. Model Performance Measurements

Setelah memlaui proses pemodelan pada RF dan XGboost, selanjutnya untuk mengukur kinerja model regresi yang sudah dibuat, penelitian ini menggunakan metode *Mean Absolute Error* (MAE) untuk mengukur nilai error absolute rata-rata. *Mean Square Error* (MSE) untuk mengukur tingkat error kuadrat rata-rata, dan metode *Root Mean Square Error* (RMSE) untuk mengukur tingkat error akar rata-rata kuadrat [13].

$$MAE = \sum_{t=1}^n \left| \frac{y_t - x_t}{y_t} \right| \quad (1)$$

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (y_t - x_t)^2}{n}} \quad (2)$$

$$MSE = \frac{1}{n} \sum_{t=1}^n (Y_i - \hat{Y}_i)^2 \quad (3)$$

III. HASIL DAN PEMBAHASAN

A. Data Preprocessing

Pada pengujian ini waktu durasi didapatkan dari modul gps, selanjutnya dataset akan dilakukan proses konversi dari file hasil traking dengan format NMEA ke format CSV. Sehingga file dapat dibaca pada data pemrosesan di python.

	A	B	C	D	E	F
1	2022-05-2 2.87'	-6.901631'	'107.6558'	'02:40:19'		
2	2022-05-2 2.82'	-6.901648'	'107.6558'	'02:40:20'		
3	2022-05-2 2.9'	-6.901645'	'107.6558'	'02:40:21'		
4	2022-05-2 3.02'	-6.901645'	'107.6558'	'02:40:22'		
5	2022-05-2 3.16'	-6.901656'	'107.6558'	'02:40:23'		
6	2022-05-2 2.97'	-6.901678'	'107.6559'	'02:40:24'		
7	2022-05-2 2.53'	-6.901688'	'107.6559'	'02:40:25'		
8	2022-05-2 1.92'	-6.901701'	'107.6559'	'02:40:26'		
9	2022-05-2 1.78'	-6.901718'	'107.6559'	'02:40:27'		
10	2022-05-2 1.37'	-6.90175'	'107.6560'	'02:40:28'		
11	2022-05-2 1.05'	-6.90176'	'107.6560'	'02:40:29'		
12	2022-05-2 1.38'	-6.901783'	'107.6560'	'02:40:30'		
13	2022-05-2 1.88'	-6.901796'	'107.6560'	'02:40:31'		
14	2022-05-2 2.32'	-6.901811'	'107.6560'	'02:40:32'		
15	2022-05-2 3.28'	-6.901846'	'107.6560'	'02:40:33'		
16	2022-05-2 4.05'	-6.901873'	'107.6560'	'02:40:34'		
17	2022-05-2 4.89'	-6.901905'	'107.6560'	'02:40:35'		

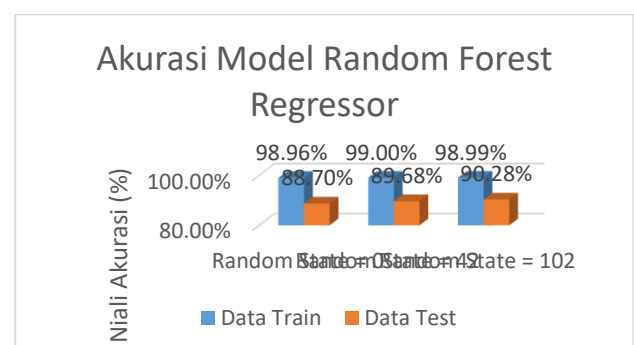
Gambar 3. Hasil konversi NMEA to CSV jalur Cicaheum ke Cibereum

B. Pengujian Machine Learning

Pada pengujian ini, diuji beberapa random state untuk menentukan parameter yang terbaik untuk memprediksi waktu durasi.

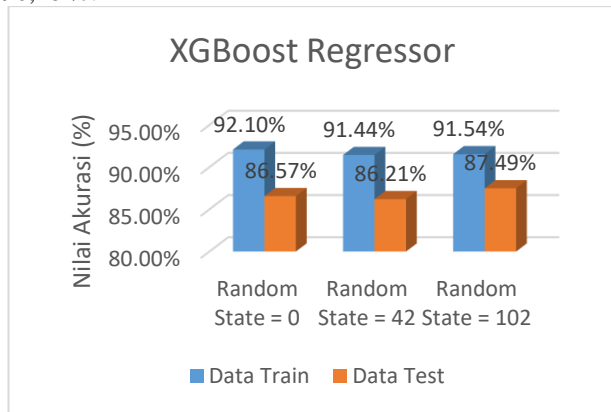
a. Pengujian Random State

Pada scenario ini, merupakan hasil dari pengujian akurasi Model *Random Forest Regressor*. Pengujian akurasi ini menggunakan *Random State* sama dengan 0, 42, dan 102.



Gambar 4. Akurasi Model Random Forest Regressor

Berdasarkan Gambar 4.5 di atas dapat diperhatikan bahwa dapat disimpulkan akurasi terbaik terletak pada *Random State* samadengan 102 dengan nilai akurasi paling tinggi dibandingkan dengan *Random State* yang lain. Pada *Random State* 102 akurasi pada data train sebesar 98,99% hamper memperoleh 99%. Sedangkan untuk hasil pengujian pada data tes mendapatkan akurasi sebesar 90,28 %.

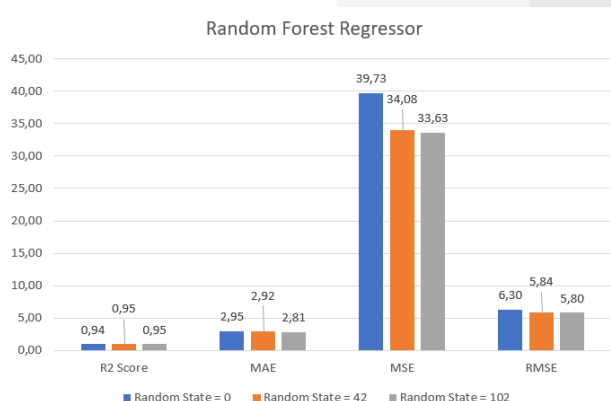


Gambar 5. Akurasi Model XGboost Regressor

Selanjutnya pada Gambar 4.6 terdapat hasil pengujian dengan menggunakan Algoritma ML *XGBoost Regressor*. Dihasilkan akurasi *training* dan tes terbaik pada *Random State* samadengan 0. Pada *Random State* 0 mendapatkan nilai akurasi pada model data train paling tinggi yaitu 92,10% dan akurasi pada data tes mendapatkan nilai 86,57%. Dimana pada algoritma RF dan *XGBoost* nilai akurasi terendah masih mendapatkan nilai diatas 80%, sehingga dapat disimpulkan bahwa model *machine learning* dapat memprediksi diatas 80% dari data actual.

b. Pengujian evaluasi model MAE, MSE, RMSE

Pada tahap ini akan dilakukan pengujian evaluasi model denngan melihat nilai *error* yang didapat dari pengujian model RF maupun *XGBoost*.



Gambar 6. Evaluasi tingkat error Random Forest

Dari grafik pada Gambar 4.7 terdapat hasil dari perhitungan R2 score, MAE, MSE dan RMSE dari algoritma RF. Pada *Random State* 42 dan *Random State* 102 mempunyai hasil yang sama yaitu 0,95. Diamana ini

merupakan hasil yang baik, karena R2 score mencari hasil yang mendekati nilai 1. Untuk pehitungan *error* MAE, MSE dan RMSE mencari nilai *error* yang paling mendekati nilai 0. Untuk perhitungan MAE *Random State* 102 mendapatkan nilai 2,81. Untuk nilai MSE *Random State* mendapatkan nilai 33,63. RMSE *Random State* 102 mendapatkan nilai 5,80. Nilai *error* yang diperoleh dari *Random State* 102 lebih rendah dibandingkan nilai dari *Random State* 42. Sehingga dapat disimpulkan untuk proses regresi menggunakan *Random State* 102 lebih baik dari menggunakan *Random State* 0 maupun 42, karena nilai *error* *Random State* 102 bernilai paling rendah atau palng mendekati 0.



Gambar 7. Evaluasi tingkat error Random Forest

Pengujian *error* selanjutnya saat menggunakan algoritma *XGBoost Regressor*. Pada pengujian ini dapat melihat pada Gambar 4.8 terdapat hasil dari perhintungan R2 score, MAE, MSE, dan RMSE. Pada pengujian *error* *XGBoost Random State* 102 mendapatkan nilai R2 score paling tinggi, dan untuk MAE tidak terlalu bagus dibandingkan dengan *Random State* lainnya, untuk MSE mendapatkan nilai paling rendah yaitu 43,26 dan nilai RMSE 6,58.

C. Pengujian Prediksi Waktu Durasi

Berikut pada Tabel 4.6 merupakan table dengan data actual. Pada tahap ini akan dilakukan perbandingan hasil *output* algoritma RF dan *XGBoost Regressor* dengan hasil durasi waktu dari data actual.

Tabel 1.Data prediksi waktu aktual

Ha ri	Jal ur	Halte Awal	Halte Sampai	Ja m	Me nit	Durasi (menit)
1	1	6	17	6	53	56,18
2	2	22	28	8	39	21,10
3	1	2	15	5	46	44,58
1	1	0	3	6	37	7,87
1	2	14	16	15	18	6,70

Tabel 2. Pengujian prediksi waktu durasi *Random Forest*

Input						Output
Hari	Jalur	Halte Awal	Halte Sampai	Jam	Menit	Durasi (menit)
1	1	6	17	6	53	47,15
2	2	22	28	8	39	21,64
3	1	2	15	5	46	47,42
1	1	0	3	6	37	7,64
1	2	14	16	15	18	9,10

Tabel 3. Pengujian prediksi waktu durasi *XGBoost*

Input						Output
Hari	Jalur	Halte Awal	Halte Sampai	Jam	Menit	Durasi (menit)
1	1	6	17	6	53	57,68
2	2	22	28	8	39	21,36
3	1	2	15	5	46	49,80
1	1	0	3	6	37	10,87
1	2	14	16	15	18	9,84

Pada scenario ini akan dilakukan perbandingan hasil *output* antara algoritma RF dan *XGBoost Regressor*. Pengujian ini ditujukan untuk mengetahui hasil selisih *output* antar algoritma dengan *output* durasi data actual. Dari hasil yang diperoleh dapat disimpulkan bahwa masing-masing algoritma terkadang mendapatkan selisih hasil *output* yang besar dengan data actual durasi. Dari Tabel 4.7 Algoritma RF pada pengujian pertama mendapatkan selisih yang cukup besar, sekitar 10 menit durasi perbedaan *output* data actual. Sedangkan dari Tabel 4.8 Algoritma *XGBoost* pada pengujian ke tiga, empat, dan lima juga mempunyai perbedaan selisih dari *output* durasi data actual.

Dari hasil pengujian yang telah dilakukan, dapat kita lihat bahwa perbedaan hasil *output* dari Algoritma RF dan Algoritma *XGBoost* mendapatkan hasil selisih *output* durasi. Hasil *output* durasi dari Algoritma RF dengan akurasi 98% lebih sering mendekati dari data actual, dibandingkan dengan algoritma *XGBoost* dengan akurasi 92%

IV. KESIMPULAN

Kesimpulan yang didapat dari penelitian ini, data yang didapat dari *tracking* akan dikonversi yang sebelumnya berbentuk protokol menjadi bentuk *text* akan mudah dibaca oleh manusia. Setelah itu data akan dilakukan *preprocessing* untuk labling data dari *categorical* diubah ke *numeric* supaya dapat dilakukan proses *machine learning* regresi. Model *machine learning* mendapatkan akurasi dari algoritma RF sebesar 98,96% dari *Random State* 0, akurasi 99% dari *Random State* 42, dan akurasi 98,99 dari *Random State* 102. *XGBoost Regressor* mendapat nilai akurasi 92,10% dari *Random State* 0, akurasi 91,44% dari *Random State* 42, dan akurasi 91,54% dari *Random State* 102. Nilai akurasi tertinggi terdapat pada algoritma RF st dengan akurasi 99% dibandingkan dengan

hasil nilai akurasi dari model *XGBoost*. Ketika dilakukan uji coba test hasil *output* prediksi waktu durasi yang dibandingkan dengan data actual, hasil *output* dari algoritma RF lebih sering mendekati sama dengan hasil *output* data actual jika dibandingkan dengan hasil *output* durasi *XGBoost*. Jadi model yang akan dideploy pada *website* adalah model dengan algoritma RF. Hasil evaluasi performa model RF dengan *Random State* 102 mendapatkan nilai *R2 score* paling besar dengan nilai 0,95 dan mendapatkan nilai error MAE sebesar 2,81, nilai MSE 33,63, dan nilai RMSE 5,80. Nilai RF dengan *Random State* 102 adalah nilai evaluasi *error* paling rendah dibandingkan dengan nilai *Random State* yang lain. Karena semakin mendekati 0 maka nilai *error* semakin bagus. Sehingga nilai dari RF dengan *Random State* 102 akan dideploy pada *website* prediksi waktu durasi.

V. REFERENSI

- [1] DINAS PERHUBUNGAN PEMERINTAH KOTA BANDUNG TAHUN, "Rencana Strategis Dinas Perhubungan Kota Bandung 2019 - 2023," DINAS PERHUBUNGAN PEMERINTAH KOTA BANDUNG TAHUN, 2019.
- [2] Listifadah and Reni Puspitasari, "EVALUASI KINERJA TRANS METRO BANDUNG," *Puslitbang Perhubungan Darat dan Perkeretaapian*, pp. 65–77, 2015.
- [3] R. H. Rahmadiensyah and T. B. Joewono, "WAKTU TUNGGU PENUMPANG BUS TRANS METRO BANDUNG," 2014.
- [4] Hafiizh Nur M. A, Sugondo Hadiyoso, Fefa Bianca Belladina, Dadan Nur Ramadan, and Inung Wijayanto, "Tracking, Arrival Time Estimator, and Passenger Information System on Bus Rapid Transit (BRT)," *Information and Communication Technology (ICICT)*, Aug. 2020.
- [5] D. Liu, J. Sun, and S. Wang, "BusTime: Which is the Right Prediction Model for My Bus Arrival Time?," Mar. 2020, [Online]. Available: <http://arxiv.org/abs/2003.10373>
- [6] M. A. Razzaque and M. R. Karim, "Hancs-On Deep Learning for ICT Train neural network models to develop intelligent lof applications."
- [7] IEEE Electromagnetic Compatibility Society, IEEE Industry Applications Society, IEEE Power & Energy Society, Institute of Electrical and Electronics Engineers, and I. Industrial and Commercial Power Systems Europe (2nd: 2018: Palermo, 2018 *conference proceedings: 2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC)*.

- [8] W. Wang, G. Chakraborty, and B. Chakraborty, "Predicting the risk of chronic kidney disease (Ckd) using machine learning algorithm," *Applied Sciences (Switzerland)*, vol. 11, no. 1, pp. 1–17, Jan. 2021, doi: 10.3390/app11010202.
- [9] M. Lu and F. Li, "Survey on lie group machine learning," *Big Data Mining and Analytics*, vol. 3, no. 4. Tsinghua University Press, pp. 235–258, Dec. 01, 2020. doi: 10.26599/BDMA.2020.9020011.
- [10] M. P. Hosseini, A. Hosseini, and K. Ahi, "A Review on Machine Learning for EEG Signal Processing in Bioengineering," *IEEE Rev Biomed Eng*, vol. 14, pp. 204–218, 2021, doi: 10.1109/RBME.2020.2969915.
- [11] W. Wang, G. Chakraborty, and B. Chakraborty, "Predicting the risk of chronic kidney disease (Ckd) using machine learning algorithm," *Applied Sciences (Switzerland)*, vol. 11, no. 1, pp. 1–17, Jan. 2021, doi: 10.3390/app11010202.
- [12] Y. Jiang, G. Tong, H. Yin, and N. Xiong, "A Pedestrian Detection Method Based on Genetic Algorithm for Optimize XGBoost Training Parameters," *IEEE Access*, vol. 7, pp. 118310–118321, 2019, doi: 10.1109/ACCESS.2019.2936454.
- [13] N. Khan, F. U. M. Ullah, Afnan, A. Ullah, M. Y. Lee, and S. W. Baik, "Batteries State of Health Estimation via Efficient Neural Networks with Multiple Channel Charging Profiles," *IEEE Access*, vol. 9, pp. 7797–7813, 2021, doi: 10.1109/ACCESS.2020.3047732.