



SNESTIK

Seminar Nasional Teknik Elektro, Sistem Informasi,
dan Teknik Informatika

<https://ejurnal.itats.ac.id/snestik> dan <https://snestik.itats.ac.id>



Informasi Pelaksanaan :

SNESTIK I - Surabaya, 26 Juni 2021

Ruang Seminar Gedung A, Kampus Institut Teknologi Adhi Tama Surabaya

Informasi Artikel:

DOI : 10.31284/p.snestik.2021.1785

Prosiding ISSN 2775-5126

Fakultas Teknik Elektro dan Teknologi Informasi-Institut Teknologi Adhi Tama Surabaya
Gedung A-ITATS, Jl. Arief Rachman Hakim 100 Surabaya 60117 Telp. (031) 5945043
Email : snestik@itats.ac.id

Smart Metering untuk Pengidentifikasi Jenis Beban Listrik Rumah Tangga Berbasis IoT Menggunakan PZEM-004T

Muhammad Ary Murti¹, Muhammad Hablul Barri², Andi Shridivia Nuran^{3*},
Selvy Zelia Sari⁴, Difa Surya Putra⁵, dan Syafri Akhsanul Iqbal⁶

Universitas Telkom^{1,2,3,4,5,6}

e-mail: *andishridivia@gmail.com

ABSTRACT

Excessive use of electricity can increase global warming and other negative impacts. Monitoring electricity usage by knowing which electrical devices are being used can prevent electricity wastage. In this research, designed a system for identifying the type of electrical load in real-time based on the Internet of Things (IoT). Load identification systems can also be utilized in various applications, such as electricity theft monitoring systems, electricity billing systems, smart grid energy management, and home electrical system automation. This system uses the PZEM-004T sensor which is integrated with the Raspberry Pi microcontroller then the data is sent to the server by accessing the Antares IoT API (application programming interface), the data sent is the type of load which is analyzed using the Decision Tree model. From this research, tests were carried out on 5 types of electronic devices: fan, 12 W LED lamp, rice cooker, TV, and smartphone. The test results show that the system can identify all 5 electronic devices with 100% accuracy, with an average time required for the system to identify and transmit data to Antares is less than 1 second.

Keywords: Active power; Decision Tree; Current; Identification; Load type.

ABSTRAK

Penggunaan listrik yang berlebihan dapat meningkatkan pemanasan global dan dampak negatif lainnya. Memantau penggunaan listrik dengan mengetahui perangkat listrik yang sedang terpakai dapat mencegah terjadinya pemborosan listrik. Pada penelitian ini, dirancang sistem pengidentifikasi jenis beban listrik secara *real time* berbasis *Internet of Things* (IoT). Sistem identifikasi beban juga dapat dimanfaatkan di berbagai aplikasi, seperti sistem pemantauan pencurian listrik, sistem penagihan listrik, manajemen energi di *smart grid*, dan otomatisasi sistem kelistrikan rumah. Sistem menggunakan sensor PZEM-004T yang

diintegrasikan dengan Raspberry Pi kemudian data dikirim ke server dengan mengakses API (*application programming interface*) IoT Antares. Data yang terkirim berupa jenis beban yang dianalisis menggunakan model Decision Tree. Dari penelitian yang dibangun, dilakukan pengujian pada 5 jenis perangkat elektronik yaitu kipas, lampu LED 12 W, *rice cooker*, TV, dan HP. Diperoleh hasil pengujian bahwa sistem dapat mengenali kelima perangkat elektronik dengan akurasi 100%, dengan rata-rata waktu yang diperlukan sistem untuk mengenali dan mengirimkan data ke Antares adalah kurang dari 1 *second*.

Kata Kunci: Arus; Daya aktif; Decision Tree; Identifikasi; Jenis beban.

PENDAHULUAN

Listrik merupakan salah satu kebutuhan pokok manusia untuk menunjang aktivitas sehari-hari. Energi listrik diperoleh dari berbagai sumber dan diubah melalui pembangkit listrik. Sistem pembangkit listrik berdampak negatif pada kesehatan, sumber daya, dan ekosistem [1]. Namun seiring berkembangnya zaman, jenis-jenis perangkat elektronik semakin beragam menyebabkan konsumsi listrik semakin lama semakin meningkat. Penghematan listrik harus dilakukan agar energi listrik mudah didapatkan dan bertahan dalam jangka waktu panjang. Dalam rumah tangga, energi listrik dimanfaatkan untuk menyalakan berbagai macam perangkat elektronik. Pemakaian listrik berlebihan salah satunya diakibatkan karena konsumen tidak bisa mengetahui beban apa saja yang sedang mengonsumsi listrik. Untuk mengatasi masalah tersebut dibutuhkan alat yang dapat mengidentifikasi perangkat listrik yang sedang dikonsumsi secara *real-time*.

Beberapa penelitian sebelumnya telah dirancang sistem untuk memantau penggunaan listrik rumah tangga maupun gedung menggunakan mikrokontroler Wemos D1 mini [2], NodeMCU [3][4], ESP32 [5], Arduino Mega [6], maupun mikrokomputer Raspberry Pi [7]. Sistem tersebut dilengkapi dengan modul PZEM-004T. Modul ini berfungsi untuk mengukur daya, tegangan, arus, dan *power factor* yang terdapat pada sebuah aliran listrik [7]. Namun pada beberapa penelitian tersebut sistem tidak dapat mendeteksi atau mengenali jenis beban listrik yang sedang dikonsumsi. Beban listrik adalah sesuatu yang ditanggung oleh pembangkit listrik [8]. Beban listrik pada rumah tangga merupakan perangkat-perangkat yang mengonsumsi listrik yang digunakan dalam rumah.

Pada penelitian ini, dirancang sebuah sistem untuk mengidentifikasi jenis beban listrik berbasis *Internet of Things* (IoT) secara *real-time* menggunakan sensor PZEM-004T sebagai *input* yang diintegrasikan dengan Raspberry Pi Model B+ sebagai *controller*. Sistem ini menggunakan *platform* API IoT Antares untuk menyimpan data jenis beban secara *real-time*. Pengidentifikasian jenis beban dilakukan berdasarkan pada perubahan konsumsi listrik yang masuk pada rumah. Proses identifikasi jenis beban diawali dengan mengumpulkan informasi yang berkaitan dengan konsumsi listrik masing-masing perangkat elektronik yang menjadi *dataset* untuk melatih model *machine learning*. Analisis klasifikasi jenis beban listrik menggunakan algoritma Decision Tree yang merupakan salah satu algoritma *supervised machine learning* yang cukup kuat, mampu melakukan penyesuaian untuk *dataset* yang cukup kompleks.

METODE

Sumber Listrik PLN

Sumber listrik PLN merupakan sumber energi listrik dengan arus bolak-balik atau *alternating current* (AC) [9]. Tegangan (*V*) dan arus (*I*) AC umumnya dinyatakan dengan nilai RMS (*root mean square*) yang merupakan akar dari nilai rata-rata berdasarkan suatu fungsi yang dikuadratkan. Daya listrik yang merupakan laju hantaran energi listrik dalam rangkaian listrik yang terbagi menjadi tiga yaitu daya aktif, daya reaktif, dan daya semu.

1. *Daya Aktif (Active Power)*

Daya aktif adalah daya yang sesungguhnya terpakai untuk melakukan energi. Satuan daya aktif adalah W [10]. Daya aktif (P) dirumuskan pada Persamaan 1.

$$P = I \times V \times \cos \phi \quad (1)$$

2. *Daya Reaktif (Reactive Power)*

Daya reaktif adalah jumlah daya yang dibutuhkan dalam pembentukan medan magnet yang akan membentuk *fluks* medan magnet. Satuan daya reaktif adalah VAR [10]. Daya reaktif (Q) dirumuskan pada Persamaan 2.

$$Q = I \times V \times \sin \phi \quad (2)$$

3. *Daya Semu (Apparent Power)*

Daya semu adalah daya yang dihasilkan dari perkalian antara tegangan RMS dan arus RMS juga daya yang dihasilkan dari penjumlahan trigonometri daya aktif dan reaktif. Satuan daya semu adalah VA [10]. Daya semu (S) dirumuskan pada Persamaan 3.

$$S = V_{RMS} \times I_{RMS} \quad (3)$$

Algoritma Decision Tree

Algoritma Decision Tree adalah salah satu algoritma *machine learning* untuk klasifikasi [11]. Decision tree merupakan algoritma yang akan menghasilkan pola atau model dalam bentuk pohon keputusan. Algoritma ini dapat digunakan pada masalah-masalah dengan *multi-output*, dan juga algoritma yang kuat karena mampu melakukan penyesuaian untuk *dataset* yang cukup kompleks.

Proses klasifikasi dimulai dari *root node* pohon keputusan dan secara rekursif berlanjut hingga mencapai *leaf node* dengan label kelas. Setiap *internal node* pada pohon keputusan membagi data ke dalam dua kelompok dengan *rule-rule* berdasarkan harga *cut-off* pada fitur di *node* tersebut untuk dilanjutkan ke arah subpohon kiri atau kanan hingga mencapai *leaf node*.

Saat membangun pohon keputusan, tujuan di setiap *node* adalah untuk memilih kondisi pemisah yang paling baik membagi *dataset* menjadi beberapa *subset* yang homogen. *Goodness of split criterion* diperkenalkan, diturunkan dari gagasan *impurity* [12]. Gini index atau Gini impurity dihitung untuk menentukan cabang pada *internal node*. Persamaan 4 digunakan untuk menghitung Gini untuk data (D):

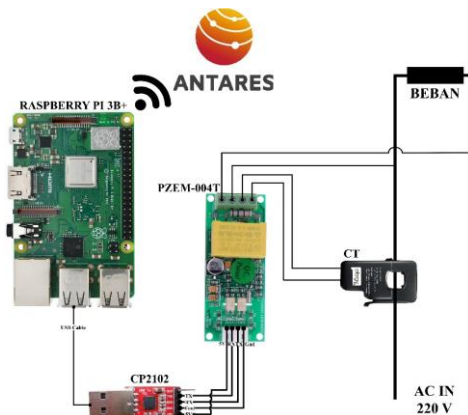
$$Gini(D) = 1 - \sum_{i=1}^c P_i^2, \quad (4)$$

dengan c adalah jumlah kelas dalam data dan P_i adalah probabilitas sampel data pada kelas tersebut. Jika kumpulan data D terpecah menjadi D_1 dan D_2 sesuai dengan fitur A , maka Persamaan 5 digunakan:

$$Gini_A(D) = \frac{|D_1|}{|D|} \times Gini(D_1) + \frac{|D_2|}{|D|} \times Gini(D_2) \quad (5)$$

Gini impurity dihitung untuk setiap nilai pada fitur. Fitur dengan Gini impurity terendah yang dipilih dan cabang dibuat berdasarkan fitur tersebut [13].

Perancangan Sistem Perangkat Keras

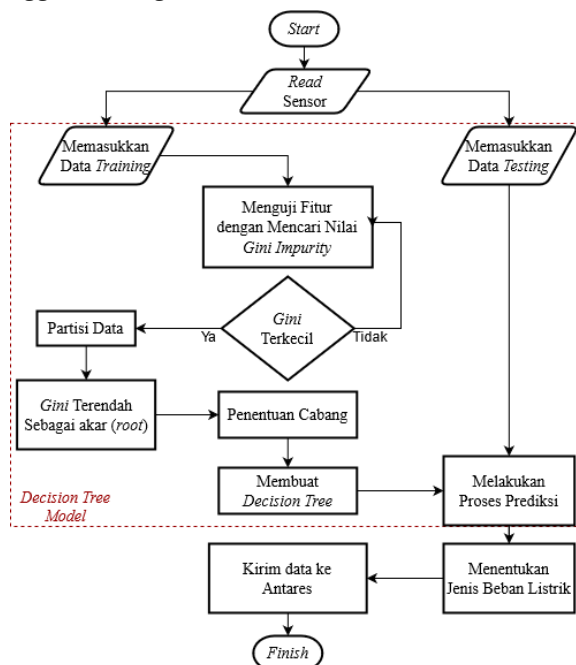


Gambar 1. Perancangan sistem perangkat keras.

Perancangan sistem perangkat keras yang dibangun pada penelitian ini dapat dilihat pada Gambar 1. Pada sistem ini, digunakan FTDI Serial USB Converter sebagai antarmuka untuk mengkomunikasikan modul PZEM-004T dengan Raspberry Pi 3 Model B+. Modul PZEM-004T yang dilengkapi dengan *Current Transformer* (CT) berperan sebagai *input*, *input* yang diperoleh berupa arus, tegangan, daya dan *power factor*. Analisis identifikasi jenis beban dilakukan dengan mengimplementasikan algoritma Decision Tree pada Raspberry Pi 3 Model B+. Sistem kemudian mengirimkan data *output* ke server Antares.

Perancangan Sistem Perangkat Lunak

Gambar 2 berikut adalah *flowchart* sistem pemantauan dan pendeteksian penggunaan listrik pada rumah menggunakan algoritma Decision Tree.



Gambar 2. Flowchart sistem.

Berdasarkan pada *flowchart* sistem yang ditunjukkan pada Gambar 2, sistem dimulai dari pembacaan sensor, kemudian data dari perubahan arus dan daya aktif yang dideteksi sensor dibagi menjadi 2 data uji dan data latih. Data uji digunakan untuk membentuk model Decision Tree pada sistem agar dapat mengidentifikasi jenis beban. Pembentukan model Decision Tree diawali dengan mencari nilai Gini *impurity* setiap fitur. Fitur dengan nilai Gini *impurity* terendah akan menjadi *rule* acuan pada *node* akar dan seterusnya hingga *node* sudah tidak memiliki cabang. Jika model Decision Tree sudah terbentuk, sistem mampu mengidentifikasi secara *real-time*. Hasil identifikasi dikirim ke server Antares untuk disajikan kepada pengguna.

HASIL DAN PEMBAHASAN

Pengujian dilakukan pada lima jenis perangkat elektronik dengan spesifikasi tertentu, yaitu *kipas*, *lampu LED 12 W*, *rice cooker (mode cook)*, TV, dan HP. Pada penelitian ini variabel data yang akan diukur adalah tegangan (V_{rms}), arus (I_{rms}), daya aktif (P) dan faktor daya. Pengukuran dilakukan menggunakan sensor PZEM-004T, sampel pengukuran masing-masing jenis perangkat elektronik yang digunakan pada Tabel 1.

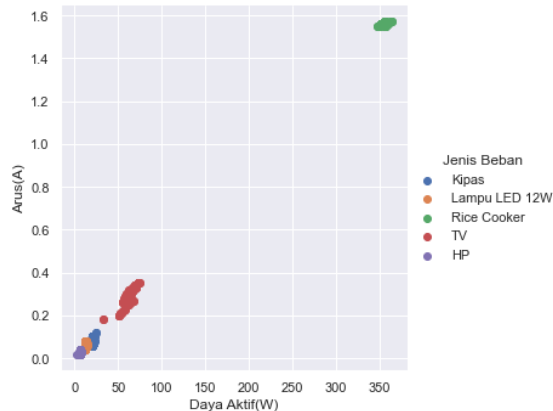
Tabel 1. Sampel pengukuran masing-masing jenis beban.

No.	Jenis Beban	Merk	V_{rms}	I_{rms}	P	Faktor daya
1	Kipas	Nagoya	224,90	0,14	21,00	0,68
2	Lampu LED (12 W)	Philips	226,30	0,11	13,40	0,54
3	Rice Cooker	Sanken	225,80	1,64	369,20	1,00
4	TV	Sharp	224,30	0,41	78,50	0,86
5	HP	Iphone	223,70	0,07	7,30	0,50

Pengambilan data dilakukan pada setiap perangkat listrik yang diaktifkan secara bergantian dengan perangkat listrik lainnya, untuk mengetahui pengaruh penggunaan listrik perangkat tersebut terhadap perangkat lain. Setiap perangkat listrik terdiri dari 80 data, dengan total keseluruhan 400 data. Setelah dilakukan pengukuran, *dataset* dimuat untuk melatih algoritma Decision Tree dan membentuk model pohon keputusan. Fitur yang digunakan oleh sistem ini pada model Decision Tree adalah perubahan arus dan daya aktif, contoh *dataset* seperti terlihat pada Tabel 2 dan plot *dataset* dapat dilihat pada Gambar 3.

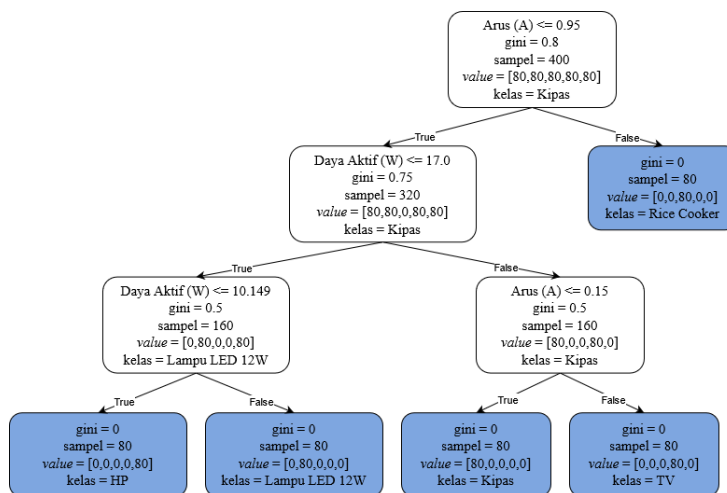
Tabel 2. Sampel *dataset*.

Arus (A)	Daya Aktif (W)	Jenis Beban
0,1200	25,300	Kipas
0,1000	22,400	Kipas
0,0700	14,690	Lampu LED 12 W
0,0700	14,699	Lampu LED 12 W
1,5690	363,700	Rice Cooker
1.5600	353,800	Rice Cooker
0,3500	74,100	TV
0,2599	55,199	TV
0,0299	7,500	HP
0,0399	7,000	HP



Gambar 3. Plot *dataset*.

Dataset dimuat dan dibagi menjadi data latih dan data uji menggunakan Scikit-Learning *library*, bertujuan untuk mengevaluasi performa model Decision Tree. Berdasarkan hasil *splitting dataset*, diperoleh akurasi sebesar 100% dengan pembagian data latih 80% (320 data) dan data uji 20% (80 data) dari 400 data pada *dataset* yang dikumpulkan. Gambar 4 merupakan visualisasi pohon keputusan yang terbentuk dari pola *dataset* yang dimuat untuk membangun model Decision Tree (DT).



Gambar 4. Visualisasi pohon keputusan.

Pohon keputusan yang ditunjukkan pada Gambar 4 terdiri dari dua fitur yang menjadi *node*. *Internal node* ditandai dengan kotak berwarna putih dan *leaf node* ditandai dengan kotak berwarna biru. Arus merupakan fitur yang menjadi acuan pada *root node* karena nilai Gini *impurity* yang diperoleh pada fitur arus lebih kecil dibandingkan dengan fitur daya berdasarkan *fitting dataset*.

Berdasarkan pengujian *dataset* yang memperoleh akurasi 100% maka dilakukan pengujian pendeteksian jenis beban secara *real-time*. Percobaan dilakukan sebanyak total 50 kali, pada kondisi *ON* dan *OFF* untuk 5 perangkat elektronik yang diuji. Kemudian, dihitung waktu yang dibutuhkan untuk memproses pendeteksian jenis beban dan mengirimkan data ke server Antares. Pengujian dilakukan pada kondisi ideal yaitu terminal listrik yang digunakan terhindar gangguan perangkat elektronik yang tidak terdapat dalam kategori pengujian dan kecepatan akses internet yang stabil. Hasil pengujian dapat dilihat pada Tabel 3.

Tabel 3. Hasil pengujian pada perangkat elektronik

Percobaan ke-	ON				OFF			
	Arus (A)	Daya (W)	Hasil	Waktu (s)	Arus (A)	Daya (W)	Hasil	Waktu (s)
Kipas								
1	0,1089	23,5	sesuai	0,46	0,1099	23,2	sesuai	0,26
2	0,1109	23,4	sesuai	0,39	0,1089	23,1	sesuai	0,20
3	0,1099	23,4	sesuai	0,39	0,1099	23,5	sesuai	0,37
4	0,1089	23,2	sesuai	0,28	0,1089	23,2	sesuai	0,36
5	0,1099	22,8	sesuai	0,35	0,1089	22,7	sesuai	0,44
Lampu LED 12W								
6	0,074	12,7	sesuai	0,19	0,074	12,5	sesuai	0,31
7	0,074	12,5	sesuai	0,39	0,074	12,4	sesuai	0,38
8	0,072	12,5	sesuai	0,10	0,072	12,4	sesuai	0,23
9	0,072	12,4	sesuai	0,20	0,071	12,3	sesuai	0,27
10	0,071	12,5	sesuai	0,15	0,072	12,4	sesuai	0,21
Rice Cooker (mode cook)								
11	1,593	365,1	sesuai	0,22	1,598	367,4	sesuai	0,29
12	1,594	365,6	sesuai	0,32	1,593	365	sesuai	0,28
13	1,587	362,4	sesuai	0,43	1,591	364,6	sesuai	0,24
14	1,596	367,1	sesuai	0,15	1,588	363,7	sesuai	0,40
15	1,597	367,7	sesuai	0,33	1,595	367,4	sesuai	0,41
TV								
16	0,24	51,5	sesuai	0,26	0,174	33,5	sesuai	0,45
17	0,174	33,5	sesuai	0,23	0,1679	32,2	sesuai	0,15
18	0,1679	32,2	sesuai	0,19	0,173	33,3	sesuai	0,68
19	0,173	33,3	sesuai	0,38	0,231	47,1	sesuai	0,37
20	0,231	47,1	sesuai	0,58	0,254	52,6	sesuai	0,27
HP charger								
21	0,027	6,6	sesuai	0,42	0,027	6,6	sesuai	0,24
22	0,27	6,6	sesuai	0,21	0,027	6,69	sesuai	0,39
23	0,028	6,5	sesuai	0,34	0,027	6,5	sesuai	0,16
24	0,028	6,6	sesuai	0,24	0,028	6,6	sesuai	0,39
25	0,0289	6,6	sesuai	0,23	0,0279	6,5	sesuai	0,43

KESIMPULAN

Berdasarkan hasil perancangan dan pengujian pada sistem pengidentifikasi jenis beban, dapat disimpulkan bahwa sistem yang dibangun pada penelitian ini adalah sistem yang berbasis *Internet of Things* (IoT) menggunakan model Decision Tree untuk pengidentifikasian jenis beban. Pengujian sistem diawali dengan mengumpulkan *dataset* untuk menguji dan mengevaluasi model Decision Tree dan memperoleh akurasi sebesar 100%. Pada pengujian sistem yang menggunakan

fitur perubahan daya dan arus pada model Decision Tree, sistem mampu mengidentifikasi jenis beban dengan akurasi 100% dan mengirimkan hasil identifikasi ke server Antares dengan waktu kurang dari 1 second (rata-rata 0,3122 second).

DAFTAR PUSTAKA

- [1] W. Krewitt, R. Friedrich, and A. Trukenmüller, "Comparison of health and environmental impacts from electricity generation systems," *Int. J. Risk Assess. Manag.*, vol. 3, no. 1, 2002, doi: 10.1504/IJRAM.2002.001519.
- [2] K. Chooruang and K. Meekul, "Design of an IoT Energy Monitoring System," *Int. Conf. ICT Knowl. Eng.*, vol. 2018-Novem, pp. 48–51, 2019, doi: 10.1109/ICTKE.2018.8612412.
- [3] T. Tantidham, S. Ngamsuriyaros, N. Tungamnuayrith, T. Nildam, K. Banthao, and P. Intakot, "Energy Consumption Collection Design for Smart Building," *2018 Int. Conf. Embed. Syst. Intell. Technol. Int. Conf. Inf. Commun. Technol. Embed. Syst. ICESIT-ICICTES 2018*, no. 4, pp. 1–6, 2018, doi: 10.1109/ICESIT-ICICTES.2018.8442052.
- [4] A. Furqon, A. B. Prasetijo, and E. D. Widiyanto, "Rancang Bangun Sistem Monitoring dan Kendali Daya Listrik pada Rumah Kos Menggunakan NodeMCU dan Firebase Berbasis Android," *Techné J. Ilm. Elektrotek.*, 2019, doi: 10.31358/techné.v18i02.202.
- [5] I. Abed and H. Naser, "ESP32 Microcontroller Based Smart Power Meter System Design and Implementation," *Al-Rafidain Eng. J.*, vol. 25, no. 2, 2020, doi: 10.33899/rengj.2020.127111.1038.
- [6] S. Anwar, T. Artono, N. Nasrul, D. Dasrul, and A. Fadli, "Pengukuran Energi Listrik Berbasis PZEM-004T," *Pros. Semin. Nas. Politek. Negeri Lhokseumawe*, vol. 3, no. 1, p. 272, 2020, [Online]. Available: <http://e-jurnal.pnl.ac.id/index.php/semnaspn/article/view/1694>.
- [7] F. Habibi, S. Setiawidayat, and M. Mukhsim, "Alat Monitoring Pemakaian Energi Listrik Berbasis Android Menggunakan Modul PZEM-004T," *Pros. Semin. Nas. Teknol. Elektro Terap. 2017*, vol. 01, no. 01, pp. 157–162, 2017.
- [8] Suaibah, S. Martha, and D. Kusnandar, "Pemodelan Double Seasonal Autoregressive Integrated Moving Average Pada Peramalan Penggunaan Beban Listrik," *Bimaster Bul. Ilm. Mat. Stat. dan Ter.*, vol. 9, no. 1, 2020, doi: 10.26418/bbimst.v9i1.38028.
- [9] T. Nusa, S. R. U. A. Sompie, and E. M. Rumbayan, "Sistem Monitoring Konsumsi Energi Listrik Secara Real Time Berbasis Mikrokontroler," *J. Tek. Elektro dan Komput.*, vol. 4, no. 5, pp. 19–26, 2015, doi: 10.35793/jtek.4.5.2015.9974.
- [10] A. Belly, A. Dadan H, C. Agusman, and B. Lukman, "Daya aktif, reaktif & nyata," *Fak. Tek. Univ. Indones.*, 2010.
- [11] J. R. Quinlan, "Induction of Decision Trees," *Mach. Learn.*, vol. 1, no. 1, 1986, doi: 10.1023/A:1022643204877.
- [12] S. Tangirala, "Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm," *Int. J. Adv. Comput. Sci. Appl.*, no. 2, 2020, doi: 10.14569/ijacsa.2020.0110277.
- [13] H. Arabshahi and H. Fazlollahtabar, "Classifying Innovative Activities Using Decision Tree and Gini Index," *Int. J. Innov. Technol. Manag.*, vol. 15, no. 3, 2018, doi: 10.1142/S0219877018500256.