

Identifikasi Beban Listrik Rumah Tangga Menggunakan Metode *Recurrent Neural Network*

1st Mutiara Nurjanah
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

mutiaranurj@student.telkomuniversity.
ac.id

2nd Randy Erfa Saputra
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

resaputra@telkomuniversity.ac.id

3rd Casi Setianingsih
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

setiacasie@telkomuniversity.ac.id

Abstrak— Kebutuhan listrik rumah tangga banyaknya alat yang menggunakan energi listrik sebagai sumber energi utama tidak menutup kemungkinan konsumsi energi listrik semakin lama semakin meningkat, hal ini menyebabkan pembayaran biaya listrik akan terus meningkat terutama pada kebutuhan listrik rumah tangga. Metode *Recurrent Neural Network* (RNN) untuk memantau penggunaan listrik satu fasa yang ada di alat rumah tangga. Dapat mengidentifikasi perangkat listrik berdasarkan jenis beban listrik, metode ini dapat mengurangi peningkatan pembayaran biaya listrik dengan mengidentifikasi perangkat listrik yang dipakai berdasarkan jenis beban listrik yang dipakai. Data diambil dengan menggunakan bantuan perangkat keras. Pengujian sistem mendeteksi jenis beban listrik yang dilakukan pada tiga perangkat listrik rumah tangga yakni kipas, pemanas air dan setrika. Penelitian ini diawali dengan pengumpulan data yang berhubungan dengan perangkat yang akan diuji. Berdasarkan hasil pengujian akan dilakukan proses pengidentifikasian jenis beban listrik yang sedang digunakan. Dari hasil pengujian sistem dapat mengidentifikasi ketiga perangkat tersebut dengan akurasi, presisi, *recall*, dan *F1-score* adalah 97.90%, 98.39%, 97.39%, dan 97.83% menggunakan metode RNN.

Kata kunci— identifikasi beban listrik; listrik rumah tangga; RNN.

juga akan selalu bertambah seiring dengan berkembangnya alat elektronik yang semakin canggih. Namun peralatan listrik maupun elektronika dalam rumah tangga maupun industri umumnya bersifat induktif sehingga akan menimbulkan beda fasa antara tegangan dan arus yang mengalir [3]. Penelitian perhitungan prediksi beban listrik menggunakan kecerdasan buatan telah banyak dilakukan. Diantaranya prediksi beban listrik di pulau Bali menggunakan jaringan syaraf tiruan [4]. *Neural Network* memecahkan masalah nonlinier tinggi dan ketidakpastian model beban listrik [5].

Pemakaian listrik yang terdapat dalam rumah tangga hanya dapat dilihat pada kWh meter yang dipasangkan oleh PT PLN pada setiap rumah. Alat tersebut tidak dapat mendeteksi alat apa saja yang digunakan, serta tidak dapat melihat konsumsi daya yang dikonsumsi pada alat elektronik rumah tangga yang sedang aktif. Metode *Recurrent Neural Network* (RNN) dapat mengidentifikasi jenis beban listrik pada perangkat elektronik rumah tangga agar penggunaan energi listrik lebih terpantau.

I. PENDAHULUAN

Pada zaman modern sektor energi listrik merupakan salah satu energi yang sangat dibutuhkan. Energi listrik digunakan pada alat rumah tangga, perkantoran, keamanan dan masih banyak lainnya. Hampir semua perangkat yang ada di kehidupan sehari-hari pasti memerlukan energi listrik. Alat elektronik yang biasa digunakan dalam menunjang kehidupan sehari-hari meliputi televisi, komputer, dispenser, kulkas, kipas, lampu, dsb. Namun setiap perangkat elektronik mempunyai daya listrik yang berbeda-beda. Identifikasi harga listrik adalah isu yang penting dalam persaingan industri tenaga [1]. Kompleksitas dalam identifikasi beban listrik rumah tangga terdapat pada volatilitas dan ketidakpastian pada beban listrik yang signifikan [2]. Perbedaan penggunaan daya listrik ini mengakibatkan perbedaan pembayaran biaya listrik bulanan pada setiap rumahnya.

Pengguna energi listrik akan selalu bertambah setiap tahun baik dalam kebutuhan rumah tangga, perkantoran maupun lainnya. Hal ini akan menyebabkan keseimbangan listrik yang mengalir akan berbeda-beda. Kebutuhan listrik

II. KAJIAN TEORI

A. Beban Listrik

Beban listrik adalah semua hal yang berhubungan dengan listrik yang ditanggung oleh pembangkit listrik, secara umum beban listrik berupa alat atau perangkat yang menggunakan energi listrik untuk sumber energi utama agar perangkat atau alat tersebut dapat berfungsi, contoh beban listrik pada kehidupan sehari-hari yaitu kipas angin, pemanas air dan setrika. Secara umum beban yang dilayani oleh sistem distribusi elektrik dibagi menjadi beberapa sektor yakni sektor perumahan, sektor industri, sektor komersial dan sektor usaha [6].

Pada keseluruhan sistem, total daya adalah semua daya aktif dan reaktif yang diperoleh dengan menjumlahkan daya dari tiap dipakai pada setiap alat elektronik yang ada di rumah. Pada beban seimbang jumlah daya yang dibangkitkan oleh generator tiga fasa atau daya yang diserap oleh beban tiga fasa, -tiap fasa [7]. Beban listrik dibagi menjadi tiga yaitu beban resesif, beban induktif, dan beban kapasitif. Dari jenis

beban diatas alat yang digunakan untuk percobaan ini menggunakan kipas, setrika, dan pemanas air.

B. Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) adalah jaringan saraf berulang atau jaringan saraf tiruan yang pemrosesannya dipanggil secara berulang untuk proses masukkan yang biasa adalah data sekuensial. Pada dasarnya *Neural Network* adalah otak dari sebuah *mechine learning*, dengan adanya jaringan ini maka *mechine learning* bisa mempelajari pola-pola dalam data secara mandiri. *Recurrent Neural Network* (RNN) disebut juga dengan RNNs karena melakukan tugas yang sama pada setiap elemennya, dan *output* yang bergantung pada komputasi sebelumnya. Dengan kata lain RNN memiliki sebuah memori yang merekam informasi yang telah dikalkulasikan. Secara teori RNNs mampu menggunakan informasi yang panjang dan beragam, akan tetapi pada prakteknya RNN terbatas untuk melihat informasi sebelumnya bahkan hanya sedikit (*only a few steps*).

C. Long Short Term Memory (LSTM)

Metode yang dilakukan pada pembuatan Tugas Akhir ini menggunakan RNN dengan *jenis Long -short Term Memory* (LSTM). Metode ini digunakan untuk mengidentifikasi jenis data yang dimiliki. Pada Tugas Akhir ini digunakan masing-masing 50 layer LSTM dan layer *embedding* dimana layer *embedding* didapatkan dari *dataset* atau data input. Training dilakukan menggunakan beberapa layer LSTM, kemudian dilanjutkan kedalam dense layer dengan pengaktifan ReLu yang dilanjutkan dengan nilai *dropout* 0,25.

D. Confusion Matrix

Confusion matrix adalah sebuah metode yang digunakan untuk menganalisis sebuah kelas yang digunakan untuk mengenali tuple dari kelas yang berbeda-beda. True Positive (TP) dan True Negative (TN) akan memberikan informasi ketika sebuah kelas bernilai benar dalam melabelkan sebuah data [21]. Sedangkan False Positif (FP) dan False Negative (FN) akan memberikan informasi ketika sebuah kelas salah dalam melabelkan sebuah data. Akurasi yakni nilai yang metrik untuk sebuah model klasifikasi yang mengukur jumlah prediksi yang benar sebagai peresentase dari jumlah dari total prediksi yang dibuat. Untuk dapat menghitung nilai akurasi adalah sebagai berikut [21].

$$Akurasi = \frac{TP+TN}{TP+FP+FN+TN} \dots \dots \dots (1)$$

Presisi berfungsi sebagai kepresisian atau kecocokan sebuah sistem antara permintaan sebuah informasi dengan jawaban yang ada. Persamaan untuk menghitung presisi adalah sebagai berikut [21].

$$Presisi = \frac{True\ positive}{True\ Positive+False\ Positive} \dots \dots \dots (2)$$

Recall berfungsi untuk menunjukkan tingkat keberhasilan atau spesifikasi agar mengetahui sebuah informasi secara benar tentang data sebuah kelas. Persamaan untuk menghitung nilai recall adalah sebagai berikut [21].

$$Recall = \frac{True\ Positive}{True\ Positive+False\ Negative} \dots \dots \dots (3)$$

F1-score bertujuan untuk mengkombinasikan antara matriks presisi dan *recall* menjadi satu matriks. Persamaan untuk f1-score dapat dilihat pada persamaan sebagai berikut [21].

$$F1 - score = \frac{2 \times Presisi \times Recall}{Presisi+Recall} \dots \dots \dots (4)$$

Keterangan :

TP : True Positive

TN : True Negative

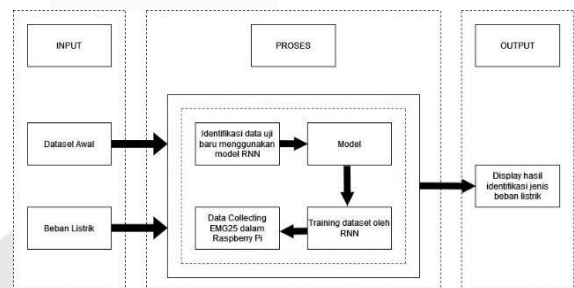
FP : False Positive

FN : False Negative

III. METODE

A. Desain Sistem

Desain sistem yang ada di dalam penelitian ini dirancang untuk memantau penggunaan listrik satu fasa yang ada di alat rumah tangga. Sistem ini dimulai dengan menghubungkan sumber listrik dengan *power meter digital* yang dapat menampilkan dan membaca aliran listrik yang sedang digunakan. Dari data yang diperoleh dikirim melalui Raspberry Pi 3 B+ dengan menggunakan RS-485 sebagai alat komunikasi serial. Setelah data dikirim melalui RS-485 barulah data dapat diunduh dari perangkat yang digunakan. Dari data yang sudah diperoleh dilakukan tahap *preprocessing* yang dapat mengidentifikasi jenis benda apa yang sedang berjalan.



GAMBAR 1
DIAGRAM FUNGSI SISTEM

B. Pengambilan Dataset

Pada penelitian ini, pengumpulan data beban listrik menggunakan *power meter* EMG 25. Data yang diambil oleh *power meter* diantaranya setrika, kipas dan pemanas air. Pengambilan data dilakukan dengan mengambil data selama 17 menit dengan rentang waktu satu detik setiap alatnya. Proses yang dilakukan pada masing-masing perangkat mengambil data setiap alat dan melakukan kombinasi setiap alat dengan jumlah data enam data kombinasi. Data yang diambil kemudian diolah menjadi *dataset* lalu akan dilanjutkan ke tahap identifikasi perangkat elektronik menggunakan metode *Recurrent Neural Network* (RNN). Langkah-langkah yang dilakukan pada pengujian sistem sebagai berikut.

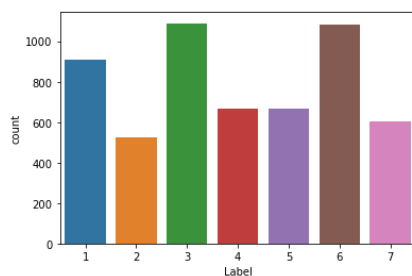
1. Memasang CT pada terminal yang kemudian dilanjutkan dengan pengujian.
2. Merangkai *power meter*, Raspberry Pi, dan Modbus

RS-485.

3. Mengaktifkan Raspberry Pi dan VNC Viewer.
4. Menghubungkan *port* usb dan mengatur *power meter* agar bisa menghubungkan dengan Raspberry Pi dan VNC Viewer.
5. Memonitor Raspberry Pi dengan VNC Viewer untuk pengambilan data dan mengirim program memastikan tidak ada error yang terjadi.
6. Pada kondisi pertama diambil data selama 17 menit dengan satu perangkat saja.
7. Pada kondisi kedua diambil data dari dua kombinasi perangkat yang digunakan diambil selama 17 menit.
8. Pada kondisi ketiga diambil dari kombinasi semua perangkat yang digunakan dan dilakukan selama 17 menit.
9. Hasil dari data yang telah dikumpulkan kemudian diolah menjadi *dataset* sesuai dengan fitur yang dibutuhkan.

IV. HASIL DAN PEMBAHASAN

A. Penggunaan SMOTE



GAMBAR 2

GRAFIK SEBELUM IMPLEMENTASI SMOTE

Berdasarkan pada gambar diatas jumlah dataset tidak seimbang. Ada nilai kelas data yang berada di orde pertama dan ada kelas data yang sudah memasuki orde kedua, maka dari itu digunakanlah SMOTE untuk menyamakan data pada masing – masing kelasnya. Pertama yang harus dilakukan adalah menampilkan data yang ada pada awal sebelum digunakan SMOTE, kemudian akan dilakukan pendekatan menggunakan *Over sampling* agar data yang ada tidak menjadi *overfit*. Pendekatan *over sampling* adalah pendekatan jumlah kelas yang lebih kecil menjadi sama dengan jumlah kelas yang lebih besar.

Dataset yang digunakan pada awal pengujian ini mempunyai jumlah 5541. Data dengan nilai terkecil ada di 450 dan tertinggi ada di 1089, maka dari itu digunakan metode SMOTE *oversampling*. Maka jumlah data pada masing-masing kelas akan dirubah menjadi 1089. Sehingga total jumlah data yang digunakan berjumlah 7623 data.

B. Skenario Pengujian Kinerja Sistem Dengan Partisi Data

Pengujian ini dilakukan untuk mengetahui kerja dari algoritma RNN dalam mengklasifikasikan data dalam kelas yang telah ditentukan sebelumnya. Pada percobaan ini diberikan data *testing* untuk menguji tabel keputusan yang sudah terbentuk. Akan diperoleh nilai *confusion matrix* dengan menghitung nilai *precision*, *recall*, dan *accuracy* dari hasil pengujian. Pengujian kinerja dilakukan sebanyak lima

kali dengan membagi data *training* dan *testing* dengan porsi yang berbeda-beda.

TABEL 1
PEMBAGIAN DATA TRAINING DAN DATA TESTING

Pengujian	Data Training %	Data testing (%)
Pengujian Pertama	50	50
Pengujian Kedua	60	40
Pengujian Ketiga	70	30
Pengujian Keempat	80	20
Pengujian Kelima	90	10

C. Skenario Pengujian *Learning Rate*

Dalam pengujian *Learning size* akan menggunakan partisi data dengan nilai *Accuracy*, *Precision*, *Recall*, dan *F1-score* yang tertinggi pada pengujian partisi data. Dan menggunakan *Batch size* sebesar 32 dengan nilai *epoch* sebesar 100. Kemudian pada pengujian *Learning rate* akan menggunakan nilai kelipatan 10 yang berawal dari 0.0001, 0.001, 0.01, 0.1, dan 1.

TABEL 2
PENGUJIAN *LEARNING SIZE*

Pengujian	Epoch	Batch size	Learning rate
Pengujian ke-1	100	32	0.0001
Pengujian ke-2	100	32	0.001
Pengujian ke-3	100	32	0.01
Pengujian ke-4	100	32	0.1
Pengujian ke-5	100	32	1

D. Skenario Pengujian *Batch Size*

Dalam melakukan pengujian *Batch size* akan menggunakan nilai dari hasil pengujian *Learning rate* dengan nilai *Accuracy*, *Precision*, *Recall*, dan *F1-score* yang tertinggi. Pengujian *batch size* akan dimulai berturut – turut dengan nilai sebesar 16, 32, 64, 128, dan 256.

TABEL 3
PENGUJIAN *BATCH SIZE*

Pengujian	Epoch	Batch Size	Learning Rate
Pengujian ke-1	100	8	Menggunakan <i>learning rate</i> dengan <i>Accuracy</i> , <i>Precision</i> , <i>Recall</i> , dan <i>F1-score</i> tertinggi pada pengujian <i>learning rate</i>
Pengujian ke-2	100	16	
Pengujian ke-3	100	32	
Pengujian ke-4	100	64	
Pengujian ke-5	100	128	

E. Skenario Pengujian Epoch

Dalam pengujian *epoch* menggunakan partisi yang mempunyai nilai *Accuracy*, *Precision*, *Recall*, dan *F1-score* tertinggi pada pengujian sebelumnya. Pengujian *Epoch* akan menggunakan nilai *Batch Size* 32 dan *Learning rate* 0,0001. Pengujian akan dilakukan berturut – turut dengan nilai *epoch* 50, 100, 150, 200, dan 250.

TABEL 4
PENGUJIAN EPOCH

Pengujian	Epoch	Batch size	Learning rate
Pengujian epoch ke-1	50	Menggunakan Batch Size dengan Accuracy, Precision, Recall, dan F1-score tertinggi pada pengujian Batch Size	Menggunakan learning rate dengan Accuracy, Precision, Recall, dan F1-score tertinggi pada pengujian learning rate
Pengujian epoch ke-2	100		
Pengujian epoch ke-3	150		
Pengujian epoch ke-4	200		
Pengujian epoch ke-5	250		

F. Pengujian Pertama

Pengujian pertama porsi yang diberikan untuk data Training dengan nilai 50% dan data Testing bernilai 50% dari 7623 dataset yang didapatkan. Hasil yang didapatkan dari perhitungan dengan jumlah kelas tujuh untuk masing-masing nilai Accuracy, Precision, Recall dan f1-score adalah sebesar 97,37%, 98,06%, 96,90%, 97,41%. Dari hasil partisi data tersebut didapatkan nilai hasil partisi data dengan tingkat akurasi tertinggi yaitu pada partisi dengan data 70%:30% dengan nilai akurasi 97,83%.

TABEL 5
KINERJA PENGUJIAN SISTEM PERTAMA

Perbandingan	Precision	Recall	Accuracy	F1 Score
50% : 50%	98,06%	96,90%	97,37%	97,41%
60% : 40%	98,30%	97,32%	97,78%	97,76%
70% : 30%	98,32%	97,34%	97,83%	97,77%
80% : 20%	98,31%	97,27%	97,75%	97,72%
90% : 10%	97,63%	96,74%	97,12%	97,11%

G. Pengujian Kedua

Pada pengujian kedua ini dilakukan menggunakan nilai learning rate yang berbeda-beda seperti tabel 4.3. Pengujian pada learning rate menggunakan hasil dari pengujian perbandingan data train dan dataset yang memiliki nilai akurasi terbaik, kemudian akan diuji menggunakan nilai learning rate yang berbeda-beda dengan kelipatan 10 kali. Hasil terbaik yang diperoleh dari pengujian ini berada pada nilai 0.001 dengan nilai Accuracy, Precision, Recall, dan F1-score adalah 97.97%, 98.49%, 97.51%, dan 98.00%. Nilai pada setiap pengujian learning rate sebagai berikut.

TABEL 6
KINERJA PENGUJIAN SISTEM KEDUA

Learning Rate	Precision	Recall	Accuracy	F1 Score
0.0001	98,43%	97,45%	97,92%	97,94%
0.001	98,50%	97,51%	97,97%	98,00%
0.01	2,65%	14,26%	18,58%	4,48%
0.1	2,75%	14,29%	19,26%	4,61%
1	2,75%	14,29%	19,26%	4,61%

H. Pengujian Ketiga

Pengujian ketiga ini adalah pengujian batch size, data yang sudah melawati pengujian kedua dan mendapatkan nilai akurasi tertinggi maka akan dilanjutkan ke dalam pengujian ini. Pengujian batch size berawal dari 16 dan seterusnya sampai dengan nilai 256. Pada pengujian ini dihasilkan nilai tertingginya berada pada saat 32. Nilai Accuracy, Precision,

Recall, dan F1-score adalah 97.95%, 98.46%, 97.46%, dan 97.90%. Hasil pengujian pada learning rate adalah sebagai berikut.

TABEL 7
KINERJA PENGUJIAN SISTEM KETIGA

Batch Size	Precision	Recall	Accuracy	F1 Score
16	98,25%	97,23%	97,71%	97,68%
32	98,46%	97,46%	97,95%	97,90%
64	98,30%	97,28%	97,77%	97,74%
128	98,38%	97,39%	97,89%	97,82%
256	98,65%	97,07%	97,65%	97,57%

I. Pengujian Keempat

Pada pengujian keempat yaitu adalah menggunakan nilai epoch. Pengujian ini menggunakan nilai data yang sudah melalui pengujian pertama, kedua serta ketiga. Nilai terbaik pada pengujian ketiga merupakan hasil akhir yang akan digunakan untuk melakukan pengujian keempat ini. Pada pengujian epoch akan dimulai dari nilai 50 menggunakan kelipatan 50 dan berakhir pada saat epoch bernilai 250. Hasil terbaik yang dilakukan pada pengujian ini berada pada saat pengujian menggunakan nilai epoch 250. Berikut adalah hasil dari pengujian epoch.

TABEL 8
KINERJA PENGUJIAN SISTEM KEEMPAT

Epoch	Precision	Recall	Accuracy	F1 Score
50	98,21%	97,07%	97,65%	97,56%
100	98,41%	97,41%	97,89%	97,85%
150	98,37%	97,36%	97,83%	97,81%
200	98,38%	97,39%	97,89%	97,82%
250	98,39%	97,39%	97,90%	97,83%

V. KESIMPULAN

Pada Tugas Akhir ini memiliki kesimpulan yang ditarik dari hasil penelitian dan pengujian, kesimpulan yang didapatkan pada Tugas Akhir sebagai berikut.

1. Sistem dapat melakukan deteksi jenis beban listrik dengan alat yang diuji yaitu kipas angin, setrika, dan pemanas air.
2. Pengujian pertama yang dilakukan menggunakan perbandingan data training dan dataset menunjukkan hasil terbaik pada saat perbandingannya bernilai 70%:30% dengan nilai akurasi 97.83%, presisi 98.32%, recall 97.34%, dan f1-score 97.83%.
3. Pengujian kedua menggunakan pengujian learning rate dengan menggunakan kelipatan 10, data terbaik yang dihasilkan pada pengujian ini ketika nilai learning rate berada pada 0.001 dengan nilai akurasi 97.97%, presisi 98.49%, recall 97.51%, dan f1-score 98.00%.
4. Pada pengujian ketiga menggunakan nilai batch size, nilai terbaik yang dihasilkan pada pengujian ini ketika berada pada nilai batch size 32 dengan nilai akurasi 97.95%, presisi 98.46%, recall 97.46%, dan f1-score 97.96%.
5. Pengujian hyperparameter terbaik berada pada saat pengujian epoch 250 menunjukkan hasil nilai akurasi 97.95%, presisi 98.46%, recall 97.46%, dan f1-score 97.96%.

REFERENSI

- [1] D. Paras Mandal, "Electricity Price Forecasting Using Neural Networks and Similar Days," *IEEE*, vol. 1, p. 1, 2017.
- [2] M. X. D. Heng Shi, "Deep Learning for Household Load Forecasting—A Novel Pooling Deep RNN," *IEEE*, vol. 9, p. 1, 2018.
- [3] D. T. D. Prita Ayuningtyas, "Prediksi Beban Listrik Pada PT. PLN (Persero) Menggunakan Regresi Interval Dengan Neural Fuzzy," *Jurnal Coding*, p. 1, 2019.
- [4] D. Lisiani, "Identifikasi dan Analisis Jenis Beban Listrik Rumah Tangga," *Tugas Akhir*, p. 5, 2019.
- [5] B. Y. D. MINH Tu Hoang, "Recurrent Neural Networks For Accurate RSSI Indoor Localization," *IEEE*, vol. 1, p. 3, 2019.
- [6] U. B. Nusantara, "Magister Of Information System Management," Universitas Bina Nusantara, 26 November 2019. [Online]. Available: <https://mmsi.binus.ac.id/2019/11/26/apakah-deep-learning/>. [Accessed 12 Desember 2021].
- [7] D. J. D. Tao Yao, "Load Prediction of Microgrid Optimal Operation Based on Improved Algorithm in Machine Learning," *Mechatronics and Applied Mechanics*, p. 3, 2020.
- [8] M. M. A. S. A. H. G. N. Y. M. D. Despa, "Monitoring dan Manajemen Energi Listrik Gedung Laboratorium Berbasis Internet of Things (IoT)," in *Scholar*, Lampung, 2018.
- [9] Z. Arifin, "Pengaruh Pembebanan Terhadap Arus Motor EG-530AD-2F," *Disprotek*, vol. 10, no. 1, pp. 43-49, 2019.
- [10] A. R. a. K. M. E. Desyantoro, "Sistem Pengendali Peralatan Elektronik dalam Rumah secara Otomatis Menggunakan Sensor PIR, Sensor LM35, dan Sensor LDR," *J. Teknol dan Sist Komput*, vol. 3, no. 3, pp. 405-411, 2015.
- [11] G. M. V. Y. S. M. G. I. M. P Arun Chandra, "Automated Energy Meter Using Wifi Enabled Rasberry Pi," in *IEEE*, India, 2016.
- [12] Y. K. S. P. A.S. Salunkhe, "Data Retrieval from RS-485 Implemented Energy Meter for Industrial Monitoring System," in *IEEE*, India, 2022.
- [13] J. D. Kelleher, *Deep Learning*, London: The MIT Press, 2019.
- [14] A. T. Mayank Dixit, "An overview of deep learning architectures, libraries and its applications areas," *IEEE*, p. 2, 2018.
- [15] F. Istardi, "Prediksi Beban Listrik Pulau Bali Dengan menggunakan Metode Backpropagasi," p. 1, 2012.
- [16] R. Siringoringo, "Klasifikasi Data Tidak Seimbang Menggunakan Algoritma SMOTE dan K-Nearest Neighbor," *Jurnal ISD*, vol. 3, no. 1, pp. 44-49, 2018.
- [17] V. A. L. A. F. S. D. R. A. E. S. Arwan, "Binus University," Binus University Graduate Program Master, 8 June 2018. [Online]. Available: <https://mti.binus.ac.id/2018/06/08/synthetic-minority-over-sampling-technique-smote-algorithm-for-handling-imbalanced-data/>. [Accessed 6 August 2022].
- [18] D. S. T. N. Y. K. N. Y. X. H. Zhaozhao Xu, "A Cluster-based Oversampling Algorithm Combining SMOTE and K-means for Imbalanced Medical Data," *Information Sciences*, vol. 572, pp. 574-589, 2021.
- [19] S. K. D. Tomáš Mikolov, "Extensions of recurrent neural network language model," *IEEE*, p. 1, 2011.
- [20] J. S. F. C. Felix A. Gers, "Learning to Forget: Continual Prediction with LSTM," *IEEE*, vol. 12, no. 10, p. 2451, 2020.
- [21] M. K. a. J. P. J. Han, "Data Mining Concepts and Techniques," in *Elsevier*, USA, 2011.