TUGAS AKHIR – EE 184801

IDENTIFIKASI BEBAN PADA SISTEM TEGANGAN RENDAH BERBASIS HARMONISA DAN *IOT* DENGAN *ARTIFICIAL NEURAL NETWORK* SECARA REAL TIME

Irfan Purwito Nugroho
NRP 07111340000113

Dosen Pembimbing
Ardyono Priyadi, S.T., M.Eng., Dr.Eng
Dr. Ir. Margo Pujiantara, M.T.

DEPATERMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

TUGAS AKHIR – EE 184801

IDENTIFIKASI BEBAN PADA SISTEM TEGANGAN RENDAH BERBASIS HARMONISA DAN *IOT* DENGAN *ARTIFICIAL NEURAL NETWORK* SECARA REAL TIME

Irfan Purwito Nugroho
NRP 07111340000113

Dosen Pembimbing
Ardyono Priyadi, S.T., M.Eng., Dr.Eng
Dr. Ir. Margo Pujiantara, M.T.

DEPATERMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

FINAL PROJECT – EE 184801

LOAD IDENTIFICATION AT LOW VOLTAGE USING HARMONIC AND IOT NEURAL NETWORK BASED IN REAL TIME

Irfan Purwito Nugroho
NRP 07111340000113

Supervisor(s)
Ardyono Priyadi, S.T., M.Eng., Dr.Eng
Dr. Ir. Margo Pujiantara, M.T.

ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Intelligent Electrical and Informatics Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2020

PERNYATAAN KEASLIAN
TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "**Identifikasi Beban pada Sistem Tegangan Rendah Berbasis Harmonisa dan IoT dengan Artificial Neural Network secara Realtime**" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juni 2020

Irfan Purwito Nugroho
NRP. 0711 13 4000 0113

IDENTIFIKASI BEBAN PADA SISTEM TEGANGAN RENDAH BERBASIS HARMONISA DAN IOT DENGAN ARTIFICIAL NEURAL NETWORK SECARA REAL TIME

TUGAS AKHIR

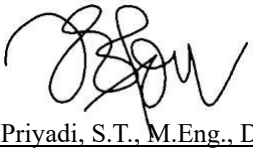
Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik

Pada

Bidang Studi Sistem Tenaga
Departemen Teknik Elektro
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing I



Ardyono Priyadi, S.T., M.Eng., Dr.Eng.
NIP. 197309271998031004

SURABAYA
JULI, 2020

IDENTIFIKASI BEBAN PADA SISTEM TEGANGAN RENDAH BERBASIS HARMONISA DAN IOT DENGAN ARTIFICIAL NEURAL NETWORK SECARA REAL TIME

TUGAS AKHIR

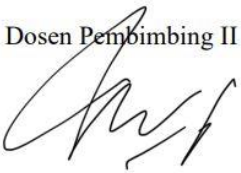
Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik

Pada

Bidang Studi Sistem Tenaga
Departemen Teknik Elektro
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing II



Dr. Ir. Margo Pujiantara, M.T.
NIP. 196603181990101001

SURABAYA
JULI, 2020

Halaman ini sengaja dikosongkan

IDENTIFIKASI BEBAN PADA SISTEM TEGANGAN RENDAH BERBASIS HARMONISA DAN IOT DENGAN ARTIFICIAL NEURAL NETWORK SECARA REAL TIME

Nama : Irfan Purwito Nugroho
Pembimbing 1 : Ardyono Priyadi, S.T., M.Eng., Dr.Eng.
Pembimbing II : Dr. Ir. Margo Pujiantara, MT.

ABSTRAK

Semakin banyaknya penggunaan beban non linier pada sistem tenaga listrik dapat menimbulkan berbagai masalah. Masalah yang sering terjadi antara lain meningkatnya rugi-rugi daya, meningkatnya temperatur konduktor kabel, adanya arus pada fasa netral, salah ukur pada KWH-meter, dan pendeknya usia peralatan listrik. Di sisi lain penggunaan peralatan listrik yang berlebihan sering dilakukan oleh pelanggan PLN. Satu diantara penyebab pemborosan penggunaan listrik ini adalah tidak termonitornya kondisi beban terpasang. Oleh karena itu, perlu dikembangkan peralatan monitoring beban yang bisa diakses secara realtime.

Dalam tugas akhir ini dikembangkan simulasi model identifikasi beban yang terdiri dari *microcontroller*, sensor arus, dan model *neural network* yang kedepannya dapat digunakan untuk membantu dalam pembuatan alat monitoring beban secara *realtime*. Metode yang digunakan dalam pengembangan simulasi identifikasi dan monitoring ini adalah dengan mendeteksi profil harmonisa arus dari setiap jenis beban dan dilakukan identifikasi beban melalui model *artificial neural network*. Dilakukan beberapa percobaan konfigurasi parameter model ANN sehingga didapatkan model dengan keakuratan dan keefisienan tertinggi.

Hasil dari tugas akhir ini diharapkan ditemukan konfigurasi ANN dengan keakuratan paling tinggi. Konfigurasi tersebut kedepannya bisa dimanfaatkan dalam pembuatan alat monitoring dan identifikasi beban berbasis harmonisa serta neural network secara *realtime*.

Kata kunci : Harmonisa, *Artificial Neural Network*, *Multi Layer Perceptron*, Python, *Internet of Things*

Halaman ini sengaja dikosongkan

LOAD IDENTIFICATION AT LOW VOLTAGE USING HARMONIC AND IOT NEURAL NETWORK BASED IN REAL TIME

Nama **: Irfan Purwito Nugroho**
1st Advisor **: Ardyono Priyadi, S.T., M.Eng., Dr.Eng.**
2nd Advisor **: Dr. Ir. Margo Pujiantara, MT.**

ABSTRACT

The increasing amount of non-linear loads on the electric power system can cause various problems. Problems that often occur include power loss, temperature rise of the conductor cable, the presence of current in the neutral phase, a measurement error on the KWH-meter, and the short lifetime of the electrical equipment. On the other hand, the over limit usage on electrical equipment is more likely done by PLN customers. One of the causes of this wasteful use of electricity is that the installed load conditions are not monitored. Therefore, it is necessary to develop a load monitoring equipment that can be accessed in real time.

In this final project a simulation model is developed that consists of a microcontroller, current sensor, and a neural network model which can be used in the future to assist in the production of load monitoring tools in real time. The method that being used in the development of this identification and monitoring simulation is to detect the current's harmonisa profile of each load type and load identification is carried out using an artificial neural network model. Several configuration experiments on parameter of ANN model are conducted in order to acquire the highest accuracy and efficiency.

The results of this final project are expected to find the highest accuracy for ANN configuration. In the future it can be used in production of monitoring tools and harmonisa based load identification as well as neural network based load in real time.

Keywords: Harmonics, Artificial Neural Network, Multi Layer Perceptron, Python, Internet of Things

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji dan syukur hanya milik Allah, Tuhan semesta alam. Shalawat serta salam semoga selalu tercurahkan kepada Rasulullah Muhammad, *shallallahu 'alaihi wa sallam*. Atas karunia dan kehendak-Nya, saya dapat menyelesaikan tugas akhir dengan baik dan tanpa kendala apapun.

Buku tugas akhir berjudul "**Identifikasi Beban pada Sistem Tegangan Rendah Berbasis Harmonisa dan IoT dengan Artificial Neural Network secara Realtime**" ini disusun untuk memenuhi syarat kelulusan pendidikan sarjana pada bidang studi Sistem Tenaga, Departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember Surabaya.

Penyelesaian tugas akhir ini tentunya tidak terjadi begitu saja. Dengan segala kerendahan hati saya mengucapkan terima kasih kepada :

1. Kedua orang tua saya yang selalu sabar dan memberikan support, doa, serta materi kepada saya.
2. Istri saya, drh. Bella Fortuna, yang selalu memotivasi saya ketika kehilangan gairah untuk menyelesaikan tugas akhir. Serta anak saya Rayya Salsabila Maryam, yang senyumnya adalah bahan bakar terbaik saya dalam menjalani hari-hari.
3. Bapak Ardyono Priyadi, S.T., M.Eng., Dr.Eng. dan Bapak Dr. Ir. Margo Pujiantara, MT., selaku dosen pembimbing yang senantiasa memberikan ilmu, waktu, dan bimbingan kepada saya untuk menyelesaikan tugas akhir ini.
4. Bapak Dr. Ir. Anang Tjahjono, MT., ayah dari sahabat saya. Naufal Rasyiq, yang tak henti-hetinya saya repoti untuk saya mintai saran serta masukan. Semoga Allah balas dengan kebaikan yang jauh lebih besar, Pak.
5. Lutfi Rusidi, ST., sahabat saya yang tak terhingga jasanya untuk penelitian tugas akhir ini.
6. Teman-teman Pelangi Perjuangan, e53, JMMI Integrasi, dan pihak-pihak lain yang membantu saya dalam mengerjakan tugas akhir ini yang tidak bisa saya sebutkan dalam satu halaman kertas ini.

Akhir kata, penulis berharap penelitian dalam tugas akir ini ada manfaatnya.

Surabaya, Juni 2020

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

HALAMAN JUDUL
LEMBAR PERNYATAAN
LEMBAR PENGESAHAN
ABSTRAK i
ABSTRACT iii
KATA PENGANTAR v
DAFTAR ISI vii
DAFTAR GAMABR ix
DAFTAR TABEL xi
BAB 1 PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Perumusan Masalah 2
1.3 Tujuan 2
1.4 Batasan Masalah 2
1.5 Metodologi 3
1.6 Sistematika Penulisan 3
1.7 Manfaat 4
BAB 2 HARMONISA DAN BACKPROPAGATION ANN 5
2.1 Harmonisa 5
2.1.1 *Total Harmonic Distortion* 5
2.2 *Artificial Neural Network* *8*
2.2.1 Fungsi Aktivasi 8
2.2.2 *Epochs* 9
2.2.3 Backpropagation 9
2.3 *Microcontroller STM32F7* 12
2.4 *Internet of Things* 13
2.5 Cara Kerja *IoT* 14
BAB 3 PERANCANGAN DAN PEMODELAN NN 15
3.1 Sistem Identifikasi Beban 15
3.2 Perencanaan 15
3.2.1 Pengambilan Data 15
3.2.2 Pemodelan 18
3.2.3 Preparing Data 19
3.2.4 Klasifikasi 20
3.2.5 Monitoring Beban Listrik yang Aktif 20

BAB 4 SIMULASI DAN ANALISIS DATA 21
4.1 Arsitektur Neural Network 21
4.1.1 Input Layer 21
4.1.2 Hidden Layer 21
4.1.3 Output Layer 21
4.2 Konfigurasi Parameter Neural Network 21
4.3 Tahap Learning dan Validasi ANN 22
4.3.1 Learning dengan Variasi Fungsi Aktivasi 23
4.3.2 Learning dengan Variasi Jumlah Epochs 29
4.3.3 Learning dengan Variasi Jumlah Neuron Hidden Layer 37
4.3.4 Learning dengan Variasi Jumlah Hidden Layer 44
BAB 5 PENUTUP 51
5.1 Kesimpulan 51
5.2 Saran 51
DAFTAR PUSTAKA 53
DAFTAR RIWAYAT HIDUP 55

DAFTAR GAMBAR

Gambar 2.1 Arsitektur NN *Backpropagation* 19
Gambar 2.2 *Microcontroller* STM32F7 11
Gambar 2.3 Implementasi *IoT* pada kehidupan 14
Gambar 3.1 Bagan pengambilan data harmonisa 16
Gambar 3.2 Contoh tampilan *file xls* pengambilan data harmonisa 18
Gambar 3.3 Pemodelan *neural network* dengan 9 buah *input* 19

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 2.1 Batas arus harmonisa maksimal pada sistem distribusi (120 - 69.000V) dalam persen I_L 7
Tabel 2.2 Batas arus harmonisa maksimal pada sistem *subtransmission* (69.001 - 161.000V) dalam persen I_L 7
Tabel 2.3 Batas arus harmonisa maksimal pada sistem transmisi (lebih dari 161.000V) dalam persen I_L 8
Tabel 2.4 Analogi jaringan saraf manusia dengan ANN 8
Tabel 3.1 31 Kombinasi beban 17
Tabel 4.1 Hasil pengukuran harmonisa lampu hemat energi 35 kali 22
Tabel 4.2 Hasil pembelajaran dan tes validasi dengan fungsti aktivasi Linier 23
Tabel 4.3 Hasil pembelajaran dan tes validasi dengan fungsti aktivasi Sigmoid 24
Tabel 4.4 Hasil pembelajaran dan tes validasi dengan fungsti aktivasi Tanh 25
Tabel 4.5 Hasil pembelajaran dan tes validasi dengan fungsti aktivasi ReLU 26
Tabel 4.6 Perbandingan nilai akurasi rata-rata tiap fungsi aktivasi 28
Tabel 4.7 Hasil pembelajaran dan tes validasi dengan jumlah *epochs* 10 29
Tabel 4.8 Hasil pembelajaran dan tes validasi dengan jumlah *epochs* 50 30
Tabel 4.9 Hasil pembelajaran dan tes validasi dengan jumlah *epochs* 100 31
Tabel 4.10 Hasil pembelajaran dan tes validasi dengan jumlah *epochs* 200 33
Tabel 4.11 Hasil pembelajaran dan tes validasi dengan jumlah *epochs* 1000 34
Tabel 4.12 Hasil pembelajaran dan tes validasi dengan jumlah *epochs* 2000 35
Tabel 4.13 Perbandingan nilai akurasi rata-rata tiap jumlah epochs 36
Tabel 4.14 Hasil pembelajaran dan tes validasi dengan jumlah neuron 9-10-1 38
Tabel 4.15 Hasil pembelajaran dan tes validasi dengan jumlah

neuron 9-15-1 39
Tabel 4.16 Hasil pembelajaran dan tes validasi dengan jumlah neuron 9-35-1 40
Tabel 4.17 Hasil pembelajaran dan tes validasi dengan jumlah neuron 9-50-1 41
Tabel 4.18 Perbandingan nilai akurasi rata-rata tiap jumlah neuron *hidden layer* 42
Tabel 4.19 Hasil Pembelajaran dan tes validasi dengan konfigurasi neuron 9-10-1 44
Tabel 4.20 Hasil pembelajaran dan tes validasi dengan konfigurasi neuron 9-10-10-1 45
Tabel 4.21 Hasil pembelajaran dan tes validasi dengan konfigurasi neuron 9-10-10-10-1 46
Tabel 4.22 Perbandingan nilai akurasi rata-rata tiap jumlah *hidden layer* 47

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Meningkatnya penggunaan beban pada sistem tenaga listrik (rumah tangga dan industri), diikuti oleh semakin banyaknya penggunaan peralatan elektronik. Peralatan-peralatan elektronik merupakan sumber harmonisa utama pada sistem kelistrikan PLN. Harmonisa merupakan salah satu indikator yang menunjukkan kualitas daya listrik.

Dampak harmonisa yang sering dijumpai antara lain meningkatnya temperatur pada kawat konduktor kabel, adanya distorsi tegangan, salah ukur pada KWH-meter, meningkatnya rugi-rugi daya, adanya arus pada fasa netral, dan menurunnya *lifetime* alat-alat listrik [2].

Penggunaan peralatan listrik penghasil harmonisa banyak terdapat pada jaringan distribusi, termasuk beban-beban rumah tangga di dalamnya. Pada standar IEEE 519.1992 (*Recommended Practices and Requirements for Harmonic Control in Electrical Power System*) ditetapkan rekomendasi batasan level yang diterima dari harmonisa arus dalam sistem tenaga listrik. Didalam standar tersebut disebutkan bahwa pelanggan listrik memiliki tanggung jawab untuk menjaga level harmonisa agar tetap rendah. Namun pada praktiknya dapat dengan mudah ditemukan pemborosan penggunaan peralatan listrik oleh pelanggan PLN. Hal ini salah satunya disebabkan oleh kurangnya monitoring dan pengontrolan terhadap alat listrik yang sedang menyala atau aktif dalam suatu sistem. Sebagai contoh, kerap ditemukan AC atau lampu yang terus menyala pada ruang-ruang kelas di Departemen Teknik Elektro ITS, padahal ruangan tersebut tidak sedang digunakan.

Oleh karena itu, dibutuhkan suatu metode monitoring harmonisa beban untuk mencegah segala dampak buruk yang bisa terjadi dikarenakan harmonisa berlebih pada suatu sistem listrik. Monitoring beban dalam penelitian tugas akhir ini dikembangkan dalam bentuk simulasi yang terdiri dari *microcontroller* sebagai piranti pengukur arus dan tegangan harmonisa, pemrograman di python sebagai alat pemodelan neural network, dan tampilan antarmuka software Python sebagai tempat meletakkan hasil *output* identifikasi beban. Nilai amplitudo harmonisa orde ganjil dari 1 sampai orde ke-17 digunakan untuk mengidentifikasi profil harmonisa beban.

Alasan khusus dari diusulkannya topik ini adalah untuk menyempurnakan penelitian yang telah dilakukan sebelumnya [1]. Hasil dari penelitian pada tugas akhir ini diharapkan simulasi berbasis model *neural network* ini dapat mengidentifikasi beban dengan keakuratan tinggi dan pengguna dapat memonitor kondisi beban (khususnya harmonisa) yang ada dalam sistem listrik secara *realtime*.

1.2 Perumusan Masalah

Dari latar belakang yang sudah dipaparkan, permasalahan yang dibahas dalam usulan tugas akhir ini adalah :

1. Cara melakukan identifikasi beban mengunakan mode *backpropagation multilayer perceptron Artificial Neural Network*
2. Cara mengonfigurasikan parameter *backpropagation* NN sehingga didapatkan akurasi terbaik
3. Cara membuat simulasi identifikasi beban berbasis *backpropagation MLP ANN* dan *IoT* secara realtime

1.3 Tujuan Tugas Akhir

Sesuai dengan latar belakang permasalahan yang telah diuraikan, maka tujuan tugas akhir ini adalah mengembangkan simulasi identifikasi beban menggunakan metode *backpropagation artificial neural network* berbasis *IoT*.

1.4 Batasan Masalah

Agar tugas akhir ini bahasannya fokus dan tidak melebar, maka batasan masalah yang berlaku antara lain :

1. Beban yang digunakan adalah beban nonlinear pada sistem tegangan rendah seperti televisi, *charger* laptop, *air conditioner* (AC), lampu, dan kipas angin.
2. Pengambilan data harmonisa beban menggunakan *microcontroller* STM32F7
3. Komputasi *ANN* menggunakan *software* Python 3.6 pada sistem operasi Windows 10 Enterprise.

1.5 Metodologi

Pada tahap pertama, dilakukan pengambilan data harmonisa tiap jenis beban menggunakan *microcontroller* yang telah diprogram untuk menjadi alat ukur arus dan tegangan. Jenis beban yang diidentifikasi pada penelitian ini adalah beban rumah tangga seperti televisi, *AC*, komputer, penanak nasi, dan sebagainya. Hasil dari analisis tersebut kemudian diuji menggunakan *power quality analyzer*. Apabila akurasi yang didapatkan tidak baik maka dilakukan perbaikan pada alat ukurnya.

Tahap keduanya adalah pembuatan *backpropagation neural network* dari data yang sudah didapatkan. Pembuatan dan pembelajaran *neural network* menggunakan *software* Python. Input *backpropagation neural network* adalah amplitudo harmonisa frekuensi kelipatan ganjil beban. Sedangkan outputnya adalah jenis beban yang sedang aktif.

Tahap ketiga yaitu dilakukan percobaan berbagai konfigurasi ANN sehingga ditemukan yang tingkat keakuratannya paling tinggi.

Jika pembuatan program *backpropagation neural network* berhasil dan performa terbaik telah ditemukan, maka dilanjutkan ke tahap selanjutnya yaitu pengujian. Pada tahap ini, dilakukan pengujian identifikasi beban dari hasil ukur harmonisa peralatan yang sedang aktif dengan program ANN yang telah dibuat.

Tahap terakhir adalah pembuatan simulasi alat identifikasi beban berbasis ANN secara realtime dengan software Python.

1.6 Sistematika

Tugas akhir ini terdiri dari lima bab yang disusun dengan sistematika sebagai berikut :

Bab 1 : Pendahuluan

Dalam bab ini dijelaskan mengenai latar belakang pengambilan topik, perumusan masalah, manfaat dan tujuan, pembatasan masalah, dan sistematika penulisan.

Bab 2 : Teori Penunjang

Dalam Bab II dijelaskan materi-materi yang berhubungan dengan topik tugas akhir yang penulis ambil, seperti harmonisa, neural network, dan *microcontroller*.

Bab 3 : Pengumpulan dan Pengolahan Data

Pada bab ini dibahas pengolahan data harmonisa yang selanjutnya digunakan sebagai bahan input training pada *backpropagation* NN.

Bab 4 : Simulasi dan Pembahasan
Didalam bab ini dibahas perbandingan hasil training *backpropagation* NN dengan berbagai konfigurasi.

Bab 5 : Penutup
Berisi kesimpulan dan saran atas hasil yang diperoleh dari tugas akhir ini.

1.7 Manfaat Tugas Akhir

Manfaat yang diharapkan dari tugas akhir ini antara lain :

1. Mengetahui pengertian harmonisa, penyebab, dan dampak yang ditimbulkannya.
2. Memahami salah satu algoritma pada kecerdasan buatan (*backpropagation MLP ANN*) dan implementasinya pada sistem tenaga listrik tegangan rendah.
3. Mengetahui cara membuat simulasi identifikasi beban berbasis *backpropagation MLP ANN* dan *IoT*.

BAB 2
HARMONISA DAN *BACKPROPAGTION NN*

2.1 Harmonisa

Harmonisa merupakan salah satu gangguan utama yang terjadi pada sistem tenaga listrik. Harmonisa pada sistem tenaga listrik didefinisikan sebagai sinyal sinusoidal tegangan dan arus yang memiliki frekuensi kelipatan bilangan bulat dari frekuensi fundamentalnya. Contohnya, jika frekuensi fundamental sebuah sistem listrik alternating current (AC) adalah 50 Hz, maka harmonisa keduanya memiliki frekuensi 100 Hz, harmonisa ketiganya berfrekuensi 150 Hz, dan seterusnya.

Penyebab utama harmonisa pada sistem tenaga listrik adalah penggunaan beban nonlinear seperti peralatan konverter daya, arcing device, dan perangkat yang bekerja dengan prinsip elektromagnetik seperti transformator serta motor [2].

Arus harmonisa pada sistem tenaga listrik mengakibatkan berbagai macam kerugian. Kerugian tersebut seperti bertambahnya rugi-rugi serta overheating pada kapasitor, transformator, dan motor [3]. Gangguan interferensi gelombang radio juga bisa disebabkan oleh adanya harmonisa pada sistem listrik.

Pada analisis distorsi gelombang yang diakibatkan oleh harmonisa terdapat istilah yang dinamakan *Total Harmonics Distortion* (THD). Semakin tinggi nilai THD, semakin besar pula bahayanya pada sistem tenaga listrik.

2.1.1 Total Harmonic Distortion

Total harmonic distortion (THD) merupakan satu dari indikator yang paling umum digunakan untuk mengukur tingkat harmonisa pada suatu sistem dan instalasi listrik. THD dapat menunjukkan tingkat distorsi atau gangguan pada arus dan tegangan yang bekerja di sistem. Oleh karena itu, THD menjadi faktor penting pada sistem audio, telekomunikasi, dan sistem tenaga listrik.

Total harmonic distortion dapat didefinisikan sebagai rasio dari tegangan root mean square (RMS) seluruh frekuensi harmonisa dibandingkan dengan nilai RMS tegangan fundamental. Nilai THD sering kali disebutkan dalam persen.

Berikut ini persamaan THD tegangan :

$$THDv = \frac{\sqrt{\sum_{n=2}^{K} v^2{}_{n_rms}}}{V_{1_rms}}$$

Keterangan :
$THDv$ adalah besar *total harmonic distortion* tegangan pada sistem
V_{n_rms} merupakan besar tegangan RMS harmonisa pada orde ke-*n*
K adalah orde harmonisa kelipatan tertinggi
V_{1_rms} Adalah tegangan *root mean square* fundamental

Tingkat THD dalam persen dapat ditemukan dengan mengalikan hasil THD persamaan di atas dengan seratus persen. Yaitu :

$$THDv = \frac{\sqrt{\sum_{n=2}^{K} v^2{}_{n_rms}}}{V_{1_rms}} \; x\; 100\%$$

Persamaan yang sama dapat juga digunakan untuk menghitung besar THD arus pada sistem tenaga listrik. Persamaan tersebut yaitu :

$$THDi = \frac{\sqrt{\sum_{n=2}^{K} i^2{}_{n_rms}}}{i_{1_rms}} \; x\; 100\%$$

Keterangan :
$THDi$ merupakan besar *total harmonic distortion* arus pada sistem
i_{n_rms} merupakan besar arus RMS harmonisa pada orde ke-*n*
K adalah orde harmonisa kelipatan tertinggi
i_{1_rms} Adalah arus *root mean square* fundamental

Tingkat THD yang rendah pada sistem memiliki banyak keuntungan seperti faktor daya (*power factor*) yang tinggi, arus puncak yang rendah, dan tingkat efisiensi yang tinggi. Sedangkan, pada sistem telekomunikasi tingkat THD yang rendah bisa menjadikan sistem tidak mudah terinferensi dengan perangkat lain.

Satu standar lain yang biasa digunakan untuk mengukur tingkat distorsi pada suatu sistem adalah *total demand distortion* (TDD). TDD

khusus dipakai untuk mengukur tingkat harmonisa arus pada sistem tenaga listrik. Persamaan TDD seperti di bawah ini :

$$TDD = \frac{\sqrt{\sum_{n=2}^{K} i^2{}_{n_rms}}}{i_{max_rms}} \; x \; 100\%$$

Keterangan :
TDD merupakan *total demand distortion* pada sistem
i_{n_rms} merupakan besar arus RMS harmonisa pada orde ke-*n*
K adalah orde harmonisa kelipatan tertinggi
i_{max_rms} Adalah arus *root mean square* maksimal rata-rata pada sistem

Pada beberapa kasus, pengukuran TDD lebih akurat dibandingkan dengan THD untuk mengukur tingkat distorsi arus sistem.

2.1.2 Standar Distorsi Arus

Batas distorsi arus harmonisa disebutkan dalam IEEE std. 519-1992. Batas tersebut seperti dalam tabel di bawah ini. Isc

Tabel 2.1 Batas arus harmonisa maksimal pada sistem distribusi (120 - 69.000V) dalam persen I_L.

I_{sc}/I_L	<11	11≤h<17	17≤h<23	23≤h<35	35≥	TDD
<20	4.0	2.0	1.5	0.6	0.3	5.0
20-50	7.0	3.5	2.5	1.0	0.5	8.0
50-100	10.0	4.5	4.0	1.5	0.7	12.0
100-1000	12.0	5.5	5.0	2.0	1.0	15.0
>1000	15.0	7.0	6.0	2.5	1.4	20.0

Tabel 2.2 Batas arus harmonisa maksimal pada sistem *subtransmission* (69.001 - 161.000V) dalam persen I_L.

I_{sc}/I_L	<11	11≤h<17	17≤h<23	23≤h<35	35≥	TDD
<20	2.0	1.0	0.75	0.3	0.15	2.5
20-50	3.5	1.75	1.25	0.5	0.25	4.0
50-100	5.0	2.25	2.0	0.75	0.35	6.0
100-1000	6.0	2.75	2.5	1.0	0.5	7.5
>1000	7.5	3.5	3.0	1.25	0.7	10.0

Tabel 2.3 Batas arus harmonisa maksimal pada sistem transmisi (lebih dari 161.000V) dalam persen I_L.

I_{sc}/I_L	<11	11≤h<17	17≤h<23	23≤h<35	35≤	TDD
<50	2.0	1.0	0.75	0.3	0.15	2.5
≥50	3.0	1.5	1.15	0.45	0.22	3.75

Keterangan :
I_{sc} adalah arus hubung singkat maksimal pada PCC.
I_L adalah arus beban maksimal (komponen frekuensi fundamental) pada PCC.

2.2 Artificial Neural Network

Artificial Neural network atau biasa disingkat *neural network* (NN) merupakan salah satu jenis metode pada *machine learning* yang menirukan cara kerja neuron pada otak manusia. ANN diciptakan sebagai bentuk heneralisasi model matematis dari pemahaman manusia (*human cognition).* Elemen-elemen ANN atau jaringan saraf tiruan terorganisasi seperti anatomi otak. ANN dapat mempelajari pola dari pengalaman, melakukan generalisasi atas contoh yang diajarkan padanya, serta mengabstraksi karakteristik esensial input.

Jaringan saraf manusia terdiri dari dendrit, soma, akson, akson, dan sinapsis. ANN terdiri dari bagian-bagian yang fungsinya menyerupai bagian pada jaringan saraf manusia tersebut. Analogi bagian-bagian tersebut seperti tabel di bawah ini.

Tabel 2.4 Analogi jaringan saraf manusia dengan ANN

Jaringan Saraf Manusia	**Artificial Neural Network**
Dendrit	Input
Soma	Neuron
Sinapsis	Bobot
Akson	Output

2.2.1 Fungsi Aktivasi

Fungsi aktivasi merupakan fungsi yang mentrasformasikan sautu input menjadi output tertentu pada *neural network*. Fungsi aktivasi menentukan hasil output dari neuron di dalam model NN. Empat fungsi aktivasi yang paling banyak digunakan antara lain sigmoid, linear, ReLU, dan Tanh. Masih ada beberapa fungsi aktivasi lain di empat yang

disebutkan di atas. Penelitian pada tugas akhir ini fokus membandingkan empat activation function paling populer yaitu sigmoid, linear, ReLU, dan Tanh.

2.2.2 *Epoch*

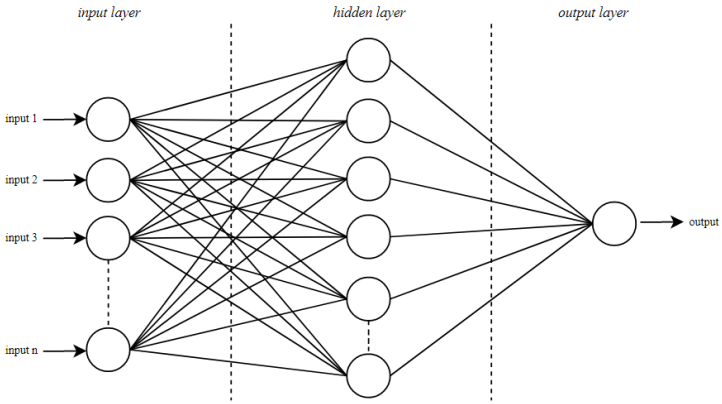
Proses pembelajaran pada model neural network ditandai dengan istilah epoch. Satu epoch berarti satu model NN telah mempelajari seluruh data training secara keseluruhan sebanyak sekali.

Pada ANN proses pengulangan pembelajaran bertujuan agar dicapai konvergensi nilai bobot atau weights. Tidak ada teori pasti yang bisa mengetahui jumlah *epoch* optimum dari sebuah model NN. Oleh karena itu pada penelitian ini dicobakan beberapa nilai epoch antara lain 10, 50, 100, 200, 1000, dan 2000.

2.2.3*Backpropagation*

Salah satu arsitektur NN yang banyak digunakan adalah *backpropagation*. *Backpropagation* merupakan algoritma yang bisa mengurangi *level error* dengan cara menyesuaikan bobotnya berdasarkan perbedaan *output* dan target yang diinginkan. *Backpopragation* memiliki tiga *layer* dalam proses pembelajarannya [6]. Layer tersebut antara lain input layer, hidden layer, dan output layer. Backpropagation NN merupakan perkembangan dari *single layer network*.

Arsitektur algoritma ANN backpropagation terdiri dari *input layer*, *hidden layer*, dan *output layer*. Pada input layer terjadi pengiriman sinyal input ke *hidden layer*. Pada hidden layer terjadi proses komputasi dengan adanya pengaruh bobot (*weight*) dan *bias*. Ada pula fungsi aktivasi khusus yang mempengaruhi nilai output model NN.



Gambar 2.1 Arsitektur NN *Backpropagation*

Algoritma pembelajaran *backpropagation artificial neural network* dapat dijelaskan sebagai berikut :
Langkah 0 :
Inisialisasi bobot (*weight*) awal.
Langkah 1 :
Menetapkan jumlah *epoch*, target kesalahan (*error*), dan *learning rate* (α)
Langkah 2 :
Tahap *feed-forward*.
Pada tahap akan dicari error dan nilai aktivasi pada semua neuron. Proses *feed-forward* tersebut seperti di bawah ini :
Xi, i=1,2,3,...,n) adalah unit sinyal input.

- Sinyal input Xi diteruskan ke *hidden layer*.
- Neuron pada hidden layer (Zj, j=1,2,3,...,n) menjumlahkan sinyal-sinyal input dengan *weight*-nya :

$$Zinj = B_{1j} + \sum_{i=1}^{n} XiVij$$

Keterangan :
B_{1j} : Bias yang menghubungkan node 1 dengan hiden layer 1.
Xi : Nilai input neuron/node i
Vij : Nilai weight yang menghubungkan input neuron i dengan neuron hidden layer j

- Output dari hidden layer 1 dijumlahkan lalu dikalikan dengan nilai *weight*. Hasilnya kemudian dijumlahkan dengan bias dari *output layer*.

$$Yin = B_{2k} + \sum_{i=1}^{p} ZjWjk$$

Keterangan :
Yin : Nilai fungsi aktivasi
B_{2k} : Nilai bias yang menghubungkan neuron 2 dengan *hidden layer* 2.
Zj : Nilai *output* dari hidden layer
Wjk : Nilai bobot yang menghubungkan *input* neuron 2 dengan *output layer* 2

- Nilai output dari neuron k didapatkan dengan memberikan fungsi aktivasi. Sebagai contoh, digunakan fungsi aktivasi sigmoid.

$$Y = f(Yin) = \frac{1}{1 + e^{-Y_{in}}}$$

Langkah 3 :
Dilakukan proses *backward propagation*.

- Dihitung nilai error dari neuron output pada layer output (Y_k)

$$\delta_k = (t_k - y_k) y_k (1 - y_k)$$

Keterangan :
δ_k : Nilai error layer output (Y_k)
t_k : Nilai output yang diinginkan
y_k : Nilai fungsi aktivasi output layer (Yk)

- Dihitung koreksi weight yang digunakan untuk memperbaiki Wjk nantinya

$$\Delta W_{jk} = \alpha . \delta_k z_j$$

- Dihitung koreksi nilai bias yang akan digunakan untuk memperbaiki W_{0k}

$$\Delta W_{0k} = \alpha . \delta_k$$

- Kemudian dilakukan pengiriman nilai δ_k ke neuron yang berada di lapisan bawahnya.
- Setiap neuron hidden layer (Z_j) menjumlahkan delta input dari neuron-neuron yang berada di lapisan atasnya.

$$\delta_{inj} = \sum_{k=1}^{m} \delta_k W_{jk}$$

- Nilai error didapatkan dari perkalian nilai di atas tersebut dengan turunan fungsi aktivasi.

$$\delta_j = \delta_{inj} f'(Z_{inj})$$

- Dihitung koreksi *weight* yang digunakan untuk memperbaiki V_{ij}

$$\Delta V_{ij} = \alpha . \delta_j x_i$$

- Dihitung koreksi bias yang digunakan untuk memperbaiki V_{oj}

$$\Delta V_{oj} = \alpha . \delta_j$$

- Perbaikan *weight* dan bias dilakukan pada tiap neuron output (Yk)

$$W_{jk(new)} = W_{jk(old)} + \Delta W_{jk}$$

- Tiap neuron hidden layer diperbaiki *weight* dan biasnya.

$$V_{ij(new)} = V_{ij(old)} + \Delta V_{ij}$$

Bobot baru dari proses inilah yang digunakan pada proses iterasi *epoch* selanjutnya. Proses looping ini dilakukan hingga jumlah epoch tercapai atau target nilai *error* terlampaui.

2.3 *Microcontroller* STM32F7

STM32F7 merupakan microcontroller buatan STMicroelectronics dengan prosesor ARM Cortex-M7. CPU dari mikrokontroler ini diklaim mampu bekerja hingga frekuensi 216 MHz.

ARM Cortex-M7 menyediakan model pemrograman yang tidak jauh berbeda dengan C dan biner. Hasil *compile* dari mikrokontroler ini kompatibel dengan prosesor Cortex versi sebelumnya.

Mikrokontroler STM32F7 memerlukan konsumsi daya sebesar 1.8 V dan frekuensi *clock* 180 MHz. Sedangkan arus yang dibutuhkan secara umum 120 µA pada mode stop.

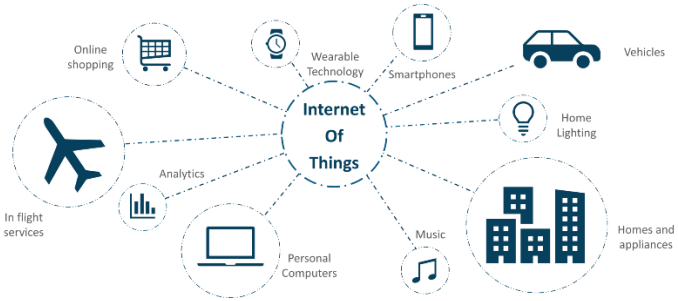
Gambar 2.2 Microcontroller STM32F7
Sumber : https://www.micrium.com/aws-iot-stm32f7-disco-shg/

2.4 *Internet of Things*

Internet of things atau IoT merupakan konsep dimana suatu objek memiliki kemampuan untuk mentransfer data ke jaringan tanpa diperlukan interaksi secara langsung dengan manusia. Internet menjadi penghubung antara perangkat IoT dengan jaringan. IoT dapat diaplikasikan di berbagai aspek kehidupan manusia. Sebagai contoh sistem *smart city* merupakan implementasai dari IoT. IoT juga banyak digunakan pada industri otomotif, konstruksi, kesehatan, dan pertanian.

Pada bidang elektro, contoh penerapan internet of things dapat dilihat pada konsep *smart home*. *Smart home* merupakan konsep dimana menyalakan dan mematikan perangkat yang ada di rumah dapat dilakaukan melalui gawai yang dibawa dimanapun berada.

Modul IoT yang paling sederhana dapat dibuat menggunakan microcontroller dasar seperti arduino uno, nano, dan sejenisnya.



Gambar 2.3 Implementasi IoT pada kehidupan
Sumber : https://www.edureka.co/blog/iot-applications/

2.4.1 Cara Kerja IoT

Internet of things bekerja dengan menggunakan argumentasi pemrograman. Setiap argumentasi dalam program tersebut mewakili suatu perintah yang akan dieksekusi oleh perangkat atau mesin yang terhubung ke jaringan secara otomatis. Proses eksekusi tersebut tidak dibatasi jarak dan bekerja tanpa campur tangan manusia. Jadi, pekerjaan tetap bisa dilakukan oleh perangkat atau mesin tanpa kehadiran manusia secara langsung di sana.

Internet menjadi penghubung sehingga interaksi tersebut bisa terjadi. Manusia bertugas sebagai pengatur dan pengawas dari perangkat-perangkat yang sedang bekerja tersebut.

BAB 3
PERANCANGAN DAN PEMODELAN NN

3.1 Sistem Identifikasi Beban

Data harmonisa di penelitian ini didapatkan dari sistem identifikasi beban yang dirancang dengan piranti utama *microcontroller* STM32F7. *Microcontroller* tersebut telah diprogram dan dirancang sedemikian hingga bisa membaca nilai harmonisa dengan *valid*.

Pemilihan jenis beban yang digunakan pada penelitian ini diambil dari beban-beban listrik rumah tangga yang mengandung harmonisa. Beban-beban tersebut antara lain *charger* laptop *full load*, *air conditioner* (AC), televisi LED, lampu hemat energi, dan kipas angin.

Data harmonisa yang telah didapatkan tersebut digunakan sebagai bahan pembelajaran terhadap jenis beban dan pengategorian output dengan *artificial neural network* (ANN). Masukan (*input*) dari proses pembelajaran ANN menggunakan data harmonisa beban orde ganjil pertama hingga kesembilan. Keluaran (*output*) ANN ini menunjukkan jenis beban. Proses pembelajaran atau *training* menggunakan Python versi 3.6.6.

3.2 Perencanaan

Proses perencanaan penelitian tugas akhir ini terdiri dari lima tahap. Tahap-tahap tersebut yaitu :

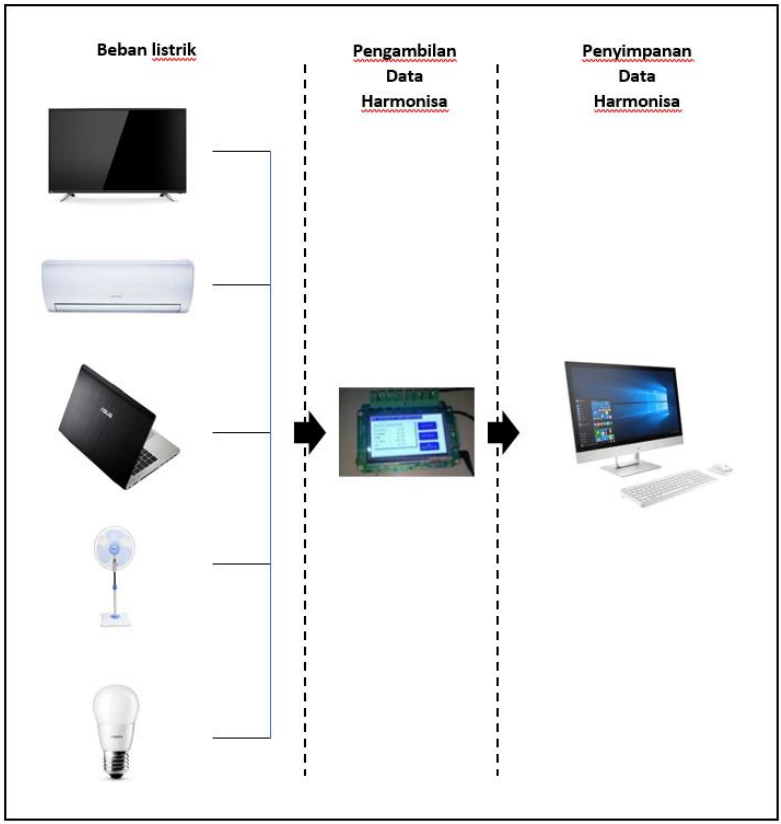
1. Pengambilan data
2. Pemodelan
3. *Preparing* Data
4. Identifikasi
5. *Monitoring* Beban Listrik yang Aktif

3.2.1 Pengambilan Data

Pada tahap awal ini dilakukan pengambilan dan pencatatan data harmonisa dari lima beban listrik rumah tangga yaitu charger laptop yang sedang digunakan untuk mengisi baterai, *air conditioner* (AC), televisi LED, lampu hemat energi, dan kipas angin. Proses pengambilan data ini dibantu oleh *microcontroller* STM32F7 yang telah diprogram dan dirancang sedemikian hingga bisa membaca nilai harmonisa dengan benar. *Microcontroller* tersebut terhubung langsung dengan komputer melalui kabel serial. Hasil pengukuran tersebut dapat langsung di-export

menjadi file XLS (File dari program *Microsoft Excel*). Alat tersebut dapat membaca harmonisa dari orde pertama hingga ke-29.

Selanjutnya dari kelima jenis beban rumah tangga di atas dilakukan 31 skenario kobinasi jenis beban yang aktif.



Gambar 3.1 Bagan pengambilan data harmonisa

Skenario seluruh kombinasi beban aktif tersebut seperti pada tabel 3.1 di bawah ini.

Tabel 3.1 31 Kombinasi beban

Kategori	Keterangan Beban
1	Lampu Hemat Energi
2	*Charger*
3	Lampu Hemat Energi + *Charger*
4	TV
5	TV + Lampu Hemat Energi
6	TV + *Charger*
7	TV + *Charger* + Lampu Hemat Energi
8	Kipas Angin
9	Kipas Angin + Lampu Hemat Energi
10	Kipas Angin + *Charger*
11	Kipas Angin + *Charger* + Lampu Hemat Energi
12	Kipas Angin + TV
13	Kipas Angin + TV + Lampu Hemat Energi
14	Kipas Angin + TV+ *Charger*
15	Kipas Angin + TV+ *Charger* + Lampu Hemat Energi
16	AC
17	AC + Lampu Hemat Energi
18	AC + *Charger*
19	AC + *Charger* + Lampu Hemat Energi
20	AC + TV
21	AC + TV + Lampu Hemat Energi
22	AC + TV + *Charger*
23	AC + TV + *Charger* + Lampu Hemat Energi
24	AC + Kipas Angin
25	AC + Kipas Angin + Lampu Hemat Energi
26	AC + Kipas Angin + *Charger*
27	AC + Kipas Angin + *Charger* + Lampu Hemat Energi
28	AC + Kipas Angin + TV
29	AC + Kipas Angin + TV + Lampu Hemat Energi
30	AC + Kipas Angin + TV + *Charger*
31	AC + Kipas Angin + TV + *Charger* + Lampu Hemat Energi

Setiap macam kombinasi beban dilakukan pengambilan dan pencatatan harmonisa sebanyak 100 kali.

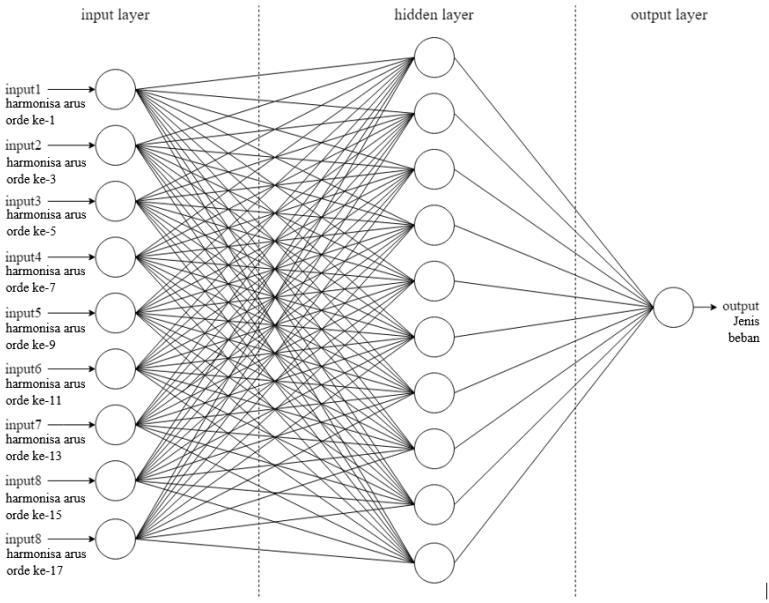
Data_Number	I1	I3	I5	I7	I9	I11	I13	I15	I17	I19	I21	I23	I25	I27	I29	Kategori
1	3.9347	0.1617	0.1477	0.0273	0.0151	0.0007	0	0	0.0011	0	0	0	0	0	0	31
2	3.9313	0.1738	0.1529	0.0156	0.0159	0.0018	0.0041	0	0.0012	0	0	0	0	0	0	31
3	3.9322	0.1617	0.1483	0.0239	0.0026	0.0035	0.0024	0	0	0	0	0	0	0	0	31
4	3.9195	0.1527	0.1461	0.0283	0.0009	0.0017	0.0005	0	0	0	0	0	0	0	0	31
5	3.9331	0.1562	0.144	0.0287	0.0049	0	0	0	0	0	0	0	0	0	0	31
6	3.9515	0.1623	0.1426	0.0321	0.0082	0	0	0	0	0	0	0	0	0	0	31
7	3.997	0.1733	0.1472	0.027	0.0078	0	0	0	0	0	0	0	0	0	0	31
8	4.0302	0.1744	0.1509	0.033	0.0055	0.0014	0	0	0	0	0	0	0	0	0	31
9	4.0237	0.1798	0.1456	0.0387	0	0.003	0.0035	0	0	0	0	0	0	0	0	31
10	4.0206	0.1925	0.1536	0.0294	0.007	0.0016	0.0045	0	0	0	0	0	0	0	0	31
11	4.0076	0.1999	0.1554	0.0205	0.0132	0	0.0052	0	0	0	0	0	0	0	0	31
12	4.0218	0.1954	0.1541	0.0235	0.0111	0	0.0037	0	0.0006	0.0007	0	0	0	0	0	31
13	4.0244	0.1958	0.1617	0.0303	0.0107	0	0.0025	0	0.0011	0	0	0	0	0	0	31
14	4.0412	0.1859	0.1582	0.039	0.0075	0.0029	0.0047	0	0.0037	0.0008	0	0	0	0	0	31
15	4.0613	0.1843	0.1674	0.0332	0.0052	0.0015	0.0075	0	0.0032	0.0009	0	0	0	0	0	31

Gambar 3.2 Contoh tampilan *file* xls pengambilan data harmonisa

3.2.2 Pemodelan

Setelah mendapatkan data harmonisa dari peralatan yang digunakan dalam penelitian ini, proses selanjutnya adalah memisahkan data *training* dengan data *validasi*. Tahap *training* model ANN pada penelitian ini menggunakan *script artificial neural network* pada bahasa pemrograman Python versi 3.6.6. Program Python dijalankan pada komputer dengan sistem operasi Windows 10 Enterprise.

Parameter yang digunakan pada proses training atau pembelajaran ini adalah dengan variasi *activation function*. Variasi tersebut antara lain linier, sigmoid, TanH, dan ReLU. Selanjutnya, dilakukan variasi jumlah *epoch* yaitu 10, 50, 100, 200, 1000, dan 2000. Kemudian dilakukan variasi jumlah *neuron hidden layer* 9-10-1, 9-15-1, 9-35-1, 9-50-1. Terakhir, dilakukan variasi jumlah hidden layer yaitu satu, dua, dan tiga hidden layer. Tujuan dari proses ini untuk mempelajari pola dari harmonisa dan mengenali jenis beban yang sedang aktif.



Gambar 3.3 Pemodelan *neural network* dengan 9 buah input

Blok diagram input berfungsi untuk memasukkan data harmonisa yang akan dites. Model yang didapatkan dari hasil training akan diujicobakan pada data harmonisa yang didapatkan.

Pertama, data input dilakukan normalisasi kemudian dimasukkan ke neural network lalu dilakukan denormalisasi. Normalisasi merupakan proses dimana data yang yang *range*-nya luas dipersempit sehingga didapatkan hasil yang lebih akurat. Sedangkan denormalisasi adalah mengembalikan data hasil normalisasi ke bentuk data aslinya.

3.2.3 *Preparing* Data

Pada tahap ini dilakukan pemisahan data antara data validasi dan data *training*. Setiap jenis beban dilakukan pencatatan data harmonisanya sebanyak seratus kali. Total data harmonisa seluruh kombinasi beban sebanyak 3100 buah. Dari data tersebut 80 persen digunakan untuk *training* ANN dan 20 persen untuk uji validasi.

3.2.4 Klasifikasi

Tahap keempat ini dilakukan klasifikasi dan identifikasi jenis beban dengan memasukkan data *test* ke dalam model *backpropagation artificial neural network* yang telah ditentukan dengan program Python.

3.2.5 Monitoring Beban Listrik yang Aktif

Pada tahap ini dilakukan penyajian data hasil identifikasi dengan model neural network yang telah dibuat. Penyajian hasil ini ditampilkan di dalam antarmuka program Python.

BAB 4
SIMULASI DAN ANALISIS DATA

Pada bab ini dibahas hasil percobaan berupa simulasi model *artificial neural network* dan analisis datanya. Perbedaan hasil output dari berbagai parameter yang dicoba juga akan dibahas di bab ini. Hasil output yang berbeda tersebut kemudian dijadikan evaluasi untuk menentukan konfigurasi parameter model ANN yang diusulkan.

4.1 Arsitektur *Neural Network*

Backpropagation Neural Network multilayer perceptron setidaknya memuliki tiga lapisan neuron. Satu lapisan masukkan atau *input layer*, minimal satu lapisan *hidden layer*, dan satu lapisan keluaran atau *output layer*.

4.1.1 *Input Layer*

Jumlah node input layer yang dibuat adalah sembilan. Input layer dari model NN yang digunakan pada penelitian ini merupakan sembilan nilai harmonik ganjil pertama dari seluruh kemungkinan kombinasi beban yang terpasang. Sembilan harmonik ganjil pertama tersebut adalah harmonisa orde ke-1, 3, 5, 7, 9. 11, 13, 15, dan 17.

4.1.2 *Hidden Layer*

Tidak ada teori yang menentukan jumlah node minimal atau maksimal agar didapatkan keakuratan atau performa maksimal. Pada penelitian ini dilakukan percobaan beberapa lapis *hidden layer*. Dilakukan pula percobaan variasi jumlah node tiap lapis *hidden layer*.

4.1.3 *Output Layer*

Output layer pada penelitian ini ditentukan sesuai jumlah seluruh kemungkinan kombinasi beban yang diuji. Seluruh kombinasi beban tersebut berjumlah 31.

4.2 Konfigurasi Parameter *Neural Network*

Pada penelitian ini dilakukan pemilihan beberapa variasi parameter ANN agar didapatkan *output* target dengan keakuratan paling tinggi. Variasi parameter tersebut meliputi jenis fungsi aktivasi, jumlah *epoch*, jumlah neuron atau node pada *hidden layer*, serta jumlah lapisan *hidden layer*.

4.3 Tahap *Learning* dan Validasi ANN

Pada tahap ini dilakukan pembelajaran model ANN dengan referensi data harmonisa setiap jenis kombinasi beban sebagai *input* dan jenis beban menyala sebagai target *output*. Tujuan dari tahap learning ini adalah agar model ANN dapat mengidentifikasi jenis beban yang sedang aktif pada sistem listrik berdasarkan nilai harmonisa yang dibaca alat ukur.

Dibawah ini adalah contoh hasil pengukuran nilai harmonisa dari beban jenis pertama (lampu hemat energi) sebanyak 35 kali.

Data_Number	I1	I3	I5	I7	I9	I11	I13	I15	I17	Target
1	0.115	0.081	0.038	0.026	0.017	0	0.0009	0	0	1
2	0.116	0.08	0.038	0.027	0.016	0.0005	0.0001	0	0	1
3	0.113	0.079	0.038	0.026	0.017	0	0.0004	0.0003	0	1
4	0.112	0.079	0.037	0.026	0.016	0.0008	0	0.0004	0	1
5	0.113	0.079	0.039	0.025	0.017	0.0017	0.0011	0	0	1
6	0.112	0.079	0.039	0.025	0.018	0.0011	0	0.0009	0.0005	1
7	0.112	0.078	0.037	0.025	0.017	0.0002	0.001	0.0027	0	1
8	0.112	0.079	0.039	0.026	0.018	0.0003	0	0.0002	0	1
9	0.112	0.078	0.038	0.026	0.018	0.0016	0.0005	0.0003	0	1
10	0.111	0.08	0.038	0.027	0.018	0.0004	0.0008	0.0017	0	1
11	0.111	0.078	0.039	0.026	0.018	0.0001	0.0017	0.0019	0	1
12	0.109	0.077	0.038	0.027	0.02	0.0026	0.0006	0.0014	0	1
13	0.11	0.076	0.038	0.026	0.02	0.0027	0.0004	0.0012	0	1
14	0.111	0.077	0.038	0.026	0.017	0.0015	0.0012	0.0001	0	1
15	0.109	0.078	0.037	0.026	0.018	0.0019	0.0008	0.0006	0	1
16	0.111	0.078	0.039	0.026	0.018	0.0017	0	0.0012	0	1
17	0.11	0.077	0.038	0.026	0.018	0.003	0	0.0009	0	1
18	0.109	0.077	0.039	0.026	0.019	0.0024	0.0006	0.0014	0	1
19	0.107	0.078	0.038	0.026	0.019	0.0033	0.0022	0	0	1
20	0.108	0.076	0.039	0.026	0.018	0.0035	0.0006	0.0009	0	1
21	0.108	0.077	0.039	0.026	0.018	0.0038	0.0009	0.0005	0	1
22	0.109	0.078	0.038	0.026	0.019	0.0033	0.0011	0.0015	0	1
23	0.108	0.078	0.039	0.025	0.019	0.003	0.001	0.0012	0	1
24	0.109	0.077	0.039	0.027	0.02	0.0034	0	0	0	1
25	0.11	0.077	0.038	0.026	0.019	0.0044	0	0.0006	0	1
26	0.108	0.077	0.04	0.026	0.021	0.0043	0	0.0004	0	1
27	0.109	0.077	0.038	0.024	0.019	0.004	0	0.0012	0	1
28	0.109	0.078	0.036	0.027	0.02	0.0031	0	0.0007	0	1
29	0.106	0.076	0.039	0.026	0.019	0.0034	0	0.0008	0	1
30	0.109	0.076	0.036	0.024	0.016	0.0014	0	0	0	1
31	0.108	0.075	0.038	0.025	0.016	0.003	0	0	0	1
32	0.107	0.076	0.038	0.026	0.017	0.0019	0	0	0	1
33	0.108	0.076	0.04	0.027	0.02	0.0041	0	0	0	1
34	0.108	0.075	0.038	0.025	0.019	0.0026	0	0.0006	0	1
35	0.107	0.077	0.038	0.025	0.018	0.002	0	0	0	1

Tabel 4.1 Hasil pengukuran harmonisa LHE 35 kali

4.3.1 *Learning* dengan Variasi Fungsi Aktivasi

Pada tahap learning pertama ini dilakukan proses pembelajaran ANN dengan menggunakan variasi fungsi aktivasi. Fungsi aktivasi pada proses learning ini antara lain linier, sigmoid, Tanh, dan ReLU. Sedangkan parameter lain yang digunakan adalah epoch sejumlah 1500 dan jumlah neuron tiap *hidden layer* 9-40-1.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan fungsi aktivasi linier, neuron tiap layer 9-40-1, dan jumlah *epoch* 1500.

Tabel 4.2 Hasil pembelajaran dan tes validasi dengan fungsti aktivasi Linier

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1500	linier	1	100
9-40-1	1500	linier	2	100
9-40-1	1500	linier	3	100
9-40-1	1500	linier	4	100
9-40-1	1500	linier	5	100
9-40-1	1500	linier	6	100
9-40-1	1500	linier	7	100
9-40-1	1500	linier	8	100
9-40-1	1500	linier	9	100
9-40-1	1500	linier	10	100
9-40-1	1500	linier	11	100
9-40-1	1500	linier	12	100
9-40-1	1500	linier	13	100
9-40-1	1500	linier	14	100
9-40-1	1500	linier	15	100
9-40-1	1500	linier	16	96
9-40-1	1500	linier	17	93,33
9-40-1	1500	linier	18	93,33
9-40-1	1500	linier	19	100
9-40-1	1500	linier	20	87,5
9-40-1	1500	linier	21	76,92
9-40-1	1500	linier	22	96,55
9-40-1	1500	linier	23	94,12
9-40-1	1500	linier	24	95,65

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1500	linier	25	95,24
9-40-1	1500	linier	26	90,91
9-40-1	1500	linier	27	100
9-40-1	1500	linier	28	87,5
9-40-1	1500	linier	29	85,71
9-40-1	1500	linier	30	100
9-40-1	1500	linier	31	85,71

Dari hasil pembelajaran di Tabel 4.2 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-21 yaitu 76,92%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran ini adalah 96,45%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan fungsi aktivasi sigmoid, neuron tiap layer 9-40-1, dan jumlah *epoch* 1500.

Tabel 4.3 Hasil pembelajaran dan tes validasi dengan fungsti aktivasi Sigmoid

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1500	sigmoid	1	100
9-40-1	1500	sigmoid	2	100
9-40-1	1500	sigmoid	3	100
9-40-1	1500	sigmoid	4	100
9-40-1	1500	sigmoid	5	100
9-40-1	1500	sigmoid	6	100
9-40-1	1500	sigmoid	7	100
9-40-1	1500	sigmoid	8	100
9-40-1	1500	sigmoid	9	100
9-40-1	1500	sigmoid	10	95,65
9-40-1	1500	sigmoid	11	100
9-40-1	1500	sigmoid	12	100
9-40-1	1500	sigmoid	13	100
9-40-1	1500	sigmoid	14	100
9-40-1	1500	sigmoid	15	100
9-40-1	1500	sigmoid	16	96
9-40-1	1500	sigmoid	17	80

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1500	sigmoid	18	93,33
9-40-1	1500	sigmoid	19	100
9-40-1	1500	sigmoid	20	82,35
9-40-1	1500	sigmoid	21	100
9-40-1	1500	sigmoid	22	100
9-40-1	1500	sigmoid	23	87,5
9-40-1	1500	sigmoid	24	95,65
9-40-1	1500	sigmoid	25	85,71
9-40-1	1500	sigmoid	26	90,91
9-40-1	1500	sigmoid	27	100
9-40-1	1500	sigmoid	28	80
9-40-1	1500	sigmoid	29	85,71
9-40-1	1500	sigmoid	30	100
9-40-1	1500	sigmoid	31	84,21

Dari hasil pembelajaran di Tabel 4.3 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-17 dam 28 yaitu 80%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan fungsi aktivasi sigmoid ini adalah 95,81%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan fungsi aktivasi Tanh, neuron tiap layer 9-40-1, dan jumlah *epoch* 1500.

Tabel 4.4 Hasil pembelajaran dan tes validasi dengan fungsti aktivasi Tanh

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1500	Tanh	1	100
9-40-1	1500	Tanh	2	100
9-40-1	1500	Tanh	3	100
9-40-1	1500	Tanh	4	100
9-40-1	1500	Tanh	5	100
9-40-1	1500	Tanh	6	100
9-40-1	1500	Tanh	7	100
9-40-1	1500	Tanh	8	100
9-40-1	1500	Tanh	9	100
9-40-1	1500	Tanh	10	100

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1500	Tanh	11	100
9-40-1	1500	Tanh	12	100
9-40-1	1500	Tanh	13	100
9-40-1	1500	Tanh	14	100
9-40-1	1500	Tanh	15	100
9-40-1	1500	Tanh	16	96
9-40-1	1500	Tanh	17	93,33
9-40-1	1500	Tanh	18	96,55
9-40-1	1500	Tanh	19	100
9-40-1	1500	Tanh	20	94,12
9-40-1	1500	Tanh	21	93,33
9-40-1	1500	Tanh	22	96,55
9-40-1	1500	Tanh	23	94,12
9-40-1	1500	Tanh	24	95,65
9-40-1	1500	Tanh	25	95,24
9-40-1	1500	Tanh	26	95,65
9-40-1	1500	Tanh	27	100
9-40-1	1500	Tanh	28	93,33
9-40-1	1500	Tanh	29	100
9-40-1	1500	Tanh	30	100
9-40-1	1500	Tanh	31	85,71

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.4 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-31 yaitu 85,71%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan fungsi aktivasi Tanh ini adalah 97,74%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan fungsi aktivasi ReLU, neuron tiap layer 9-40-1, dan jumlah *epoch* 1500.

Tabel 4.5 Hasil pembelajaran dan tes validasi dengan fungsti aktivasi ReLU

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1500	ReLU	1	100
9-40-1	1500	ReLU	2	100
9-40-1	1500	ReLU	3	100

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1500	ReLU	4	100
9-40-1	1500	ReLU	5	100
9-40-1	1500	ReLU	6	100
9-40-1	1500	ReLU	7	100
9-40-1	1500	ReLU	8	100
9-40-1	1500	ReLU	9	100
9-40-1	1500	ReLU	10	95,65
9-40-1	1500	ReLU	11	100
9-40-1	1500	ReLU	12	100
9-40-1	1500	ReLU	13	100
9-40-1	1500	ReLU	14	100
9-40-1	1500	ReLU	15	100
9-40-1	1500	ReLU	16	96
9-40-1	1500	ReLU	17	93,33
9-40-1	1500	ReLU	18	92,86
9-40-1	1500	ReLU	19	100
9-40-1	1500	ReLU	20	88,89
9-40-1	1500	ReLU	21	100
9-40-1	1500	ReLU	22	96,55
9-40-1	1500	ReLU	23	94,12
9-40-1	1500	ReLU	24	95,65
9-40-1	1500	ReLU	25	95,24
9-40-1	1500	ReLU	26	91,67
9-40-1	1500	ReLU	27	88,89
9-40-1	1500	ReLU	28	85,71
9-40-1	1500	ReLU	29	92,31
9-40-1	1500	ReLU	30	100
9-40-1	1500	ReLU	31	84,21

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.5 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-31 yaitu 84,21%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan fungsi aktivasi ReLU ini adalah 96,77%.

Tabel 4.6 Perbandingan nilai akurasi rata-rata tiap fungsi aktivasi

Jenis Beban	Akurasi Rata-Rata Tiap Fungsi Aktivasi (%)			
	Linear	Sigmoid	Tanh	ReLU
1	100	100	100	100
2	100	100	100	100
3	100	100	100	100
4	100	100	100	100
5	100	100	100	100
6	100	100	100	100
7	100	100	100	100
8	100	100	100	100
9	100	100	100	100
10	100	95,65	100	95,65
11	100	100	100	100
12	100	100	100	100
13	100	100	100	100
14	100	100	100	100
15	100	100	100	100
16	96	96	96	96
17	93,33	80	93,33	93,33
18	93,33	93,33	96,55	92,86
19	100	100	100	100
20	87,5	82,35	94,12	88,89
21	76,92	100	93,33	100
22	96,55	100	96,55	96,55
23	94,12	87,5	94,12	94,12
24	95,65	95,65	95,65	95,65
25	95,24	85,71	95,24	95,24
26	90,91	90,91	95,65	91,67
27	100	100	100	88,89
28	87,5	80	93,33	85,71
29	85,71	85,71	100	92,31
30	100	100	100	100
31	85,71	84,21	85,71	84,21
Rata-Rata	96,45	95,81	97,74	96,77

Dari hasil pembelajaran dengan variasi jenis *activation function* didapatkan nilai akurasi validasi rata-rata terendah pada model

backpropagation NN dengan fungsi aktivasi sigmoid, yaitu 95,81%. Sedangkan nilai akurasi tertinggi didapatkan dengan fungsi aktivasi Tanh yaitu 97,74%. Perbandingan nilai akurasi dari setiap jenis fungsi aktivasi tidak berbeda jauh. Hal ini menunjukkan bahwa pemilihan jenis fungsi aktivasi tidak memberikan dampak signifikan pada hasil pembelajaran *backpropagation* NN dengan kasus penggunaan seperti pada penelitian kali ini.

4.3.2 *Learning* dengan Variasi Jumlah *Epoch*

Tahap selanjutnya dilakukan learning ANN dengan variasi jumlah *epoch*. Jumlah *epoch* yang dibandingkan pada penelitian ini yaitu 10, 50, 100, 200, 1000, dan 2000. Jumlah input harmonisa yang digunakan sembilan, fungsi aktivasi linier, dan menggunakan neuron tiap *layer* 9-40-1.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah epoch 10, neuron tiap layer 9-40-1, dan fungsi aktivasi linier.

Tabel 4.7 Hasil pembelajaran dan tes validasi dengan jumlah *epoch* 10

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	10	linier	1	55,74
9-40-1	10	linier	2	0
9-40-1	10	linier	3	72,73
9-40-1	10	linier	4	82,76
9-40-1	10	linier	5	0
9-40-1	10	linier	6	68,97
9-40-1	10	linier	7	75,56
9-40-1	10	linier	8	56,67
9-40-1	10	linier	9	0
9-40-1	10	linier	10	88,46
9-40-1	10	linier	11	66,67
9-40-1	10	linier	12	0
9-40-1	10	linier	13	44,9
9-40-1	10	linier	14	0
9-40-1	10	linier	15	80
9-40-1	10	linier	16	0
9-40-1	10	linier	17	46,81

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	10	linier	18	76,92
9-40-1	10	linier	19	12,5
9-40-1	10	linier	20	62,22
9-40-1	10	linier	21	50
9-40-1	10	linier	22	7,41
9-40-1	10	linier	23	71,19
9-40-1	10	linier	24	58,62
9-40-1	10	linier	25	35,29
9-40-1	10	linier	26	69,77
9-40-1	10	linier	27	0
9-40-1	10	linier	28	29,63
9-40-1	10	linier	29	56,41
9-40-1	10	linier	30	100
9-40-1	10	linier	31	40

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.7 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-2, 5, 9, 14, 16, dan 27 yaitu 0%. Artinya pada jenis beban tersebut model ANN satu ini tidak bisa mengidentifikasi beban aktif sama sekali. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah epoch sepuluh ini adalah 52,9%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah epoch 50, neuron tiap layer 9-40-1, dan fungsi aktivasi linier.

Tabel 4.8 Hasil pembelajaran dan tes validasi dengan jumlah *epoch* 50

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	50	linier	1	88,89
9-40-1	50	linier	2	83,72
9-40-1	50	linier	3	92,68
9-40-1	50	linier	4	96
9-40-1	50	linier	5	87,8
9-40-1	50	linier	6	80
9-40-1	50	linier	7	100
9-40-1	50	linier	8	100
9-40-1	50	linier	9	93,02

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	50	linier	10	83,72
9-40-1	50	linier	11	92,31
9-40-1	50	linier	12	92,31
9-40-1	50	linier	13	97,14
9-40-1	50	linier	14	88,37
9-40-1	50	linier	15	92,31
9-40-1	50	linier	16	88,89
9-40-1	50	linier	17	66,67
9-40-1	50	linier	18	86,67
9-40-1	50	linier	19	97,56
9-40-1	50	linier	20	68,75
9-40-1	50	linier	21	77,42
9-40-1	50	linier	22	86,96
9-40-1	50	linier	23	80,95
9-40-1	50	linier	24	89,47
9-40-1	50	linier	25	58,82
9-40-1	50	linier	26	84,62
9-40-1	50	linier	27	93,33
9-40-1	50	linier	28	77,55
9-40-1	50	linier	29	69,57
9-40-1	50	linier	30	100
9-40-1	50	linier	31	85

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.8 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-25 yaitu 58.82%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah epoch 50 ini adalah 86,45%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah epoch 100, neuron tiap layer 9-40-1, dan fungsi aktivasi linier.

Tabel 4.9 Hasil pembelajaran dan tes validasi dengan jumlah *epoch* 100

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	100	linier	1	88,89
9-40-1	100	linier	2	94,74
9-40-1	100	linier	3	92,68

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	100	linier	4	95,83
9-40-1	100	linier	5	92,31
9-40-1	100	linier	6	84,62
9-40-1	100	linier	7	100
9-40-1	100	linier	8	100
9-40-1	100	linier	9	93,02
9-40-1	100	linier	10	93,88
9-40-1	100	linier	11	92,31
9-40-1	100	linier	12	92,86
9-40-1	100	linier	13	97,14
9-40-1	100	linier	14	90,48
9-40-1	100	linier	15	97,56
9-40-1	100	linier	16	97,56
9-40-1	100	linier	17	73,17
9-40-1	100	linier	18	81,36
9-40-1	100	linier	19	97,56
9-40-1	100	linier	20	75
9-40-1	100	linier	21	78,79
9-40-1	100	linier	22	89,36
9-40-1	100	linier	23	93,02
9-40-1	100	linier	24	96,97
9-40-1	100	linier	25	68,57
9-40-1	100	linier	26	79,25
9-40-1	100	linier	27	93,33
9-40-1	100	linier	28	79,17
9-40-1	100	linier	29	76,19
9-40-1	100	linier	30	100
9-40-1	100	linier	31	85

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.9 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-25 yaitu 68,57%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah epoch 100 ini adalah 89,19%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah epoch 200, neuron tiap layer 9-40-1, dan fungsi aktivasi linier.

Tabel 4.10 Hasil pembelajaran dan tes validasi dengan jumlah *epoch* 200

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	200	linier	1	88,89
9-40-1	200	linier	2	100
9-40-1	200	linier	3	92,68
9-40-1	200	linier	4	97,96
9-40-1	200	linier	5	97,44
9-40-1	200	linier	6	93,33
9-40-1	200	linier	7	100
9-40-1	200	linier	8	100
9-40-1	200	linier	9	93,02
9-40-1	200	linier	10	97,87
9-40-1	200	linier	11	92,31
9-40-1	200	linier	12	96,3
9-40-1	200	linier	13	100
9-40-1	200	linier	14	94,74
9-40-1	200	linier	15	100
9-40-1	200	linier	16	94,74
9-40-1	200	linier	17	76,19
9-40-1	200	linier	18	86,21
9-40-1	200	linier	19	97,56
9-40-1	200	linier	20	77,42
9-40-1	200	linier	21	85,71
9-40-1	200	linier	22	91,3
9-40-1	200	linier	23	97,87
9-40-1	200	linier	24	91,89
9-40-1	200	linier	25	72,73
9-40-1	200	linier	26	85,19
9-40-1	200	linier	27	95,45
9-40-1	200	linier	28	80
9-40-1	200	linier	29	84,21
9-40-1	200	linier	30	100
9-40-1	200	linier	31	92,31

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.10 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-20 yaitu 72,73%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah *epoch* 200 ini adalah 91,94%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah epoch 1000, neuron tiap layer 9-40-1, dan fungsi aktivasi linier.

Tabel 4.11 Hasil pembelajaran dan tes validasi dengan jumlah *epoch* 1000

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	1000	linier	1	97,14
9-40-1	1000	linier	2	100
9-40-1	1000	linier	3	100
9-40-1	1000	linier	4	100
9-40-1	1000	linier	5	97,44
9-40-1	1000	linier	6	100
9-40-1	1000	linier	7	100
9-40-1	1000	linier	8	100
9-40-1	1000	linier	9	100
9-40-1	1000	linier	10	100
9-40-1	1000	linier	11	100
9-40-1	1000	linier	12	100
9-40-1	1000	linier	13	100
9-40-1	1000	linier	14	100
9-40-1	1000	linier	15	100
9-40-1	1000	linier	16	92,31
9-40-1	1000	linier	17	93,33
9-40-1	1000	linier	18	86,67
9-40-1	1000	linier	19	100
9-40-1	1000	linier	20	83,87
9-40-1	1000	linier	21	83,33
9-40-1	1000	linier	22	93,62
9-40-1	1000	linier	23	97,87
9-40-1	1000	linier	24	88,89
9-40-1	1000	linier	25	93,33
9-40-1	1000	linier	26	84,62
9-40-1	1000	linier	27	97,67
9-40-1	1000	linier	28	85,71
9-40-1	1000	linier	29	87,18
9-40-1	1000	linier	30	100
9-40-1	1000	linier	31	94,74

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.11 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-21 yaitu 83,33%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah *epoch* 1000 ini adalah 95,16%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah epoch 2000, neuron tiap layer 9-40-1, dan fungsi aktivasi linier.

Tabel 4.12 Hasil pembelajaran dan tes validasi dengan jumlah *epoch* 2000

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	2000	linier	1	97,14
9-40-1	2000	linier	2	100
9-40-1	2000	linier	3	100
9-40-1	2000	linier	4	100
9-40-1	2000	linier	5	97,44
9-40-1	2000	linier	6	96,55
9-40-1	2000	linier	7	100
9-40-1	2000	linier	8	100
9-40-1	2000	linier	9	100
9-40-1	2000	linier	10	100
9-40-1	2000	linier	11	97,67
9-40-1	2000	linier	12	100
9-40-1	2000	linier	13	100
9-40-1	2000	linier	14	100
9-40-1	2000	linier	15	100
9-40-1	2000	linier	16	92,31
9-40-1	2000	linier	17	88,37
9-40-1	2000	linier	18	86,67
9-40-1	2000	linier	19	100
9-40-1	2000	linier	20	81,25
9-40-1	2000	linier	21	83,33
9-40-1	2000	linier	22	93,62
9-40-1	2000	linier	23	97,87
9-40-1	2000	linier	24	91,43
9-40-1	2000	linier	25	84,85
9-40-1	2000	linier	26	84,62

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-40-1	2000	linier	27	97,67
9-40-1	2000	linier	28	83,33
9-40-1	2000	linier	29	87,18
9-40-1	2000	linier	30	100
9-40-1	2000	linier	31	94,74

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.12 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-20 yaitu 81,25%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah *epoch* 2000 ini adalah 94,52%.

Tabel 4.13 Perbandingan nilai akurasi rata-rata tiap jumlah epoch

Jenis Beban	Akurasi Rata-Rata Tiap Jumlah *Epoch* (%)					
	10	50	100	200	1000	2000
1	55,74	88,89	88,89	88,89	97,14	97,14
2	0	83,72	94,74	100	100	100
3	72,73	92,68	92,68	92,68	100	100
4	82,76	96	95,83	97,96	100	100
5	0	87,8	92,31	97,44	97,44	97,44
6	68,97	80	84,62	93,33	100	96,55
7	75,56	100	100	100	100	100
8	56,67	100	100	100	100	100
9	0	93,02	93,02	93,02	100	100
10	88,46	83,72	93,88	97,87	100	100
11	66,67	92,31	92,31	92,31	100	97,67
12	0	92,31	92,86	96,3	100	100
13	44,9	97,14	97,14	100	100	100
14	0	88,37	90,48	94,74	100	100
15	80	92,31	97,56	100	100	100
16	0	88,89	97,56	94,74	92,31	92,31
17	46,81	66,67	73,17	76,19	93,33	88,37
18	76,92	86,67	81,36	86,21	86,67	86,67
19	12,5	97,56	97,56	97,56	100	100
20	62,22	68,75	75	77,42	83,87	81,25
21	50	77,42	78,79	85,71	83,33	83,33
22	7,41	86,96	89,36	91,3	93,62	93,62
23	71,19	80,95	93,02	97,87	97,87	97,87

Jenis Beban	Akurasi Rata-Rata Tiap Jumlah *Epoch* (%)					
	10	50	100	200	1000	2000
24	58,62	89,47	96,97	91,89	88,89	91,43
25	35,29	58,82	68,57	72,73	93,33	84,85
26	69,77	84,62	79,25	85,19	84,62	84,62
27	0	93,33	93,33	95,45	97,67	97,67
28	29,63	77,55	79,17	80	85,71	83,33
29	56,41	69,57	76,19	84,21	87,18	87,18
30	100	100	100	100	100	100
31	40	85	85	92,31	94,74	94,74
Rata-Rata	52,9	86,45	89,19	91,94	95,16	94,52

Dari tabel 4.13 dapat dilihat bahwa akurasi rata-rata terkecil terdapat pada pembelajaran dengan jumlah epoch 50. Sedangkan nilai akurasi rata-rata terbesar adalah pada jumlah epoch 1000. Bertambahnya jumlah epoch pada pembelajaran berbanding lurus dengan nilai akurasi model ANN. Namun, setelah jumlah epoch 1000 model ANN tidak mengalami perbaikan nilai akurasi. Bahkan nilai akurasi pada jumlah epoch 2000 berkurang 0,64% dibandingkan dengan jumlah epoch 1000.

Setelah dilakukan beberapa kali percobaan validasi data didapatkan nilai akurasi yang tidak selalu sama. Ada perbedaan nilai akurasi pada tiap jenis konfigurasi yang besarnya <1%.

4.3.3 *Learning* dengan Variasi Jumlah Neuron *Hidden Layer*

Pada tahap ini dilakukan proses pembelajaran dengan variasi jumlah neuron atau node di dalam *hidden layer*. Jumlah hidden layer pada tahap ini berjumlah satu. *Activation function*-nya linier dan epoch berjumlah 100. Sedangkan jumlah neuronnya dilakukan variasi sehingga konfigurasi tiap layernya adalah 9-10-1, 9-15-1, 9-35-1, dan 9-50-1.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah jumlah neuron tiap layer 9-10-1, jumlah *epoch* 100, dan fungsi aktivasi linier.

Tabel 4.14 Hasil pembelajaran dan tes validasi dengan jumlah neuron 9-10-1

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-10-1	100	linier	1	93,75
9-10-1	100	linier	2	90,91
9-10-1	100	linier	3	88,24
9-10-1	100	linier	4	76,36
9-10-1	100	linier	5	97,56
9-10-1	100	linier	6	91,89
9-10-1	100	linier	7	97,96
9-10-1	100	linier	8	98,04
9-10-1	100	linier	9	97,3
9-10-1	100	linier	10	91,89
9-10-1	100	linier	11	86,49
9-10-1	100	linier	12	72,73
9-10-1	100	linier	13	100
9-10-1	100	linier	14	90,91
9-10-1	100	linier	15	100
9-10-1	100	linier	16	79,17
9-10-1	100	linier	17	85,71
9-10-1	100	linier	18	64,86
9-10-1	100	linier	19	84,62
9-10-1	100	linier	20	70,97
9-10-1	100	linier	21	95,24
9-10-1	100	linier	22	57,14
9-10-1	100	linier	23	84,21
9-10-1	100	linier	24	74,42
9-10-1	100	linier	25	80,95
9-10-1	100	linier	26	68,42
9-10-1	100	linier	27	96,77
9-10-1	100	linier	28	38,89
9-10-1	100	linier	29	72,22
9-10-1	100	linier	30	100
9-10-1	100	linier	31	92,31

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.14 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-22 yaitu 52,63%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan

model pembelajaran dengan jumlah neuron tiap layer 9-10-1 ini adalah 84,52%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah jumlah neuron tiap layer 9-15-1, jumlah *epoch* 100, dan fungsi aktivasi linier.

Tabel 4.15 Hasil pembelajaran dan tes validasi dengan jumlah neuron 9-15-1

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-15-1	100	linier	1	90,91
9-15-1	100	linier	2	83,87
9-15-1	100	linier	3	91,43
9-15-1	100	linier	4	81,63
9-15-1	100	linier	5	100
9-15-1	100	linier	6	94,74
9-15-1	100	linier	7	100
9-15-1	100	linier	8	100
9-15-1	100	linier	9	94,44
9-15-1	100	linier	10	85
9-15-1	100	linier	11	88,89
9-15-1	100	linier	12	83,87
9-15-1	100	linier	13	100
9-15-1	100	linier	14	93,02
9-15-1	100	linier	15	100
9-15-1	100	linier	16	80,95
9-15-1	100	linier	17	83,33
9-15-1	100	linier	18	89,47
9-15-1	100	linier	19	89,36
9-15-1	100	linier	20	80
9-15-1	100	linier	21	95,24
9-15-1	100	linier	22	93,75
9-15-1	100	linier	23	82,35
9-15-1	100	linier	24	86,27
9-15-1	100	linier	25	82,61
9-15-1	100	linier	26	91,43
9-15-1	100	linier	27	96,77

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-15-1	100	linier	28	84,44
9-15-1	100	linier	29	78,95
9-15-1	100	linier	30	100
9-15-1	100	linier	31	86,79

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.15 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-20 yaitu 71,43%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah neuron tiap layer 9-15-1 ini adalah 89,84%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah jumlah neuron tiap layer 9-15-1, jumlah *epoch* 100, dan fungsi aktivasi linier.

Tabel 4.16 Hasil pembelajaran dan tes validasi dengan jumlah neuron 9-35-1

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-35-1	100	linier	1	90,91
9-35-1	100	linier	2	90,91
9-35-1	100	linier	3	94,12
9-35-1	100	linier	4	79,12
9-35-1	100	linier	5	100
9-35-1	100	linier	6	97,3
9-35-1	100	linier	7	100
9-35-1	100	linier	8	100
9-35-1	100	linier	9	94,44
9-35-1	100	linier	10	89,47
9-35-1	100	linier	11	91,89
9-35-1	100	linier	12	82,54
9-35-1	100	linier	13	100
9-35-1	100	linier	14	95,45
9-35-1	100	linier	15	100
9-35-1	100	linier	16	82,93
9-35-1	100	linier	17	85,71
9-35-1	100	linier	18	89,47

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-35-1	100	linier	19	87,50
9-35-1	100	linier	20	75,00
9-35-1	100	linier	21	90,48
9-35-1	100	linier	22	93,75
9-35-1	100	linier	23	94,44
9-35-1	100	linier	24	86,27
9-35-1	100	linier	25	82,61
9-35-1	100	linier	26	88,89
9-35-1	100	linier	27	96,77
9-35-1	100	linier	28	82,61
9-35-1	100	linier	29	76,92
9-35-1	100	linier	30	100
9-35-1	100	linier	31	92,00

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.16 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-20 yaitu 69,23%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah neuron tiap layer 9-35-1 ini adalah 90,48%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah jumlah neuron tiap layer 9-50-1, jumlah *epoch* 100, dan fungsi aktivasi linier.

Tabel 4.17 Hasil pembelajaran dan tes validasi dengan jumlah neuron 9-50-1

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-50-1	100	linier	1	93,75
9-50-1	100	linier	2	90,91
9-50-1	100	linier	3	91,43
9-50-1	100	linier	4	79,17
9-50-1	100	linier	5	100
9-50-1	100	linier	6	92,31
9-50-1	100	linier	7	100
9-50-1	100	linier	8	100
9-50-1	100	linier	9	97,30

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-50-1	100	linier	10	89,47
9-50-1	100	linier	11	88,89
9-50-1	100	linier	12	84,38
9-50-1	100	linier	13	100
9-50-1	100	linier	14	92,68
9-50-1	100	linier	15	100
9-50-1	100	linier	16	86,36
9-50-1	100	linier	17	88,89
9-50-1	100	linier	18	89,47
9-50-1	100	linier	19	87,5
9-50-1	100	linier	20	80
9-50-1	100	linier	21	90,48
9-50-1	100	linier	22	93,75
9-50-1	100	linier	23	97,3
9-50-1	100	linier	24	89,8
9-50-1	100	linier	25	84,44
9-50-1	100	linier	26	91,43
9-50-1	100	linier	27	96,77
9-50-1	100	linier	28	84,44
9-50-1	100	linier	29	78,95
9-50-1	100	linier	30	100
9-50-1	100	linier	31	92

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.17 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-20 yaitu 71,43%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah neuron tiap layer 9-50-1 ini adalah 91,13 %.

Tabel 4.18 Perbandingan nilai akurasi rata-rata tiap jumlah neuron *hidden layer*

Jenis Beban	Akurasi Rata-Rata Tiap Neuron *Hidden Layer* (%)			
	9-10-1	9-15-1	9-35-1	9-50-1
1	93,75	90,91	90,91	93,75
2	90,91	83,87	90,91	90,91
3	88,24	91,43	94,12	91,43
4	76,36	81,63	79,12	79,17

Jenis Beban	Akurasi Rata-Rata Tiap Neuron *Hidden Layer* (%)			
	9-10-1	9-15-1	9-35-1	9-50-1
5	97,56	100	100	100
6	91,89	94,74	97,3	92,31
7	97,96	100	100	100
8	98,04	100	100	100
9	97,3	94,44	94,44	97,30
10	91,89	85	89,47	89,47
11	86,49	88,89	91,89	88,89
12	72,73	83,87	82,54	84,38
13	100	100	100	100
14	90,91	93,02	95,45	92,68
15	100	100	100	100
16	79,17	80,95	82,93	86,36
17	85,71	83,33	85,71	88,89
18	64,86	89,47	89,47	89,47
19	84,62	89,36	87,50	87,5
20	70,97	80	75,00	80
21	95,24	95,24	90,48	90,48
22	57,14	93,75	93,75	93,75
23	84,21	82,35	94,44	97,3
24	74,42	86,27	86,27	89,8
25	80,95	82,61	82,61	84,44
26	68,42	91,43	88,89	91,43
27	96,77	96,77	96,77	96,77
28	38,89	84,44	82,61	84,44
29	72,22	78,95	76,92	78,95
30	100	100	100	100
31	92,31	86,79	92,00	92
Rata-Rata	84,52	89,84	90,48	91,13

Tabel 4.18 menunjukkan menunjukkah bahwa nilai akurasi rata-rata tertinggi didapatkan pada model ANN dengan formasi *layer* 9-50-1 yaitu 91,13%. Sedangkan akurasi terendah didapatkan pada konfigurasi *layer* 9-10-1 yaitu sebesar 84,52%. Dari percobaan yang dilakukan ddapat disimpulkan bahwa jumlah neuron *hidden layer* berbanding lurus dengan besar akurasi model ANN. Seiring dengan kenaikan jumlah *hidden layer* NN, proses komputasi yang dilakukan semakin kompleks dan membuat waktu eksekusi program semakin lama.

4.3.4 *Learning* dengan Variasi Jumlah *Hidden Layer*

Selanjutnya dilakukan variasi pada jumlah hidden layer model NN untuk dibandingkan tingkat akurasinya. Variasi tersebut yaitu 9-10-1, 9-10-10-1, dan 9-10-10-10-1. Jumlah *epoch* pada tahap ini adalah 100. *Activation function* yang digunakan adalah linier.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah jumlah neuron tiap layer 9-10-1, jumlah *epoch* 100, dan fungsi aktivasi linier.

4.19 Hasil Pembelajaran dan tes validasi dengan konfigurasi neuron 9-10-1

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-10-1	100	linier	1	93,75
9-10-1	100	linier	2	90,91
9-10-1	100	linier	3	88,24
9-10-1	100	linier	4	76,36
9-10-1	100	linier	5	97,56
9-10-1	100	linier	6	91,89
9-10-1	100	linier	7	97,96
9-10-1	100	linier	8	98,04
9-10-1	100	linier	9	97,3
9-10-1	100	linier	10	91,89
9-10-1	100	linier	11	86,49
9-10-1	100	linier	12	72,73
9-10-1	100	linier	13	100
9-10-1	100	linier	14	90,91
9-10-1	100	linier	15	100
9-10-1	100	linier	16	79,17
9-10-1	100	linier	17	85,71
9-10-1	100	linier	18	64,86
9-10-1	100	linier	19	84,62
9-10-1	100	linier	20	70,97
9-10-1	100	linier	21	95,24
9-10-1	100	linier	22	57,14
9-10-1	100	linier	23	84,21
9-10-1	100	linier	24	74,42

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-10-1	100	linier	25	80,95
9-10-1	100	linier	26	68,42
9-10-1	100	linier	27	96,77
9-10-1	100	linier	28	38,89
9-10-1	100	linier	29	72,22
9-10-1	100	linier	30	100
9-10-1	100	linier	31	92,31

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.19 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-28 yaitu 38,89%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah neuron tiap layer 9-10-1 ini adalah 84,52%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah jumlah neuron tiap layer 9-10-10-1, jumlah *epoch* 100, dan fungsi aktivasi linier.

4.20 Hasil pembelajaran dan tes validasi dengan konfigurasi neuron 9-10-10-1

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-10-10-1	100	linier	1	90,91
9-10-10-1	100	linier	2	90,91
9-10-10-1	100	linier	3	91,43
9-10-10-1	100	linier	4	81,63
9-10-10-1	100	linier	5	100
9-10-10-1	100	linier	6	92,31
9-10-10-1	100	linier	7	100
9-10-10-1	100	linier	8	100
9-10-10-1	100	linier	9	94,44
9-10-10-1	100	linier	10	91,89
9-10-10-1	100	linier	11	88,89
9-10-10-1	100	linier	12	85,71
9-10-10-1	100	linier	13	100
9-10-10-1	100	linier	14	92,68
9-10-10-1	100	linier	15	100

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-10-10-1	100	linier	16	85,71
9-10-10-1	100	linier	17	73,17
9-10-10-1	100	linier	18	83,33
9-10-10-1	100	linier	19	89,8
9-10-10-1	100	linier	20	47,62
9-10-10-1	100	linier	21	81,82
9-10-10-1	100	linier	22	93,75
9-10-10-1	100	linier	23	86,49
9-10-10-1	100	linier	24	90,2
9-10-10-1	100	linier	25	65
9-10-10-1	100	linier	26	86,49
9-10-10-1	100	linier	27	100
9-10-10-1	100	linier	28	77,55
9-10-10-1	100	linier	29	73,68
9-10-10-1	100	linier	30	100
9-10-10-1	100	linier	31	80

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.20 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-20 yaitu 47,62%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah neuron tiap layer 9-10-10-1 ini adalah 87,9%.

Berikut ini hasil pembelajaran pengujian identifikasi beban dengan jumlah jumlah neuron tiap layer 9-10-10-10-1, jumlah *epoch* 100, dan fungsi aktivasi linier.

4.21 Hasil pembelajaran dan tes validasi dengan konfigurasi neuron 9-10-10-10-1

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-10-10-10-1	100	linier	1	93,75
9-10-10-10-1	100	linier	2	100
9-10-10-10-1	100	linier	3	91,43
9-10-10-10-1	100	linier	4	83,33
9-10-10-10-1	100	linier	5	92,68
9-10-10-10-1	100	linier	6	94,74

Neuron	*Epoch*	Fungsi aktivasi	Target	Akurasi (%)
9-10-10-10-1	100	linier	7	95,83
9-10-10-10-1	100	linier	8	100
9-10-10-10-1	100	linier	9	100
9-10-10-10-1	100	linier	10	100
9-10-10-10-1	100	linier	11	85,71
9-10-10-10-1	100	linier	12	87,5
9-10-10-10-1	100	linier	13	100
9-10-10-10-1	100	linier	14	95,24
9-10-10-10-1	100	linier	15	97,3
9-10-10-10-1	100	linier	16	70,59
9-10-10-10-1	100	linier	17	82,05
9-10-10-10-1	100	linier	18	85
9-10-10-10-1	100	linier	19	89,8
9-10-10-10-1	100	linier	20	81,48
9-10-10-10-1	100	linier	21	88,89
9-10-10-10-1	100	linier	22	86,67
9-10-10-10-1	100	linier	23	97,3
9-10-10-10-1	100	linier	24	82,76
9-10-10-10-1	100	linier	25	78,05
9-10-10-10-1	100	linier	26	82,35
9-10-10-10-1	100	linier	27	96,77
9-10-10-10-1	100	linier	28	80
9-10-10-10-1	100	linier	29	77,78
9-10-10-10-1	100	linier	30	100
9-10-10-10-1	100	linier	31	92

Dari hasil pembelajaran yang bisa dilihat di Tabel 4.21 didapatkan nilai rata-rata akurasi terkecil adalah pada jenis beban ke-20 yaitu 68,75%. Sedangkan nilai akurasi rata-rata keseluruhan beban dengan model pembelajaran dengan jumlah neuron tiap layer 9-10-10-10-1 ini adalah 89,84%.

Tabel 4.22 Perbandingan nilai akurasi rata-rata tiap jumlah *hidden layer*

Jenis Beban	Akurasi Rata-Rata Tiap Konfigurasi *Hidden Layer* (%)		
	9-10-1	9-10-10-1	9-10-10-10-1
1	93,75	90,91	93,75
2	90,91	90,91	100

Jenis Beban	Akurasi Rata-Rata Tiap Konfigurasi *Hidden Layer* (%)		
	9-10-1	9-10-10-1	9-10-10-10-1
3	88,24	91,43	91,43
4	76,36	81,63	83,33
5	97,56	100	92,68
6	91,89	92,31	94,74
7	97,96	100	95,83
8	98,04	100	100
9	97,3	94,44	100
10	91,89	91,89	100
11	86,49	88,89	85,71
12	72,73	85,71	87,5
13	100	100	100
14	90,91	92,68	95,24
15	100	100	97,3
16	79,17	85,71	70,59
17	85,71	73,17	82,05
18	64,86	83,33	85
19	84,62	89,8	89,8
20	70,97	47,62	81,48
21	95,24	81,82	88,89
22	57,14	93,75	86,67
23	84,21	86,49	97,3
24	74,42	90,2	82,76
25	80,95	65	78,05
26	68,42	86,49	82,35
27	96,77	100	96,77
28	38,89	77,55	80
29	72,22	73,68	77,78
30	100	100	100
31	92,31	80	92
Rata-Rata	84,52	87,9	89,84

Pada tabel 4.22 di atas dapat diketahui bahwa nilai akurasi validasi rata-rata terendah adalah pada model NN dengan satu buah *hidden layer* berkonfigurasi 9-10-1. Sedangkan akurasi tertinggi didapatkan pada model NN dengan tiga *hidden layer* yaitu sebesar 89,84%. Hal tersebut menunjukkan bahwa nilai akurasi model NN berbanding lurus dengan jumlah *hidden layer* yang dimiliki. Semakin

banyak jumlah hidden layer maka NN semakin mudah mengenali jenis beban dan semaki baik akurasinya.

Halaman ini sengaja dikosongkan

BAB 5
PENUTUP

5.1 Kesimpulan

Beberapa hal yang bisa disimpulkan dari hasil pembelajaran, pengujian validasi, serta analisis data pada penelitian tugas akhir ini antara lain :

1. Dari hasil pengujian validasi model ANN dengan variasi jenis fungsi aktivasi didapatkan akurasi terbaik pada fungsi aktivasi Tanh yaitu sebesar 97,74%. Hal tersebut menunjukkan bahwa Tanh cocok digunakan pada kasus dengan data uji yang linier seperti pada penelitian ini.
2. Dari hasil pengujian validasi model ANN dengan variasi jumlah *epoch* didapatkan akurasi terbaik pada jumlah *epoch* 1000 yaitu sebesar 95,16. Hal ini menunjukkan bahwa semakin besar jumlah *epoch*, akurasi dari model ANN semakin baik. Namun, jumlah *epoch* memiliki titik optimum tertentu. Jika nilai tersebut dilampaui maka model ANN tidak akan mengalami kenaikan akurasi.
3. Dari hasil pengujian validasi model ANN dengan variasi jumlah neuron *hidden layer* didapatkan akurasi terbaik pada model NN dengan konfigurasi *layer* 9-50-1 yaitu sebesar 91,13%. Hal tersebut menunjukkan bahwa semakin banyak neuron pada *hidden layer* menjadikan model NN memiliki akurasi yang semakin baik.
4. Dari hasil pengujian validasi model ANN dengan variasi jumlah *hidden layer* didapatkan akurasi terbaik pada model ANN dengan tiga buah layer (9-10-10-10-1) yaitu sebesar 89,84%. Semakin banyak *hidden layer*, ANN memiliki *weight* dan *bias* yang makin banyak dan perhitungannya menjadi lebih kompleks sehingga tingkat akurasinya pun makin baik.

5.2 Saran

Saran yang diberikan pada penelitian ini adalah :

1. Pada penelitian ini digunakan beberapa alat dengan nilai harmonisa arus yang besarnya hampir serupa. Disarankan untuk menggunakan jenis beban yang nilai harmonisa arusnya signifikan agar proses identifikasi beban semakin akurat.

2. Pada penelitian ini hanya menggunakan lima jenis beban. Sebaiknya gunakan jenis beban yang lebih banyak untuk pada penelitian selanjutnya.
3. Pada penelitian ini digunakan *backpropagation multilayer perceptron artificial neural network*. Dapat dilakukan percobaan identifikasi de081gan jenis NN lain untuk dibandingkan hasil akurasinya dengan *backpropagation NN*.

DAFTAR PUSTAKA

[1] Naufal Rasyiq, "Identifikasi Beban Pada Sistem Tegangan Rendah Menggunakan Harmonisa Secara Real Time Berbasis Fast Fourier Transform dan Neural Network", Indonesia, 2018

[2] Arrillaga, J., Watson, N.R., "Power System Harmonics Second Edition", John Wiley & Sons Ltd, Inggris, 2003

[3] W. Mack Grady, S. Santoso, "Understanding power system harmonics", IEEE Power Eng. Rev., pp. 8–11, 2001

[4] Dugan, Roger C., "Electrical Power Systems Quality", McGraw-Hill, Second Edition, 2004

[5] W. Tan, V. I. John, "Nonlinear fluorescent systems: their impact on power quality", Proc. Can. Conf., Sep. 25–28, 1994, pp. 144–147

[6] D. Srinivasan, W. S. Ng, A. C. Liew, "Neural network-based signature recognition for harmonic source identification," IEEE Trans. Power Del., vol. 21, no. 01, pp. 398–405, 2006

[7] Baggini, Angelo, "Handbook of Power Quality", John Wiley & Sons Ltd., Inggris, 2008

[8] Poisson, O., Rioual, P., Meunier, M., "Detection and Measurement of Power Quality Disturbances Using Wavelet Transform", IEEE Trans. Power Del, Vol. 15, No.3, Jul. 2013

[9] Moolayil, Jojo, "Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python", Apress, Canada, 2019

Halaman ini sengaja dikosongkan

RIWAYAT HIDUP PENULIS

Penulis bernama lengkap Irfan Purwito Nugroho. Teman-temannya biasa memanggil dengan panggilan Irfan atau Papang. Lahir di Ngawi pada tanggal 24 Agustus 1994. Sejak kecil tinggal di Tuban bersama orang tua. Pada usia 7 tahun, orang tuanya memutuskan untuk kembali ke kampung halamannya di Ngawi. Jenjang sekolah dasar ditempuh di SDN Margomulyo 1 Ngawi. Kemudian melanjutkan pendidikannya di SMPN 2 Ngawi dan SMAN 2 Ngawi. Penulis menyelesaikan pendidikan menengah atasnya di tahun 2013. Selepas SMA, penulis melanjutkan pendidikannya di Departemen Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya.

Saat kelas 8 SMP penulis mulai berkenalan dengan dunia *internet marketing*. Berbagai metode *make money online* pernah penulis coba dan praktikkan. Hingga, penulis memutuskan untuk fokus di bidang *blogging* dan *social media optimization*. Ketika kelas 9 SMP, penulis pernah menjadi juara 1 *Speedy Blog Competition* yang didakan oleh PT. Telkom tahun 2010. Saat ini penulis aktif mengelola beberapa blog dan akun media sosial.

Disamping aktivitasnya sebagai *internet marketer* tersebut, penulis tetap berusaha menyelesaikan studinya di ITS Surabaya. Penelitian yang tertulis pada buku ini merupakan langkah terakhir penulis sebelum menyentuh garis *finish* studi sarjana. *In syaa Allah*. Penulis dapat dihubungi via email di irfaneering@gmail.com atau via *Whatsapp* di 082-111-111-560.