

BAB II

TINJAUAN PUSTAKA

II.1 Penelitian Terdahulu

Beberapa penelitian sebelumnya yang membahas tentang Artificial Neural Network Backpropagation dan deep learning diantaranya seperti penelitian yang dilakukan oleh:

Penelitian yang dilakukan oleh (Zajmi et al., 2018) dengan judul penelitian “Concepts, Methods, and Performances of Particle Swarm Optimization, Backpropagation, and Neural Networks”, penelitian ini bertujuan untuk kemajuan Pembelajaran Mesin, sejak awal dan selama beberapa tahun terakhir, perhatian khusus telah diberikan pada Jaringan Syaraf Tiruan. Sebagai inspirasi dari seleksi alam kelompok hewan dan sistem saraf manusia, Jaringan Saraf Tiruan yang juga dikenal sebagai Neural Networks telah menjadi kekuatan komputasi baru yang digunakan untuk memecahkan masalah dunia nyata. Neural Networks sendiri sebagai sebuah konsep melibatkan berbagai metode untuk mencapai kesuksesan mereka; dengan demikian, makalah ulasan ini menjelaskan ikhtisar masing-masing metode yang disebut Particle Swarm Optimization, Backpropagation, dan Neural Network itu sendiri. Penjelasan singkat tentang konsep, sejarah, kelebihan, dan kekurangan diberikan, diikuti dengan penelitian terbaru yang dilakukan pada metode ini. Uraian solusi dan aplikasi di berbagai sektor industri seperti Kedokteran atau Teknologi Informasi telah disediakan. Bagian terakhir membahas secara

singkat arah, tantangan Neural Networks saat ini, dan masa depan untuk mencapai tingkat keberhasilan tertinggi dalam memecahkan masalah dunia nyata.

Penelitian yang dilakukan oleh (Mutiara S. Simanjuntak, Wanayumini, Rika Rosnelly, 2019) dengan judul “The Activity Activation Function Of Multilayer Perceptron - Based Cardiac Abnormalities”, penelitian ini dalam pengimplementasiannya bertujuan untuk melihat gangguan jantung mengacu pada aktivitas tidak teratur di jantung. Kelainan jantung terkadang tidak menunjukkan gejala apa pun dan tidak masuk akal yang dapat menyebabkan kematian mendadak karena fungsi jantung yang retak. Artikel ini mengembangkan program yang mampu mendeteksi aktivitas kelainan jantung melalui aplikasi Multilayer Perceptron (MLP). Sejumlah data sinyal detak jantung dari elektrokardiogram (EKG) akan digunakan dalam makalah ini untuk melatih dan menguji kinerja jaringan MLP. MLP dilatih dengan beberapa teknik yaitu Backpropagation (BP), Bayesian Regularity (BR), dan Levenberg-Marquardt (LM).

Penelitian yang dilakukan oleh (Purba et al., 2020) dengan judul “The optimalization of backpropagation neural networks to simplify decision making” penelitian ini dilakukan karena PT. FIF Medan memiliki kendala dalam menentukan apakah konsumen diberikan pembiayaan. Penelitian menggunakan Jaringan Syaraf Tiruan Back Propagation. Back Propagation Neural Network adalah metode yang menyederhanakan masalah yang kompleks. Penyederhanaan model dengan mengambil inti permasalahan yang paling esensial. Penelitian akan dilakukan dengan tahapan mengidentifikasi masalah, menentukan kebutuhan, menganalisis, dan memilih alternatif. 8 (delapan) kriteria akan menjadi parameter,

antara lain jumlah tanggungan, lama bekerja, status rumah, riwayat keuangan, tempat kerja karyawan, jenis kendaraan, masa kerja pembiayaan, dan pendapatan bulanan.

Penelitian yang dilakukan oleh (Yan et al., 2020) dengan judul “Trustworthy Network Anomaly Detection Based on an Adaptive Learning Rate and Momentum in IIoT” penelitian ini dalam pengimplementasiannya menggunakan Internet of Things (IIoT) industri membawa kenyamanan bagi industri, itu juga membawa masalah keamanan. Karena banyaknya data yang dihasilkan oleh lonjakan perangkat IIoT, tidak mungkin untuk memastikan apakah data ini mengandung serangan atau data yang tidak dapat dipercaya, oleh karena itu, bagaimana memastikan keamanan dan kepercayaan perangkat IIoT menjadi masalah yang mendesak untuk dipecahkan. Dalam artikel ini, kami merancang algoritme klasifikasi engsel baru berdasarkan penurunan gradien mini-batch dengan kecepatan dan momentum pembelajaran adaptif (HCA-MBGDALRM) untuk meminimalkan efek serangan keamanan. Algoritme secara signifikan meningkatkan kinerja pelatihan jaringan dalam dibandingkan dengan jaringan saraf tradisional, pohon keputusan, dan regresi logistik dalam hal skala dan kecepatan. Selain itu, kami telah memecahkan masalah kemiringan data dalam fase acak, dan kami menerapkan kerangka kerja paralel untuk HCA-MBGDALRM untuk mempercepat kecepatan pemrosesan kumpulan data lalu lintas yang sangat besar.

Penelitian yang dilakukan oleh (Windarto, Naam, et al., 2020) dengan judul penelitian “Model Arsitektur Neural Network Dengan Kombinasi K-Medoids dan Backpropagation pada kasus Pandemi Covid-19 di Indonesia”, penelitian ini

bertujuan untuk membuat model prediksi pada arsitektur neural network terbaik dengan mengkombinasi metode k-medoids dan backpropagation pada kasus pandemi COVID-19 di Indonesia. Data diperoleh dari Kementerian Kesehatan, dicuplik dan diolah dari covid19.go.id dan bnpb.go.id. Kasus yang diangkat adalah jumlah persebaran pandemi COVID-19 di Indonesia per tanggal 7 juli 2020 yang terdiri 34 record. Variabel yang digunakan pada penelitian adalah jumlah kasus positif (x1), jumlah kasus sembuh (x2) dan jumlah kasus meninggal dunia (x3) berdasarkan provinsi. Proses analisis data menggunakan bantuan software RapidMiner. Solusi yang diberikan adalah dengan mengkombinasi metode k-medoids dan backpropagation. Dimana metode k-medoids melakukan pemetaan berupa klaster yang ditentukan. Label klaster yang digunakan adalah klaster tinggi (C1= zona merah), klaster waspada (C2= zona kuning), klaster rendah (C3= zona hijau). Hasil dari pemetaan klaster dilanjutkan ke metode backpropagation untuk memprediksi hasil akurasi dari klaster yang ada. Dengan menggunakan model arsitektur terbaik 3-2-1 diperoleh nilai akurasi 94,17% dengan learning_rate= 0.696. Hasil pemetaan klaster diperoleh 9 provinsi berada di klaster tinggi (C1= zona merah), 3 provinsi berada di klaster waspada (C2= zona kuning) dan 22 provinsi berada di klaster rendah (C3= zona hijau). Diharapkan hasil dari penelitian dapat memberikan informasi kepada pemerintah berupa pemetaan klaster terhadap wilayah di Indonesia.

Penelitian yang dilakukan oleh (Mardiyah, 2020) dengan judul penelitian “Implementasi Deep Learning untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Kebun dan Sawah”, penelitian

ini dalam pengimplementasiannya bertujuan untuk mengklasifikasi berbagai kekayaan alam yang ada di Indonesia seperti pertanian dan perkebunan. Pertanian dan perkebunan di Indonesia sangat beraneka ragam, seperti sawah yang dapat menghasilkan padi, kedelai, jagung, umbi-umbian, dan lain-lainnya. Sedangkan, perkebunan di Indonesia seperti kebun dengan hasil berupa kopi, teh, lada, kelapa sawit, pala, dan lain-lain. sawah yang merupakan contoh pertanian dan perkebunan yang merupakan contoh kebun memiliki karakteristik yang sama sehingga sulit untuk dibedakan oleh masyarakat atau orang awam apabila dilihat dengan menggunakan foto udara atau foto yang diambil dari ketinggian tertentu serta membantu pemerintah dalam melihat ketahanan pangan yang ada di Indonesia. Salah satu cara untuk membentuk model yaitu dengan memanfaatkan ilmu komputasi yang salah satu tujuannya adalah untuk mengambil informasi dari citra digital untuk mengenal obyek secara otomatis. Salah satu metode deep learning yang sedang berkembang saat ini yaitu Convolutional Neural Network (CNN). Metode CNN merupakan metode yang masukkan (input data) berupa citra. Metode ini memiliki lapisan khusus yang bernama lapisan konvolusi dimana pada lapisan sebuah citra masukkan (input citra) akan menghasilkan sebuah pola dari beberapa bagian citra yang nantinya akan lebih mudah untuk diklasifikasikan. Lapisan konvolusi memiliki fungsi pembelajaran citra menjadi lebih efisien untuk diimplementasikan. Oleh karena itu peneliti ingin memanfaatkan metode CNN ini untuk dapat mengklasifikasi kebun dan sawah, dengan tujuan untuk membedakan karakteristik kebun dan sawah. Berdasarkan hasil klasifikasi diperoleh hasil akurasi

testing sebesar 75%. Dapat disimpulkan bahwa metode CNN dapat mengklasifikasikan citra kebun dan sawah dengan baik.

Penelitian yang dilakukan oleh (Zajmi et al., 2018) dengan judul penelitian “Concepts, Methods, and Performances of Particle Swarm Optimization, Backpropagation, and Neural Networks”, penelitian ini bertujuan untuk kemajuan Pembelajaran Mesin, sejak awal dan selama beberapa tahun terakhir, perhatian khusus telah diberikan pada Jaringan Syaraf Tiruan. Sebagai inspirasi dari seleksi alam kelompok hewan dan sistem saraf manusia, Jaringan Saraf Tiruan yang juga dikenal sebagai Neural Networks telah menjadi kekuatan komputasi baru yang digunakan untuk memecahkan masalah dunia nyata. Neural Networks sendiri sebagai sebuah konsep melibatkan berbagai metode untuk mencapai kesuksesan mereka; dengan demikian, makalah ulasan ini menjelaskan ikhtisar masing-masing metode yang disebut Particle Swarm Optimization, Backpropagation, dan Neural Network itu sendiri. Penjelasan singkat tentang konsep, sejarah, kinerja, kelebihan, dan kekurangan diberikan, diikuti dengan penelitian terbaru yang dilakukan pada metode ini. Uraian solusi dan aplikasi di berbagai sektor industri seperti Kedokteran atau Teknologi Informasi telah disediakan. Bagian terakhir membahas secara singkat arah, tantangan Neural Networks saat ini, dan masa depan untuk mencapai tingkat keberhasilan tertinggi dalam memecahkan masalah dunia nyata.

Penelitian yang dilakukan oleh (Windarto, Naam, et al., 2020) dengan judul penelitian “Model Arsitektur Neural Network Dengan Kombinasi K-Medoids dan Backpropagation pada kasus Pandemi Covid-19 di Indonesia”, penelitian ini bertujuan untuk membuat model prediksi pada arsitektur neural network terbaik

dengan mengkombinasi metode k-medoids dan backpropagation pada kasus pandemi COVID-19 di Indonesia. Data diperoleh dari Kementerian Kesehatan, dicuplik dan diolah dari covid19.go.id dan bnpb.go.id. Kasus yang diangkat adalah jumlah persebaran pandemi COVID-19 di Indonesia per tanggal 7 juli 2020 yang terdiri 34 record. Variabel yang digunakan pada penelitian adalah jumlah kasus positif (x_1), jumlah kasus sembuh (x_2) dan jumlah kasus meninggal dunia (x_3) berdasarkan provinsi. Proses analisis data menggunakan bantuan software RapidMiner. Solusi yang diberikan adalah dengan mengkombinasi metode k-medoids dan backpropagation. Dimana metode k-medoids melakukan pemetaan berupa klaster yang ditentukan. Label klaster yang digunakan adalah klaster tinggi (C_1 = zona merah), klaster waspada (C_2 = zona kuning), klaster rendah (C_3 = zona hijau). Hasil dari pemetaan klaster dilanjutkan ke metode backpropagation untuk memprediksi hasil akurasi dari klaster yang ada. Dengan menggunakan model arsitektur terbaik 3-2-1 diperoleh nilai akurasi 94,17% dengan $learning_rate=0.696$. Hasil pemetaan klaster diperoleh 9 provinsi berada di klaster tinggi (C_1 = zona merah), 3 provinsi berada di klaster waspada (C_2 = zona kuning) dan 22 provinsi berada di klaster rendah (C_3 = zona hijau). Diharapkan hasil dari penelitian dapat memberikan informasi kepada pemerintah berupa pemetaan klaster terhadap wilayah di Indonesia.

Penelitian yang dilakukan oleh (Mardiyah, 2020) dengan judul penelitian “Implementasi Deep Learning untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Kebun dan Sawah”, penelitian ini dalam pengimplementasiannya bertujuan untuk mengklasifikasi berbagai

kekayaan alam yang ada di Indonesia seperti pertanian dan perkebunan. Pertanian dan perkebunan di Indonesia sangat beraneka ragam, seperti sawah yang dapat menghasilkan padi, kedelai, jagung, umbi-umbian, dan lain-lainnya. Sedangkan, perkebunan di Indonesia seperti kebun dengan hasil berupa kopi, teh, lada, kelapa sawit, pala, dan lain-lain. sawah yang merupakan contoh pertanian dan perkebunan yang merupakan contoh kebun memiliki karakteristik yang sama sehingga sulit untuk dibedakan oleh masyarakat atau orang awam apabila dilihat dengan menggunakan foto udara atau foto yang diambil dari ketinggian tertentu serta membantu pemerintah dalam melihat ketahanan pangan yang ada di Indonesia. Salah satu cara untuk membentuk model yaitu dengan memanfaatkan ilmu komputasi yang salah satu tujuannya adalah untuk mengambil informasi dari citra digital untuk mengenal obyek secara otomatis. Salah satu metode deep learning yang sedang berkembang saat ini yaitu Convolutional Neural Network (CNN). Metode CNN merupakan metode yang masukkan (input data) berupa citra. Metode ini memiliki lapisan khusus yang bernama lapisan konvolusi dimana pada lapisan sebuah citra masukkan (input citra) akan menghasilkan sebuah pola dari beberapa bagian citra yang nantinya akan lebih mudah untuk diklasifikasikan. Lapisan konvolusi memiliki fungsi pembelajaran citra menjadi lebih efisien untuk diimplementasikan. Oleh karena itu peneliti ingin memanfaatkan metode CNN ini untuk dapat mengklasifikasi kebun dan sawah, dengan tujuan untuk membedakan karakteristik kebun dan sawah. Berdasarkan hasil klasifikasi diperoleh hasil akurasi testing sebesar 75%. Dapat disimpulkan bahwa metode CNN dapat mengklasifikasikan citra kebun dan sawah dengan baik.

Penelitian yang dilakukan oleh (Wanto et al., 2020) dengan judul penelitian “Analisis Penurunan Gradien dengan Kombinasi Fungsi Aktivasi pada Algoritma JST untuk Pencarian Akurasi Terbaik”, penelitian ini bertujuan untuk menganalisa gradient descent yang terbaik yang dapat dijadikan rujukan untuk digunakan pada algoritma JST, khususnya algoritma backpropagation dalam masalah prediksi data, klasifikasi maupun pengenola pola. Metode gradient descent yang akan dianalisa diantaranya: Gradient descent backpropagation (traingd), Gradient descent with momentum backpropagation (traingdm), Gradient descent with adaptive learning rate backpropagation (traingda), dan Gradient descent with momentum and adaptive learning rate backpropagation (traingdx). Fungsi training akan dikombinasikan dengan fungsi aktivasi (fungsi transfer) sigmoid bipolar (tansig), transfer linier (purelin) dan sigmoid biner (logsig). Data sampel yang digunakan untuk proses analisa adalah data times-series Indeks Pembangunan Manusia di Indonesia yang diperoleh dari Badan Pusat Statistik (BPS). Model arsitektur yang digunakan untuk analisa gradient descent diantaranya: 6-10-15-1, 6-15- 20-1, 6-20-25-1 dan 6-25-30-1. Berdasarkan hasil analisa, fungsi training terbaik adalah traingda dengan model arsitektur 6- 15-20-1 yang menghasilkan tingkat akurasi 91% dan MSE Testing 0,000731529 (lebih kecil dibandingkan metode lainnya).

Penelitian yang dilakukan oleh (Redho Islami 2020) dengan judul penelitian “Implementasi Deep Learning Dalam Mendeteksi Penyakit Menggunakan Convolutional Neural Network Dan Pytorch”, penelitian ini dalam pengimpelemntasinya bertujuan untuk mendeteksi penyakit dengan melihat dari kesehatan fisik yang merupakan hal penting untuk diperhatikan, namun terkadang

banyak orang yang abai untuk memperhatikannya. Kerap kali keluhan yang dirasakan seseorang dan gejala penyakit tertentu sulit dideteksi sejak dini oleh yang bersangkutan. Gangguan pada saluran pencernaan (gastrointestinal) ialah suatu kelainan atau penyakit pada jalan makanan/pencernaan. Penyakit gastrointestinal yang termasuk yaitu kelainan penyakit kerongkongan (esophagus), lambung (gaster), usus halus (intestinum), usus besar (colon), hati (liver), saluran empedu (traktus biliaris) dan pankreas. Penyakit saluran pencernaan merupakan penyakit yang berbahaya dan menyebabkan kematian, sehingga perlu adanya media bantu berupa sistem mudah memberikan solusi yang tepat untuk menangani permasalahan tersebut. Pemanfaatan computer vision sangat berguna bagi bidang kesehatan dengan menggunakan convolutional neural network dan Pytorch dalam mendeteksi sebuah objek penyakit. Penelitian ini bermaksud untuk memberikan sebuah inovasi kepada tenaga medis yang ahli dibidangnya dalam menganalisa sebuah penyakit gastrointestinal dengan menggunakan metode deep learning. Hasil testing model yang diperoleh dari uji validasi dengan kategori polyps nilai akurasi sebesar 83% - 90%, normal-pylorus nilai akurasi sebesar 86% - 92%, dyed- lifted-polyps nilai akurasi sebesar 83% - 97%, esophagitis nilai akurasi sebesar 82% - 92%, dan dyed-resection-margins nilai akurasi sebesar 82% - 93%.

Penelitian yang dilakukan oleh (Yan et al., 2020) dengan judul “Trustworthy Network Anomaly Detection Based on an Adaptive Learning Rate and Momentum in IIoT” penelitian ini dalam pengimplementasiannya menggunakan Internet of Things (IIoT) industri membawa kenyamanan bagi industri, itu juga membawa masalah keamanan. Karena banyaknya data yang dihasilkan oleh

lonjakan perangkat IIoT, tidak mungkin untuk memastikan apakah data ini mengandung serangan atau data yang tidak dapat dipercaya, oleh karena itu, bagaimana memastikan keamanan dan kepercayaan perangkat IIoT menjadi masalah yang mendesak untuk dipecahkan. Dalam artikel ini, kami merancang algoritme klasifikasi engsel baru berdasarkan penurunan gradien mini-batch dengan kecepatan dan momentum pembelajaran adaptif (HCA-MBGDALRM) untuk meminimalkan efek serangan keamanan. Algoritme secara signifikan meningkatkan kinerja pelatihan jaringan dalam dibandingkan dengan jaringan saraf tradisional, pohon keputusan, dan regresi logistik dalam hal skala dan kecepatan. Selain itu, kami telah memecahkan masalah kemiringan data dalam fase acak, dan kami menerapkan kerangka kerja paralel untuk HCA-MBGDALRM untuk mempercepat kecepatan pemrosesan kumpulan data lalu lintas yang sangat besar.

Penelitian yang dilakukan oleh (Yanto et al., 2021) dengan judul penelitian “Klasifikasi Tekstur Kematangan Buah Jeruk Manis Berdasarkan Tingkat Kecerahan Warna dengan Metode Deep Learning Convolutional Neural Network”, Penelitian ini bertujuan untuk mengembangkan sistem untuk mengklasifikasikan tekstur kematangan buah jeruk manis berdasarkan tingkat kecerahan warna. Pada pengimplementasiannya, jeruk manis sangat banyak dikonsumsi oleh manusia, karena jeruk kaya akan vitamin C. Pada umumnya proses klasifikasi untuk menentukan jeruk yang layak (baik) dan tidak berkualitas (busuk) masih menggunakan cara manual yaitu pengamatan langsung secara visual terhadap buah. Ada beberapa kelemahan pengklasifikasian dengan cara ini, yaitu adanya keterbatasan penglihatan manusia, dipengaruhi oleh kondisi psikologis pengamatan dan

membutuhkan waktu yang lama terutama untuk perkebunan besar. Salah satu cara untuk mengklasifikasikan jeruk manis melalui sistem komputer memakai algoritma deep learning convolutional neural network (CNN). Dengan dikembangkan pada lapisan multilayer perceptron (MLP), algoritma tersebut dapat diolah dua dimensi data, terhadap gambar serta mampu melakukan klasifikasi pada citra dengan kelas-kelas yang lebih banyak atau besar. Pada penelitian ini dengan 100 dataset gambar jeruk manis menunjukkan tingkat akurasi klasifikasi maka hasilnya sebesar 97.5184 %. dilakukan klasifikasi maka hasilnya sebesar 67.8221 %. pengujian sebanyak 10 citra buah jeruk yang terbagi menjadi 5 citra jeruk bagus dan 5 citra jeruk busuk sebesar 96% untuk training 92% untuk testing yang dinilai telah mampu melakukan klasifikasi kelayakan buah jeruk manis dengan sangat baik. Grafik hasil akurasi testing nya adalah 0.92 atau 92%. Hasil ini cukup bagus, terhadap Tampilan histogram RGB citra jeruk bagus.

Penelitian yang dilakukan oleh (Irnanda et al., 2022) dengan judul penelitian “Optimasi Particle Swarm Optimization Pada Peningkatan Prediksi dengan Metode Backpropagation Menggunakan Software RapidMiner”. Penelitian ini menggunakan teknik hybrid yaitu dengan membuat pembobotan pada uji algoritma Particle Swarm Optimization (PSO) yang kemudian di masukan bobot ke dalam algoritma Backpropagation. Backpropagation sering mengalami terjadinya lokal minimum, masalah waktu pelatihan yang lama untuk mencapai konvergen dan proses penentuan parameter (learning rate dan momentum) yang tepat dalam proses pelatihan, maka penggunaan pada Particle Swarm Optimization (PSO) untuk mencari posisi terbaik untuk masalah optimasi dalam ruang fitur yang terdiri dari

sekumpulan partikel yang mencari posisi terbaik. Hasil penelitian yang dilakukan dengan menggunakan data jumlah indeks pembangunan manusia dari tahun 2017- 2020, implementasi Backpropagation + particle swarm optimization lebih baik dibandingkan hanya menggunakan implementasi Backpropagation. Hasil uji coba yang dilakukan dengan software Rapid Miner model arsitektur terbaik adalah model 2-9-1 dengan jumlah RMSE 0.056 +/- 0.000 dalam implementasi Backpropagation, sedangkan dalam implementasi Backpropagation + particle swarm optimization jumlah RMSE 0.055 +/- 0.000 (lebih baik 0.001). Semakin kecil RMSE (Root Mean Squared Error) maka semakin baik model tersebut.

Tabel II.1 Hasil Penelitian Terkait Neural Network

No	Penulis	Judul Penelitian	Metode	Kekurangan	Kelebihan	Akurasi
1.	(Zajmi et al., 2018)	Concepts, Methods, and Performances of Particle Swarm Optimization, Backpropagation, and Neural Networks	Artificial Neural Network Backpropagation dan Particle Swarm Optimization	Penelitian ini hanya mengkaji dan memberikan ulasan singkat tentang konsep, metode, dan kinerja Particle Swarm Optimization, Backpropagation, dan Neural Networks	Memberikan ulasan singkat tentang konsep, metode, dan kinerja Particle Swarm Optimization, Backpropagation, dan Neural Networks. Kesimpulannya, studi tentang Jaringan Syaraf Tiruan dan metodenya menjanjikan dan berkelanjutan, terutama studi perbaikan yang diperlukan di sisi kesalahan pembelajaran dan akurasi klasifikasi.	Hasil analisis diperoleh bahwa peningkatan kedua kategori ini merupakan bagian besar untuk solusi Neural Networks untuk teknologi bidang baru yang muncul seperti Deep Learning in Medicine, Cloud Computing, dan bahkan dalam Keamanan Informasi

No	Penulis	Judul Penelitian	Metode	Kekurangan	Kelebihan	Akurasi
2.	(Mutiar a S. Simanj untak, Wanay umini, Rika Rosnell y, 2019)	The Activity Activation Function Of Multilayer Perceptron - Based Cardiac Abnormalities	MLP	Menganalisis kemampuan jaringan MLP untuk mengklasifikasi kan aktivitas deteksi kelainan jantung dengan kinerja akurasi tinggi dan pembacaan MSE rendah. Belum optimal karena perlu ada parameter pembandingan.	Program yang dikembangkan mampu mendeteksi aktivitas kelainan jantung melalui aplikasi Multilayer Perceptron (MLP). Sejumlah data sinyal detak jantung dari elektrokardiogram (EKG) akan digunakan dalam makalah ini untuk melatih dan menguji kinerja jaringan MLP. MLP dilatih dengan beberapa teknik yaitu Backpropagation (BP), Bayesian Regularity (BR), dan Levenberg- Marquardt (LM).	Hasil evaluasi pada 30 citra sediaan darah tipis menunjukkan bahwa metode yang diusulkan mendapatkan nilai akurasi sebesar 64%.
3	(Purba et al., 2020)	The optimalization of backpropagation neural networks to simplify decision making	Backpro pagation	Pada pengujian data atau tarining data masih menggunakan tray-error dalam membaut arsitektur terbaik untuk emndapatkan akurasi terbaik.	Proses penyederhanaan model dengan mengambil inti masalah yang paling esensial tanpa melibatkan semua masalah yang konkrit. Pembuatan modeling masalah kompleks di mana pengambil keputusan mencoba menyederhanakan masalah rumit.	Hasil yang dimaksud adalah sebagai berikut: Untuk prediksi ditolak, jika output bernilai 0 sampai dengan 0,3. Untuk memprediksi dapat diterima, jika output antara 0,56 hingga 1. Dengan demikian output kecerdasan Andre adalah "TERIMA" karena 0,9511 berada di antara batas 0,56 hingga 1. Sesuai dengan target yang diharapkan.
4	(Yan et	Trustworthy	Adaptive	Perlu	Merancang	Dalam HCA-

No	Penulis	Judul Penelitian	Metode	Kekurangan	Kelebihan	Akurasi
	al., 2020)	Network Anomaly Detection Based on an Adaptive Learning Rate and Momentum in IIoT	Learning Rate and Momentum in IIoT	menggunakan HCA-BAGDALRM, hiperparameter dan digunakan untuk mengontrol masalah cold-start dari algoritma. Untuk meningkatkan hasil kami, kami menyetel hiperparameter ini.	algoritme klasifikasi engsel baru berdasarkan penurunan gradien mini-batch dengan kecepatan dan momentum pembelajaran adaptif (HCA-MBGDALRM) untuk meminimalkan efek serangan keamanan. Industrial Internet of Things (IIoT) menghadirkan kenyamanan untuk industri, itu juga membawa masalah keamanan.	BAGDALRM, hiperparameter dan digunakan untuk mengontrol masalah cold-start dari algoritma. Untuk meningkatkan hasil kami, kami menyetel hiperparameter ini. Hasilnya dijelaskan secara rinci pada Gambar 11 dan Gambar 12. Ketika = 0,9, algoritma mencapai konvergensi yang cepat dan akurasi yang tinggi, dan osilasi yang cukup besar tidak terjadi. Ketika = 0,7, tingkat konvergensi dari algoritma beresilasi hebat dan tidak konvergen ke hasil yang sama seperti yang diperoleh dengan = 0,9. Oleh karena itu, kami memilih = 0,9 sebagai parameter model
5	(Windarto, Naam, et al., 2020)	Model Arsitektur Neural Network Dengan Kombinasi K-Medoids dan Backpropagation pada kasus Pandemi Covid-19 di Indonesia	Arsitektur Neural Network Dengan Kombinasi K-Medoids dan Backpropagation	Perlu dikembangkan dengan melakukan optimasi menggunakan metode algoritma genetika untuk menghasilkan pemilihan model arsitektur yang lebih baik.	Membuat model prediksi pada arsitektur neural network terbaik dengan bisa di kombinasikan antara metode k-medoids dan backpropagation pada kasus pandemi COVID-19 di Indonesia.	Hasil dari pemetaan kluster dilanjutkan ke metode backpropagation untuk memprediksi hasil akurasi dari kluster yang ada. Dengan menggunakan model arsitektur terbaik 3-2-1 diperoleh nilai akurasi 94,17% dengan learning_rate= 0.696. Hasil pemetaan kluster

No	Penulis	Judul Penelitian	Metode	Kekurangan	Kelebihan	Akurasi
						diperoleh 9 provinsi berada di klaster tinggi (C1= zona merah), 3 provinsi berada di klaster waspada (C2= zona kuning) dan 22 provinsi berada di klaster rendah (C3= zona hijau).
6	(Mardiyah, 2020)	Implementasi Deep Learning untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Kebun dan Sawah	Convolutional Neural Network (CNN)	Citra yang digunakan perlu di tambah data citra kebun dan sawah lebih banyak lagi untuk memperoleh akurasi yang akurat. Dan perlu parameter sebagai pembanding guna untuk mendapatkan aritektur yang lebih baik lagi.	Metode CNN yang masukkan (input data) berupa citra. yang menghasilkan sebuah pola dari beberapa bagian citra yang nantinya akan lebih mudah untuk diklasifikasikan dengan pendekatan deep learning untuk optimasi pengklasifikasiannya.	Dari hasil pengujian untuk dapat mengklasifikasi kebun dan sawah, dengan tujuan untuk membedakan karakteristik kebun dan sawah. Berdasarkan hasil klasifikasi diperoleh hasil akurasi testing sebesar 75%.
7	(Wanto et al., 2020)	Analisis Penurunan Gradien dengan Kombinasi Fungsi Aktivasi pada Algoritma JST untuk Pencarian Akurasi Terbaik	Backpropagation	Pada penggunaan kombinasi Gradient descent backpropagation (traingd), Gradient descent with momentum backpropagation (traingdm), Gradient descent with adaptive learning rate backpropagation (traingda), dan Gradient descent with momentum	Backpropagation dengan kombinasi Gradient descent backpropagation (traingd), Gradient descent with momentum backpropagation (traingdm), Gradient descent with adaptive learning rate backpropagation (traingda), dan Gradient descent with momentum and adaptive	Dari hasil analisa, fungsi training terbaik adalah traingda dengan model arsitektur 6- 15-20-1 yang menghasilkan tingkat akurasi 91% dan MSE Testing 0,000731529 (lebih kecil dibandingkan metode lainnya)

N o	Penulis	Judul Penelitian	Metode	Kekurangan	Kelebihan	Akurasi
				and adaptive learning rate belum efektif jika pemilihan arsitektur masih menggunakan random.	learning rate backpropagation (traingdx). Dan Fungsi training akan dikombinasikan dengan fungsi aktivasi (fungsi transfer) sigmoid bipolar (tansig), transfer linier (purelin) dan sigmoid biner (logsig).	
8	(Redho Islami 2020)	Implementasi Deep Learning Dalam Mendeteksi Penyakit Menggunakan Convolutional Neural Network Dan Pytorch	Deep Learning	Klasifikasi menggunakan dataset dengan jumlah dan ukuran yang besar, maka membutuhkan waktu eksekusi yang cukup lama saat menggunakan central processing unit (CPU). Dan pemanfaatan learning rate belum maksimal.	Pemanfaatan computer vision dan penggunaan metode deep learning sangat baik dan sangat berguna bagi bidang kesehatan dengan menggunakan convolutional neural network dan Pytorch dalam mendeteksi sebuah objek penyakit.	Hasil testing model yang diperoleh dari uji validasi dengan kategori polyps nilai akurasi sebesar 83% - 90%, normal-pylorus nilai akurasi sebesar 86% - 92%, dyed-lifted-polyps nilai akurasi sebesar 83% - 97%, esophagitis nilai akurasi sebesar 82% - 92%, dan dyed-resection-margins nilai akurasi sebesar 82% - 93%
9	(Yanto et al., 2021)	Indentifikasi Pola Aksara Arab Melayu Dengan Jaringan Syaraf Tiruan Convolutional Neural	Convolut ional Neural Network (CNN)	Pada pengenalan pola tulisan arab sangat diperlukan menggunakan data yang cukup besar untuk dapat mengenali pola aksra tersebut. Sehingga untuk traning data	CNN mampu mengenali pola tulisan Aksara dengan baik ketika dituliskan dalam kertas. Hasil inputan citra berupa tulisan tangan yang dituliskan di kertas kemudian di scanner dalam	Persentase akurasi hasil metode Convolutioal Neural Network (CNN) karena sangat bagus dalam mengeskraksi fitur pola citra gambar dengan pengujian sebanyak 10 citra buah jeruk yang terbagi menjadi 5 citra jeruk bagus dan 5 citra

No	Penulis	Judul Penelitian	Metode	Kekurangan	Kelebihan	Akurasi
				cukup lama karena tidak adanya pengambilan momentum error dan perubahan learning rate untuk menambah kecepatan training data.	bentuk format gambar JPEG. Pengujian dengan memanfaatkan software Pytorch dengan menerapkan algoritma jaringan syaraf tiruan Convolutioal Neural Network (CNN).	jeruk busuk sebesar 96% untuk training 92% untuk testing yang dinilai telah mampu melakukan klasifikasi kelayakan buah jeruk manis dengan sangat baik. Grafik hasil akurasi testing nya adalah 0.92 atau 92%.
10	(Irnanda et al., 2022)	Optimasi Particle Swarm Optimization Pada Peningkatan Prediksi dengan Metode Backpropagation Menggunakan Software RapidMiner	Particle Swarm Optimization	Melakukan optimasi bobot jaringan saraf tiruan untuk menghindari terjadinya lokal minimum atau seringkali terjebak pada kondisi lokal minimum, masalah waktu pelatihan yang lama untuk mencapai konvergen dan proses penentuan parameter (learning rate dan momentum) yang tepat dalam proses pelatihan.	Penggunaan Particle Swarm Optimization (PSO) yang untuk menyelesaikan permasalahan optimisasi dengan meningkatkan prediksi cukup baik dan RMSE (Root Mean Squared Error) mengecil dari backpro standar.	Hasil uji coba yang dilakukan dengan software Rapid Miner model arsitektur terbaik adalah model 2-9-1 dengan Backpropagation + particle swarm optimization jumlah RMSE 0.055 +/- 0.000 (lebih baik 0.001).

Seperti yang telah dipaparkan dalam sistematika penulisan pada bab sebelumnya, bab ini akan membahas tentang dasar-dasar teori penunjang yang berkaitan dengan topik pada penelitian ini. Adapun teori-teori yang akan dipaparkan adalah sebagai berikut :

II.2 Kecerdasan Buatan

Kecerdasan Buatan atau Intelegensi Artifisial (bahasa Inggris: *Artificial Intelligence* atau hanya disingkat *AI*) didefinisikan sebagai kecerdasan *entitas* ilmiah. Sumbangan terbesar di bidang *AI* diawali oleh tulisan dari Alan Turing pada tahun 1950 berjudul *Computing Machinery and Intelligence* mendiskusikan syarat sebuah mesin di anggap cerdas. Kecerdasan yang dimaksud di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil dan mampu mengambil keputusan seperti yang dilakukan manusia. Kecerdasan buatan diciptakan dan dimasukkan ke dalam suatu mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan manusia. Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar, permainan komputer (*games*), *logika fuzzy*, Jaringan Saraf Tiruan dan Robotika (Handayani et al., 2022).

Agar mesin bisa cerdas (bertindak seperti dan sebaik manusia) maka harus diberi bekal pengetahuan dan mempunyai kemampuan untuk menalar. Dua bagian utama yg dibutuhkan untuk aplikasi kecerdasan buatan :

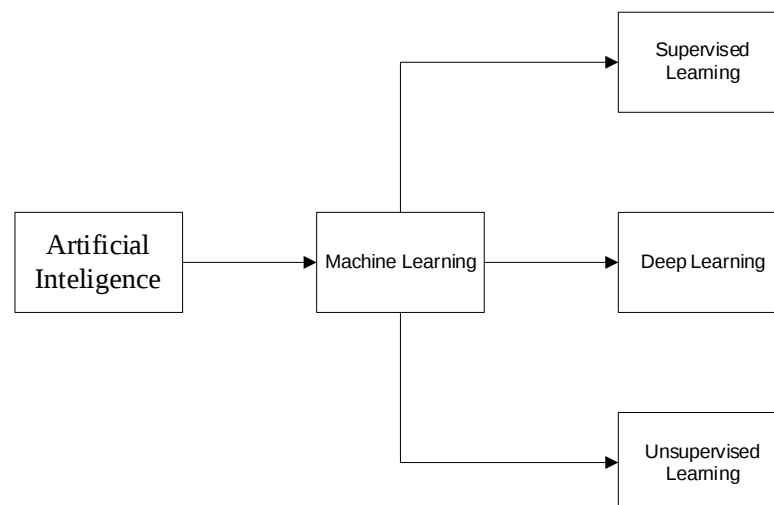
1. Basis pengetahuan (*knowledge base*): berisi fakta-fakta, teori, pemikiran dan hubungan antara satu dengan lainnya.
2. Motor inferensi (*inference engine*) : kemampuan menarik kesimpulan berdasarkan pengetahuan

Beberapa kegiatan yang dirancang untuk dilakukan adalah pengenalan suara, pembelajaran, perencanaan dan pemecahan masalah (Ziyad, 2019). *Artificial Intelligence* (AI) adalah topik yang lebih luas yang mencakup *Machine Learning* (ML) dan *Deep Learning* (DL) (Priyatno, 2021) .

II.3 Machine Learning

Machine Learning adalah cabang dari kecerdasan buatan (AI) dan ilmu komputer yang berfokus pada penggunaan data dan algoritma untuk meniru cara manusia belajar, secara bertahap meningkatkan akurasi (Education, 2021). *Machine Learning* dapat didefinisikan sebagai aplikasi komputer dan algoritma matematika yang diadopsi dengan cara pembelajaran yang berasal dari data dan menghasilkan prediksi di masa yang akan datang (Huang et al., 2006). Jika *Artificial intelligence* fokus pada membuat komputer yang cerdas, maka untuk mencapai kecerdasan tersebut digunakan *Machine Learning* (样本量估算-Machine Learning.Pdf, n.d.)

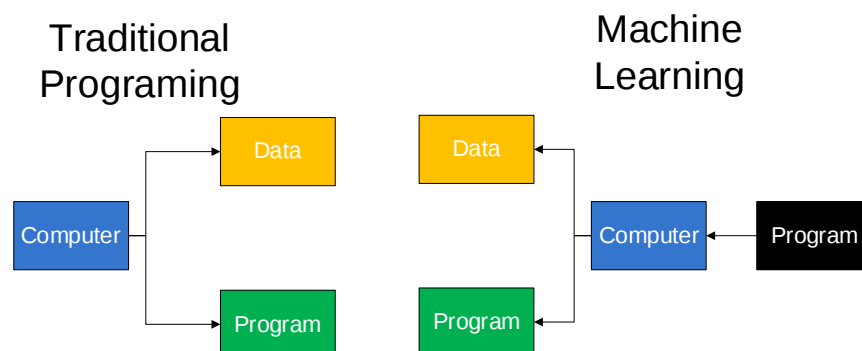
Penelitian terkini mengungkapkan bahwa *Machine Learning* terbagi menjadi tiga kategori: *Supervised Learning*, *Unsupervised Learning* dan *Reinforcement Learning* (Somvanshi et al., 2017). Skema keterkaitan *artificial intelligence* dan *Machine Learning* dapat dijelaskan dalam Gambar 1.



Gambar II.1. Skema Artificial Intelligence dan *Machine Learning*

Dari gambar diatas dapat kita lihat bahwa peranan *Machine Learning* (ML) adalah melakukan proses dalam mencapai kecerdasan buatan (AI). Dapat dijelaskan pula bahwa Teknik yang digunakan oleh *Supervised Learning* adalah metode klasifikasi di mana kumpulan data sepenuhnya diberikan label untuk mengklasifikasikan kelas yang tidak dikenal. Sedangkan teknik *Unsupervised Learning* sering disebut cluster dikarenakan tidak ada kebutuhan untuk pemberian label dalam kumpulan data dan hasilnya tidak mengidentifikasi contoh di kelas yang telah ditentukan (Thupae et al., 2018). Sedangkan *Reinforcement Learning* biasanya berada antara *Supervised Learning* dan *Unsupervised Learning* (Financial Stability Board, 2017), teknik ini bekerja dalam lingkungan yang dinamis di mana konsepnya harus menyelesaikan tujuan tanpa adanya pemberitahuan dari komputer secara eksplisit jika tujuan tersebut telah tercapai (Das & Nene, 2018). *Supervised learning* memiliki beberapa algoritma populer seperti *Naive Bayesian*, *Metode Rocchio*, *Decision Tree*, *k-Nearest Neighbor*, *Neural Network* (Darujati, 2016).

Berikut merupakan gambaran umum untuk *Machine Learning* dibandingkan dengan pemrograman yang lainnya (tradisional) (Danukusumo, 2017).



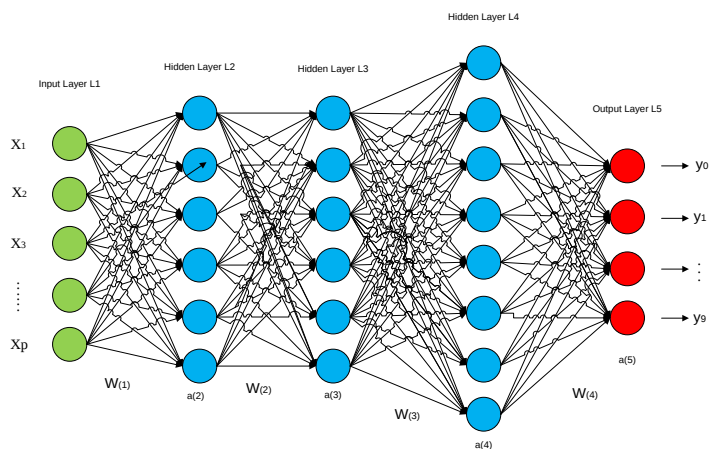
Gambar II.2. Perbedaan Traditional Programming dan *Machine Learning*

(sumber : *Machine Learning* vs Traditional Programming - Avenga, 2021)

Dari gambar diatas dapat kita lihat bahwa *Traditional Programming* membutuhkan Data sebagai input dan program yang sudah dibuat untuk menghasilkan output sedangkan *Machine Learning* hanya membutuhkan Data sebagai input dan contoh output yang diinginkan, dan sistem akan mencari dengan sendirinya program perhitungannya. Agar mesin dapat belajar, dibutuhkan 2 jenis data yaitu : Data *Training* berisi data fakta yang ada dan Data *Test* berisi data yang ingin diprediksi (Pengenalan Data Science, 2019)

II.4 Deep Learning

Deep Learning adalah bagian dari *NeuralNetwork* yang memiliki memiliki arsitektur lebih kompleks dan lebih banyak jumlah *layer* yang digunakan, sehingga diharapkan mampu menangani permasalahan yang lebih rumit dengan lebih banyak data dan mampu menghasilkan akurasi pada output yang cukup baik.



Gambar II.3. Deep Learning

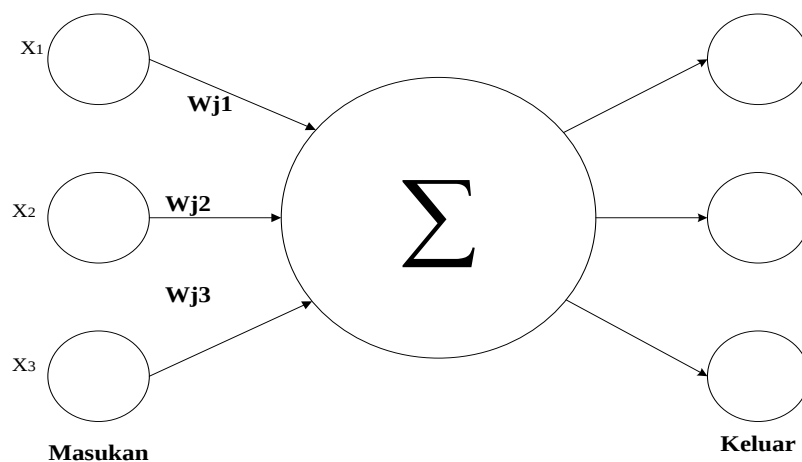
(Sumber : Nuli Giarsyani, Ahmad Fathan Hidayatullah 2020)

II.5 Jaringan Saraf Tiruan

Jaringan saraf tiruan (JST) (*artificial neural network (ANN) / simulated neural network (SNN) / neural network(NN)*) merupakan jaringan yang terdiri atas sekelompok unit pemroses kecil yang dimodelkan berdasarkan jaringan saraf manusia. JST tercipta sebagai suatu generalisasi model matematis dari pemahaman manusia (*human cognition*) yang didasarkan atas asumsi pemrosesan informasi terjadi pada elemen sederhana yang disebut *neuron*, sinyal mengalir diantara sel saraf/*neuron* melalui suatu sambungan penghubung, setiap sambungan penghubung memiliki bobot yang bersesuaian. Bobot ini untuk menggandakan/ mengalihkan sinyal yang dikirim, setiap sel saraf akan menerapkan fungsi aktivasi terhadap sinyal hasil penjumlahan berbobot yang masuk kepadanya untuk menentukan sinyal keluarannya.

Model Neuron

Satu sel saraf terdiri dari 3 bagian, yaitu : fungsi penjumlahan (*summing function*), fungsi aktivasi (*activation function*), dan keluaran (*output*).



Gambar II.4. Model Neuron

(Sumber : (Prayudha et al., 2019)

Jika dilihat, *neuron* buatan diatas mirip dengan sel *neuron* biologis. Informasi (*input*) akan dikirim ke *neuron* dengan bobot tertentu. *Input* ini akan diproses oleh suatu fungsi yang akan menjumlahkan nilai-nilai bobot yang ada. Hasil penjumlahan kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap *neuron*. Apabila *input* tersebut melewati suatu nilai ambang tertentu, maka *neuron* tersebut akan diaktifkan, jika tidak, maka *neuron* tersebut akan mengirimkan *output* melalui bobot-bobot *output*nya ke semua *neuron* yang berhubungan dengannya.

Sehingga dapat disimpulkan bahwa *neuron* terdiri dari 3 elemen pembentuk yaitu :

- a. Himpunan unit-unit yang dihubungkan dengan jalur koneksi. Jalur-jalur tersebut memiliki bobot yang berbeda-beda. Bobot yang bernilai positif akan memperkuat sinyal dan yang bernilai negatif akan memperlemah sinyal yang dibawa.
- b. Suatu unit penjumlahan yang akan menjumlahkan *input* - *input* sinyal yang sudah dikalikan dengan bobotnya.
- c. Fungsi aktivasi yang akan menentukan apakah sinyal dari *input neuron* akan diteruskan ke *neuron* lain atau tidak.

Jadi dengan tingkat kemampuan yang sangat baik, beberapa aplikasi jaringan syaraf tiruan sangaat cocok untuk diterapkan pada :

1. *Klasifikasi*, memilih suatu input data ke dalam satu kategori tertentu yang diterapkan.

2. *Asosiasi*, menggambarkan suatu objek secara keseluruhan hanya dengan sebuah bagian dari objek lain.
3. *Self organizing*, kemampuan untuk mengolah data-data input tanpa harus memiliki data sebagai target.
4. *Optimasi*, menemukan suatu jawaban atau solusi yang paling baik sehingga seringkali dengan meminimalisasikan suatu fungsi biaya (*optimizer*).

II.5.1 *Activation Function*

Tujuan *activation function* adalah memperkenalkan *non-linearitas* dalam *output Neuron* (Priyatno, 2021). Hal tersebut penting karena sebagian besar data pada dunia nyata adalah non-linear. Fungsi aktivasi merupakan fungsi yang digunakan pada jaringan saraf untuk mengaktifkan atau tidak mengaktifkan *Neuron*. Karakteristik yang harus dimiliki oleh fungsi aktivasi jaringan perambatan balik antara lain harus kontinu, terdiferensialkan, dan tidak menurun secara monotonis (Julpan et al., 2015). Fungsi aktivasi mendefinisikan keluaran dari suatu *Neuron/ node* yang diberikan suatu masukan atau sekumpulan masukan (output dari beberapa *Neuron*). Ini meniru stimulasi *Neuron* biologis (Pere, 2020).

Jika fungsi aktivasi tidak digunakan dalam jaringan saraf maka sinyal output hanya akan menjadi fungsi linier sederhana yang hanya polinomial satu derajat. Meskipun persamaan linier sederhana dan mudah untuk dipecahkan tetapi kompleksitas persamaannya terbatas dan tidak memiliki kemampuan untuk mempelajari dan mengenali *mapping* data secara kompleks. *Neural network* tanpa fungsi aktivasi bertindak sebagai Model Regresi Linear dengan kinerja dan daya

terbatas. Sangat diharapkan bahwa *Neural Network* tidak hanya belajar dan menghitung fungsi linier tetapi melakukan tugas yang lebih rumit dari itu seperti pemodelan jenis data yang rumit seperti gambar, video, audio, pidato, teks, dll (Sharma et al., 2020).

Dalam Penelitian ini beberapa *activation function* yang akan dibahas adalah sebagai berikut:

II.5.2 Fungsi Aktivasi *Sigmoid*

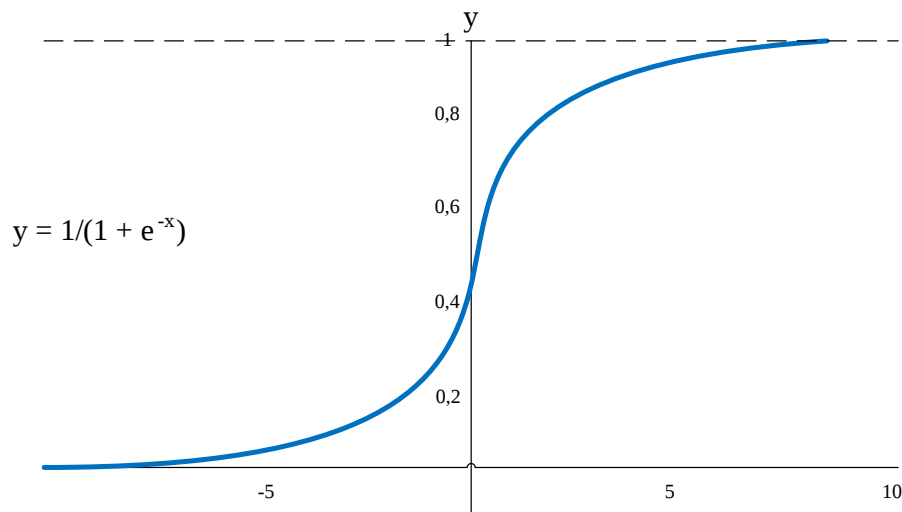
Activation function ini akan menerima input bernilai nyata dan menekan nilai tersebut kedalam kisaran 0 dan 1 (Priyatno, 2021). Fungsi *sigmoid*, juga disebut fungsi logistik, adalah fungsi kontinu berbentuk “S” dengan domain atas semua R . Namun, jangkauannya hanya di atas (0, 1) (Kyurkchievand Markov 2015). Fungsi *Sigmoid*, juga disebut sebagai *logistic function*, fungsi ini dapat didefinisikan dalam persamaan :

$$\sigma : R \rightarrow (0, 1) \quad (1)$$

didefinisikan sebagai

$$\sigma(x) = \frac{1}{(1+e^{-x})} \quad (2)$$

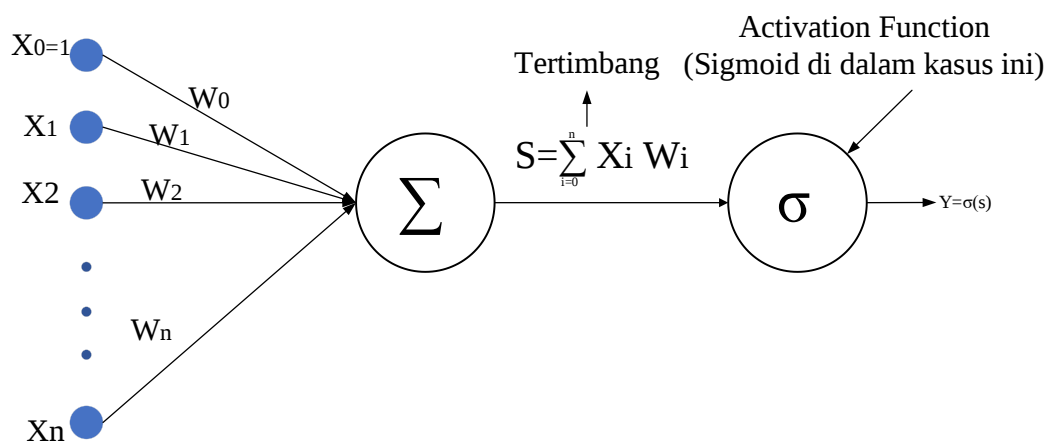
Grafik fungsi sigmoid adalah kurva berbentuk S seperti yang ditunjukkan oleh garis hijau pada grafik di bawah ini. Gambar tersebut juga menunjukkan grafik turunan dalam warna pink. Ekspresi untuk turunan, bersama dengan beberapa properti penting ditampilkan di sebelah kanan.



Gambar II.5. Grafik Fungsi Sigmoid

Fungsi sigmoid juga disebut fungsi squashing karena domainnya adalah himpunan semua bilangan real, dan jangkauannya adalah (0, 1). Oleh karena itu, jika input ke fungsi tersebut adalah bilangan negatif yang sangat besar atau bilangan positif yang sangat besar, keluarannya selalu antara 0 dan 1. Hal yang sama berlaku untuk bilangan apa pun antara $-\infty$ dan $+\infty$.

Sigmoid Sebagai Fungsi Aktivasi Dalam *Neural Network* digunakan sebagai fungsi aktivasi pada jaringan syaraf tiruan.



Gambar II.6. Unit sigmoid dalam jaringan saraf

Dari gambar II.14 dapat kita lihat peran fungsi aktivasi dalam satu lapisan jaringan saraf. Jumlah input yang tertimbang dilewatkan melalui fungsi aktivasi dan output ini berfungsi sebagai input ke lapisan berikutnya.

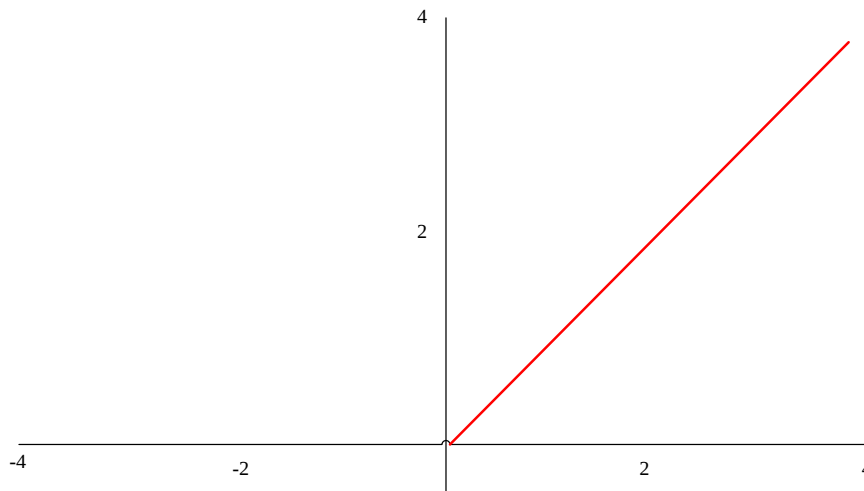
Ketika fungsi aktivasi untuk *Neuron* adalah fungsi sigmoid, ini pasti bahwa output dari unit ini akan selalu antara 0 dan 1. Juga, karena sigmoid adalah fungsi *non-linier*, output dari unit ini akan menjadi fungsi *nonlinier* dari penjumlahan input *weight*. *Neuron* yang menggunakan fungsi sigmoid sebagai fungsi aktivasi disebut sebagai *unit sigmoid*.

II.5.3 Fungsi Aktivasi *ReLU*

ReLU adalah fungsi aktivasi yang diperkenalkan oleh (Hahnioser et al., 2000), yang memiliki dasar biologis dan matematis yang kuat. Pada tahun 2011, fungsi aktivasi tersebut digunakan lebih untuk meningkatkan pelatihan *deep neural network*. *ReLU* bekerja dengan nilai ambang batas pada 0.

ReLU adalah singkatan dari *Rectified Linear Unit*. Pada intinya ReLU hanya membuat pembatas pada bilangan nol. Artinya apabila $x \leq 0$ maka $x = 0$ dan apabila $x > 0$ maka $x = x$. Persamaan ReLU seperti pada persamaan (1) grafiknya seperti pada Gambar II.15:

$$f(x) = \max(0, x). \quad (3)$$



Gambar II.7. Rectified Linear Unit (ReLU)

Dari gambar diatas dapat kita lihat bahwa fungsi aktivasi menghasilkan 0 sebagai output ketika $x < 0$, dan kemudian menghasilkan linier dengan kemiringan 1 ketika $x > 0$. Biasanya fungsi aktivasi ini *ReLU* tidak hanya digunakan sebagai fungsi aktivasi di setiap *hidden lalyer* dari jaringan saraf, tetapi juga sebagai klasifikasi berfungsi pada *layer* terakhir. Oleh karena itu, kelas yang diprediksi untuk pengklasifikasi *ReLU* adalah $\hat{y}$ (Agarap, 2018),

$$\hat{y} = \arg \max$$

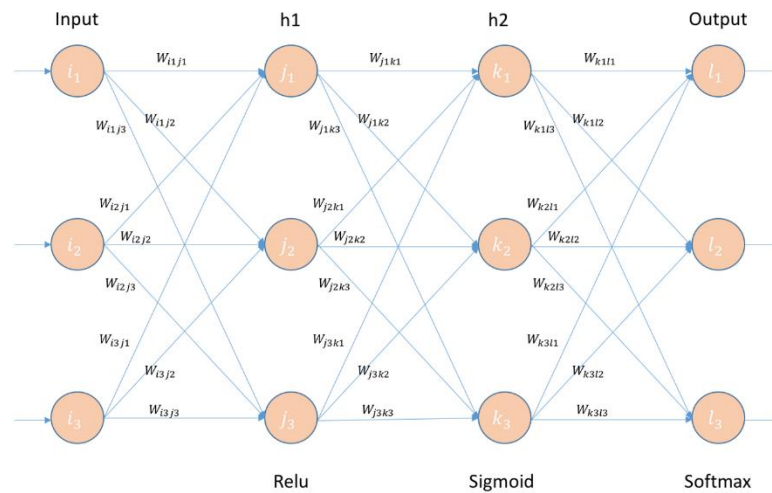
(4)

$$i \in 1, \dots, N \max(0, o)$$

(5)

ReLU lebih efisien daripada fungsi lain karena secara keseluruhan *neuron* tidak diaktifkan pada saat yang sama, melainkan sejumlah *neuron* diaktifkan pada suatu waktu. Dalam beberapa kasus bila nilai gradien adalah nol maka bobot dan bias tidak diperbarui selama langkah *backpropagation* dalam pelatihan jaringan

saraf. *ReLU* Sebagai Fungsi Aktivasi Dalam *neural network* digunakan sebagai fungsi aktivasi pada jaringan syaraf tiruan.



Gambar II.8. Contoh *ReLU* pada *Backpropagation*

Dari gambar diatas dapat kita lihat bahwa fungsi aktivasi *ReLU* digunakan pada saat proses konvolusi. Aktivasi *ReLU* tersebut diterapkan untuk meningkatkan *non-linieritas* pada citra yang akan menggantikan nilai negatif menjadi 0 dan mempertahankan nilai positif (Priyatno, 2021).

II.6 Model Jaringan *Backpropagation*

Model jaringan *backpropagation* merupakan suatu teknik pembelajaran atau pelatihan *supervised learning* yang paling banyak digunakan. Metode ini merupakan salah satu metode yang sangat baik dalam menangani masalah pengenalan pola-pola kompleks. Didalam jaringan *backpropagation*, setiap unit yang berada di lapisan *input* berhubungan dengan setiap unit yang ada di lapisan tersembunyi. Setiap unit yang ada di lapisan tersembunyi terhubung dengan setiap

unit yang ada di lapisan *output*. Jaringan ini terdiri dari banyak lapisan (*multilayer network*). Ketika jaringan ini diberikan pola masukan sebagai pola pelatihan, maka pola tersebut menuju unit-unit lapisan tersembunyi untuk selanjutnya diteruskan pada unit-unit dilapisan keluaran. Kemudian unit-unit lapisan keluaran akan memberikan respon sebagai keluaran jaringan saraf tiruan. Saat hasil keluaran tidak sesuai dengan yang diharapkan, maka keluaran akan disebarkan mundur (*backward*) pada lapisan tersembunyi kemudian dari lapisan tersembunyi menuju lapisan masukan.

Tahap pelatihan ini merupakan langkah untuk melatih suatu jaringan saraf tiruan, yaitu dengan cara melakukan perubahan bobot, sedangkan penyelesaian masalah akan dilakukan jika proses pelatihan tersebut telah selesai, fase ini disebut fase pengujian.

$$X^1 = \frac{0,8+(x-a)}{b-a} + 0,1 \quad (2.1)$$

Di mana :

a = data minimum

b = data maksimum

x = nilai asli dari data

x^1 = nilai transformasi dari data

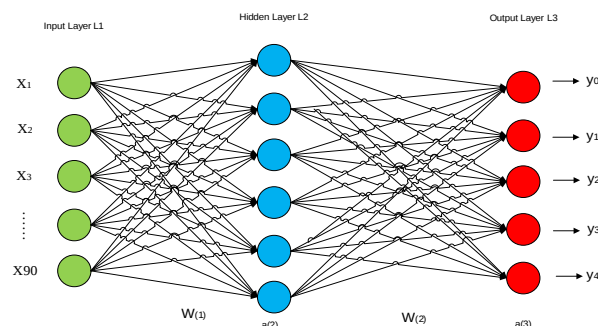
II.6.1 Arsitektur Jaringan Backpropagation

Setiap unit dari *layer input* pada jaringan *backpropagation* selalu terhubung dengan setiap unit yang berada pada *layer* tersembunyi, demikian juga

setiap unit *layer* tersembunyi selalu terhubung dengan unit pada *layer output*.

Jaringan *backpropagation* terdiri dari banyak lapisan (*multilayer network*) yaitu:

- Lapisan *input* (1 buah), yang terdiri dari 1 hingga n unit *input*.
- Lapisan tersembunyi (minimal 1 buah), yang terdiri dari 1 hingga p unit tersembunyi.
- Lapisan *output* (1 buah), yang terdiri dari 1 hingga m unit *output*.



Gambar II.9. Arsitektur Jaringan Backpropagation Standar

II.6.2 Pelatihan Jaringan Backpropagation

Aturan pelatihan jaringan *backpropagation* terdiri dari 2 tahapan, feedforward dan backward propagation. Pada jaringan diberikan sekumpulan contoh pelatihan yang disebut set pelatihan. Set pelatihan ini digambarkan dengan sebuah vektor *feature* yang disebut dengan vektor *input* yang diasosiasikan dengan sebuah *output* yang menjadi target pelatihannya. Dengan kata lain set pelatihan terdiri dari vektor *input* dan juga vektor *output* target. Keluaran dari jaringan berupa sebuah vektor *output* aktual. Selanjutnya dilakukan perbandingan antara *output* aktual yang dihasilkan dengan *output* target dengan cara melakukan pengurangan diantara kedua *output* tersebut. Hasil dari pengurangan berupa *Error*

. *Error* dijadikan sebagai dasar dalam melakukan perubahan dari setiap bobot dengan mempropagationkannya kembali.

Setiap perubahan bobot yang terjadi dapat mengurangi *Error*. Siklus perubahan bobot (*epoch*) dilakukan pada setiap set pelatihan sehingga kondisi berhenti dicapai, yaitu bila mencapai jumlah *epoch* yang diinginkan atau hingga sebuah nilai ambang yang ditetapkan terlampaui.

Algoritma pelatihan jaringan *backpropagation* terdiri dari 3 tahapan yaitu:

- a. Tahap umpan maju (*feedforward*).
- b. Tahap umpan mundur (*backpropagation*).
- c. Tahap pengupdatean bobot dan bias.

Secara rinci algoritma pelatihan jaringan *backpropagation* dapat diuraikan sebagai berikut :

Langkah 0 : Inisialisasi bobot-bobot, konstanta laju pelatihan (α), toleransi *Error* atau nilai bobot (bila menggunakan nilai bobot sebagai kondisi berhenti) atau set maksimal epoch (jika menggunakan banyaknya epoch sebagai kondisi berhenti).

Langkah 1 : Selama kondisi berhenti belum dicapai, maka lakukan langkah ke-2 hingga langkah ke-9.

Langkah 2 : Untuk setiap pasangan pola pelatihan, lakukan langkah ke-3 sampai langkah ke-8.

Langkah 3 : {Tahap I : Umpan maju (*feedforward*)}. Tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi diatasnya.

Langkah 4 : Masing-masing unit di lapisan tersembunyi (dari unit ke-1 hingga unit

ke-p) dikalikan dengan bobotnya dan dijumlahkan serta ditambahkan

$$\text{dengan biasnya. } Z_{inj} = V_{0j} + \sum_{i=1}^n X_i V_{ij} \quad (2.2)$$

kemudian dihitung nilai *output* dengan menggunakan fungsi aktivasi

$$\text{yang digunakan: } z_j = f(z_{in j})$$

dimana fungsi aktivasi yang digunakan adalah fungsi *sigmoid biner*

$$\text{yang mempunyai persamaan: } y = f(x) = \frac{1}{1 + e^{-x}} \quad (2.3)$$

Hasil fungsi tersebut dikirim ke semua *neuron* pada lapisan di atasnya.

Langkah 5 : Masing-masing unit *output* ($y_k, k=1,2,3,\dots,m$) dikalikan dengan bobot dan dijumlahkan serta ditambahkan dengan biasnya.

$$y_{ink} = W_{ok} + \sum_{j=1}^p Z_j W_{jk} \quad (2.4)$$

Langkah 6 : {Tahap II : Umpan mundur (*backward propagation*)}. Masing-masing unit *output* ($y_k, k=1,2,3,\dots,m$) menerima pola target t_k sesuai dengan pola masukan/*input* saat pelatihan dan kemudian informasi kesalahan/*Error* lapisan *output* (δ_k) dihitung. δ_k dikirim ke lapisan di bawahnya dan digunakan untuk menghitung besarnya koreksi bobot dan bias (ΔW_{jk} dan ΔW_{ok}) antara lapisan tersembunyi dengan lapisan *output*.

$$\delta_k = (t_k - y_k) f'(y_{ink}) \quad (2.5)$$

Langkah 7 : Pada setiap unit di lapisan tersembunyi (dari unit ke-1 hingga ke-p; $i=1\dots n; k=1\dots m$) dilakukan perhitungan informasi kesalahan lapisan tersembunyi (δ_j). δ_j kemudian digunakan untuk menghitung besar koreksi bobot dan bias (ΔV_{ji} dan ΔV_{jo}) antara lapisan *input* dan

$$\text{lapisan tersembunyi. } \delta_{inj} = \sum_{k=1} \delta_k w_{kj} \quad (2.6)$$

Kemudian nilai tersebut dikalikan dengan nilai turunan dari fungsi aktivasi untuk menghitung informasi kesalahan:

$$\delta_j = \delta_{inj} f'(z_{inj}) \quad (2.7)$$

Hitung koreksi nilai bobot yang kemudian digunakan untuk memperbaharui v_{ij} :

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (2.8)$$

Dan hitung nilai koreksi bias yang kemudian digunakan untuk memperbaharui nilai

$$\Delta v_{0j} = \alpha \delta_j \quad (2.9)$$

Memperbaharui nilai bobot dan nilai bias:

Langkah 8 : {Tahap III : Pengupdatean bobot dan bias}. Masing-masing unit *output/keluaran* (y_k , $k=1,2,3,\dots,m$) dilakukan pengupdatean bias dan bobotnya ($j=0,1,2,\dots,p$) sehingga menghasilkan bobot dan bias baru. Demikian juga untuk setiap unit tersembunyi mulai dari unit ke-1 sampai dengan unit ke- p dilakukan pengupdatean bobot dan bias.

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta V_{ij} \quad (2.10)$$

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (2.11)$$

Langkah 9 : Uji kondisi berhenti (akhir iterasi).

II.6.3 Pengujian Jaringan Backpropagation

Setelah bobot yang terbaik pada tahap pelatihan didapat, maka nilai pembobot tersebut digunakan untuk mengolah data masukan untuk menghasilkan

keluaran yang sesuai. Hal ini digunakan untuk menguji apakah JST dapat bekerja dengan baik yaitu dapat memprediksi pola data yang telah dilatihkan dengan tingkat kesalahan yang kecil. Notasi yang digunakan dalam algoritma pengujian:

X_i : Unit *input* ke-i

Z_j : *Hidden* unit ke-j

Y_k : Unit *output* ke-k

$v0_j$: Bias untuk *hidden* unit ke-j

v_{ij} : Bobot antara unit *input* ke-i dengan *hidden* unit ke-j

$w0_k$: Bias untuk unit *output* ke-k

W_{jk} : Bobot antara *hidden* unit ke-j dengan unit *output* ke-k

Step 1 Inisialisasi Inisialisasi nilai bobot dan bias sesuai dengan bobot yang dihasilkan pada proses pelatihan

Step 2 Setiap input (X_i , $i = 1, \dots, n$) menyebarkan sinyal *input* pada seluruh *hidden* unit

Step 3 Setiap *hidden* unit (Z_j , $j = 1, \dots, p$) akan menghitung sinyal-sinyal *input* dengan bobot dan biasnya.

$$z_{in_j} = v0_j + \sum_{i=1}^n X_i V_{ij} \quad (2.12)$$

Kemudian dengan menggunakan fungsi aktivasi yang telah ditentukan diperoleh sinyal *output* dari *hidden* unit tersebut.

$$z_j = f(z_{in_j}) \quad (2.13)$$

Step 4 Setiap unit *output* (Y_k , $k = 1, \dots, m$) akan menghitung sinyal-sinyal dari *hidden* unit dengan bobot dan biasnya.

$$y_{in_k} = v0_k + \sum_{j=1}^p Z_j W_{jk} \quad (2.14)$$

Kemudian dengan menggunakan fungsi aktivasi yang telah ditentukan diperoleh sinyal *output* dari unit *output* tersebut.

$$y_k = f(y_{in_k}) \quad (2.15)$$

II.7 Adaptive Learning Rate

Adaptive Learning Rate merupakan pendekatan atau metode yang bertujuan untuk meningkatkan efektifitas dari parameter tingkat pembelajaran atau learning rate, dimana tingkat pembelajaran merupakan parameter yang berfungsi untuk meningkatkan kecepatan belajar dari jaringan backpropagation.

Adaptive learning rate muncul karena penelitian yang dilakukan pada nilai yang konstan pada tingkat pembelajaran menyebabkan metode jaringan backpropagation menjadi tidak efisien, dikarenakan sangat bergantung pada nilai tingkat pembelajaran yang dipilih. Pemilihan tingkat pembelajaran yang tidak tepat akan menyebabkan jaringan sangat lambat mencapai local optima. Karena alasan tersebut maka muncullah pendekatan adaptive learning rate (Jeffrey M Ede, 2020).

Implementasi adaptive learning rate adalah mengganti nilai learning rate yang digunakan dalam koreksi bobot pada jaringan pada tiap iterasi menggunakan persamaan yang diusulkan oleh sebagai berikut.

$$W^{t+1} = W^t - \lambda_t \nabla E(W^t) \quad (2.16)$$

Dimana :

W^{t+1} = Bobot baru untuk iterasi berikutnya (t+1)

W^t = Bobot pada iterasi saat (t)

λ_t = Adaptive Learning Rate

$\nabla E(W^t)$ = Fungsi Error pada bobot iterasi saat (t)

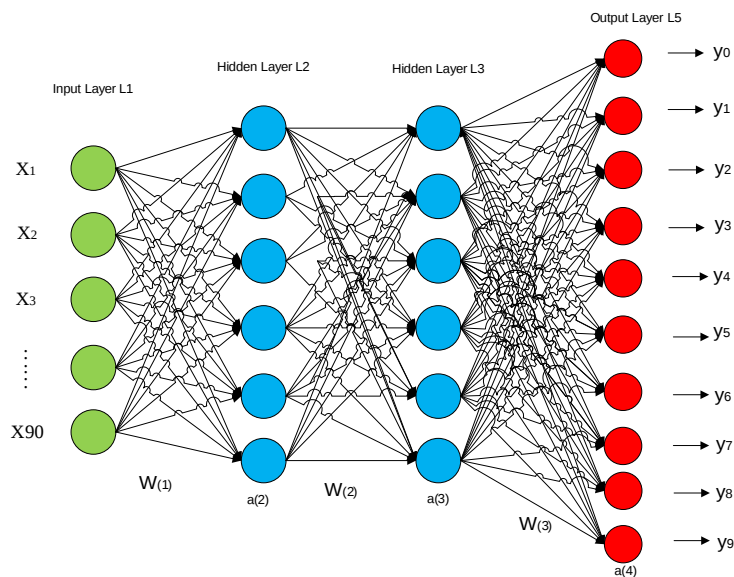
Nilai λ_t = dapat diperoleh dari persamaan berikut.

$$\lambda_t = \left\{ \frac{n_t}{\mu n_{t-1}}, \left| \frac{n_t}{n_{t-1}} \right| \leq \mu \right\} \quad (2.17)$$

Dimana :

$$n_t = \frac{(\delta_{t-1}, \delta_{t-1},)}{(\delta_{t-1}, \psi_{t-1},)}$$

μ = Faktor Pertumbuhan Maksimum



Gambar II.10. Struktur Arsitektur Jaringan Backpropagation dengan Pendekatan Deep Learning

II.8 Perbandingan Backpropagation Konvensional dan Adaptive Learning Rate

Algoritma backpropagation merupakan algoritma yang digunakan dalam mengidentifikasi karakter huruf atau angka yang terdapat pada citra input. Secara garis besar, algoritma backpropagation dibagi menjadi dua tahap, yaitu tahap forward propagation dan backward propagation. Seperti yang telah dijelaskan pada

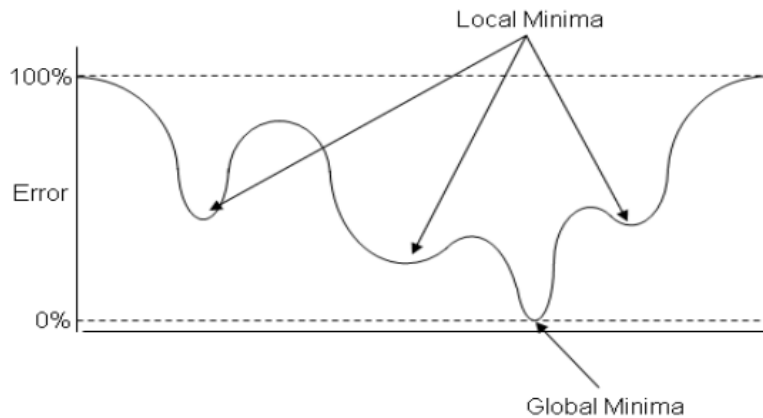
bab landasan teori sebelumnya, tahap forward propagation merupakan tahap dimana data input berupa bit-bit karakter diproses melalui lapisan – lapisan yang terdapat pada jaringan dan menghasilkan output. Jika output tidak sesuai dengan yang diharapkan, maka proses pembelajaran akan dilakukan atau tahap backward propagation.



Gambar II.11. Ilustrasi Pembelajaran Karakter Angka

Backward propagation melakukan koreksi nilai bobot pada tiap lapisan mulai dari lapisan-lapisan output menuju ke lapisan input, koreksi bobot didasarkan atas selisih antara output yang dihasilkan oleh jaringan dengan output yang diharapkan. Setelah proses koreksi bobot selesai maka tahap forward propagation akan di lakukan kembali dengan bobot baru yang telah dikoreksi.

Proses forward dan backward propagation terus dilakukan sampai jaringan mampu menghasilkan output yang diharapkan atau dianggap mampu mengenali karakter yang diberikan. Secara normal metode backpropagation membutuhkan waktu yang sangat lama dalam proses pembelajarannya dimana waktu yang dibutuhkan berbanding lurus dengan banyaknya data pelatihan, yang artinya semakin besar data pelatihan maka akan semakin lama proses pembelajaran yang dilakukan.



Gambar II.12. Ilustrasi Lokal Minimal dan Global Minimal

Suatu jaringan dikatakan sudah belajar jika jaringan tersebut sudah mencapai kondisi global minima atau kondisi dimana jaringan telah mencapai nilai error terendah. Untuk mempercepat jaringan menuju ke global minima maka diperlukan beberapa pengembangan terhadap metode backpropagation tersebut, yang dalam penelitian ini salah satu pengembangan yang diusulkan adalah adaptive learning rate.

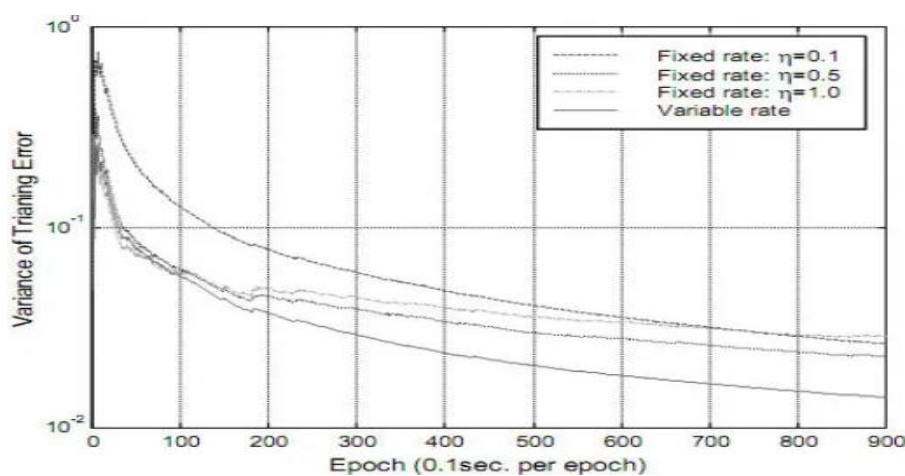
Adaptive learning rate merupakan pengembangan pada algoritma backpropagation. Metode adaptive learning rate dilakukan pada saat koreksi bobot berlangsung. Algoritma backpropagation normal menggunakan parameter learning rate sebagai konstanta, dimana parameter tersebut digunakan terus menerus selama proses iterasi pembelajaran tanpa mengalami perubahan. Dengan menggunakan adaptive learning rate, parameter learning rate atau tingkat pembelajaran terus mengalami perubahan seiring proses pembelajaran yang perubahan nilainya bergantung pada selisih error pada tiap iterasi pembelajaran.

Pada operasi backpropagation normal, parameter learning rate bernilai statis dimana nilai parameter dipilih pada awal pembelajaran dan terus digunakan pada

setiap iterasi tanpa mengalami perubahan.

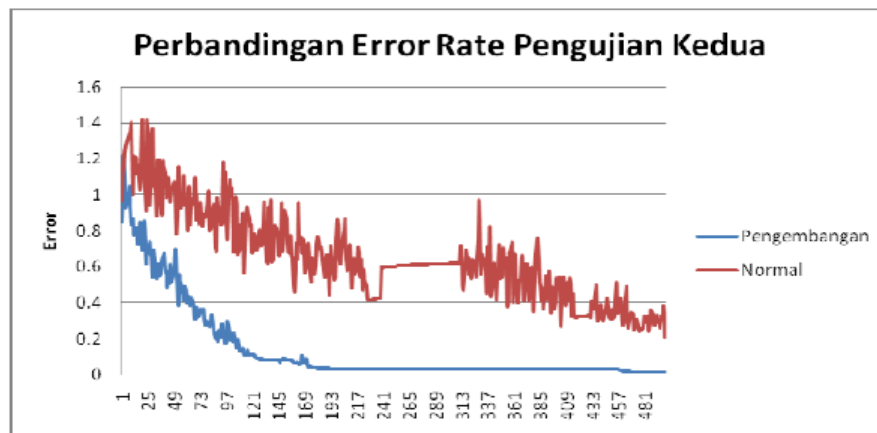
Penggunaan adaptive learning rate memberikan dampak positif dimana proses menuju local optima akan semakin cepat. Dimana dengan tingkat pembelajaran yang selalu menyesuaikan diri dengan nilai error yang dihasilkan mampu memberikan stabilitas dalam proses pembelajaran dibandingkan dengan tingkat pembelajaran yang konstan atau statis.

Pada penelitian yang dilakukan oleh Daohang Sha dan Vladimir mengenai adaptive learning rate, mereka melakukan perbandingan performa dari learning rate statis dan penggunaan adaptive learning rate yang dapat dilihat pada gambar berikut.



Gambar II.13. Perbandingan Laju Error Pada Percobaan Learning Rate

Pada gambar 10 dapat dilihat lajur error dari Variable rate atau dengan kata lain Adaptive learning rate memiliki laju eror yang paling baik dibandingkan dengan Fixed rate atau nilai learning rate statis. Pada penelitian lain yang dilakukan oleh Rajesh et al, penggunaan metode adaptive learning rate memberikan kecepatan laju pembelajaran yang signifikan seperti yang terlihat pada gambar 7 berikut ini.



Gambar II.14. Grafik Perbandingan Implementasi Learning Rate

II.9 Citra Digital

Citra digital didefinisikan sebagai fungsi $f(x,y)$ dua dimensi, dimana x dan y adalah koordinat spasial dan $f(x,y)$ adalah disebut dengan intensitas atau tingkat keabuan citra pada koordinat x dan y (R.C.Gonzales, R.E.Woods,2002). Jika x , y , dan nilai f terbatas dalam diskrit, maka disebut dengan citra digital. Citra digital dibentuk dari sejumlah elemen terbatas, yang masing-masing elemen tersebut memiliki nilai dan koordinat tertentu. Pixel adalah elemen citra yang memiliki nilai yang menunjukkan intensitas warna.

Citra digital diperoleh melalui proses penangkapan atau akuisisi pada objek pada lingkungan nyata melalui perangkat yang dilengkapi dengan sensor optic yang mampu mendeteksi intensitas cahaya dan merepresentasikan intensitas tersebut menjadi nilai diskrit.

II.9.1 Sifat Citra Digital

Citra digital bersifat diskrit yang dapat diolah oleh komputer. Citra ini dapat

dihasilkan melalui kamera digital dan scanner ataupun citra yang telah mengalami proses digitalisasi. Sebuah citra berukuran 150 x 100 pixel dapat dinyatakan dengan matriks yang berukuran sesuai dengan pikselnya atau biasa dinyatakan dalam ukuran $N \times M$ dimana N untuk baris dan M untuk kolom.

II.9.2 Citra Grayscale

Citra yang ditampilkan dari citra jenis ini terdiri atas warna abu-abu, bervariasi pada warna hitam pada bagian yang intensitas terlemah dan warna putih pada intensitas terkuat. Citra grayscale berbeda dengan citra "hitam-putih", dimana pada konteks komputer, citra hitam putih hanya terdiri atas 2 warna saja yaitu "hitam" dan "putih" saja. Pada citra grayscale warna bervariasi antara hitam dan putih, tetapi variasi warna diantaranya sangat banyak. Citra grayscale seringkali merupakan perhitungan dari intensitas cahaya pada setiap pixel pada spektrum elektromagnetik single band.