

**RANGKUMAN MATERI**  
**“PRAKTIKUM ARTIFICIAL INTELLIGENCE”**

*“Disusun sebagai tugas akhir pada mata kuliah Praktikum Artificial Intelligence , dengan Dosen Randi Proska Sandra, M.Sc dan Widya Darwin S.Pd., M.Pd.T”*



*Disusun Oleh :*

**IFDAL LISYUKRI  
21346012**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
JURUSAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI PADANG  
TAHUN 2023**

## **KATA PENGANTAR**

Puji syukur kita hantarkan kehadiran Tuhan Yang Maha Esa, yang telah melimpahkan rahmat dan karunianya kepada kita, sehingga kita dapat menyusun dan menyajikan Rangkuman Materi Praktikum Artificial Intelligence ini dengan tepat waktu. Tak lupa pula kita mengucapkan terima kasih kepada berbagai pihak yangtelah memberikan dorongan dan motivasi. Sehingga makalah ini dapat tersusun dengan baik.

Rangkuman Materi ini dibuat sebagai salah satu syarat untuk memenuhi tugas mata kuliah Praktikum Artificial Intelligence. Kami menyadari bahwa dalam penyusunan makalah ini masih terdapat banyak kekurangan dan jauh dari kata sempurna. Oleh karena itu, Kami mengharapkan kritik dan saran untuk menyempurnakan makalah ini dan dapat menjadi acuan dalam menyusun makalah-makalah selanjutnya. Kami muga memohon maaf apabila dalam penulisan makalah ini terdapat kesalahan kata - kata, pengetikan dan kekeliruan, sehingga membingungkan pembaca dalam memahami maksud penulis.

Adapun Rangkuman ini penulis rangkum dari beberapa sumber yang dapat dipercaya yangsajian penulisnya dengan harapan makalah ini dapatmenambah pengetahuan kita tentang Pendidikan Di Era Teknologi Informasi Dan Komunikasi.Demikian yang dapat kami sampaikan. Akhir kata, semoga makalah ini dapat menambah wawasan bagi kita semua.

Sago, Juni 2023

Ifdal Lisyukri

## DAFTAR ISI

### **Praktikum Artificial Intelligence**

KATA PENGANTAR .....	i
DAFTAR ISI.....	ii
BAB I.....	1
PENDAHULUAN .....	1
A.    Defenisi Kecerdasan Buatan .....	1
B.    Teknik Penyelesaian Kasus dalam Kecerdasan Buatan .....	1
1.    Logika berpredikat (Predicate Logic).....	1
2.    Pohon Keputusan (Decision Tree).....	2
3.    Jaringan Syaraf Tiruan (Neural Networks) .....	3
4.    Algoritma Genetika (Genetic Algorithms) .....	4
5.    Pembelajaran Mesin (Machine Learning) .....	5
6.    Pemrosesan Bahasa Alami (Natural Leanguage Processing/NLP) .....	6
BAB II.....	8
RUANG MASALAH & SISTEM PRODUKSI .....	8
A.    Ruang Masalah .....	8
B.    Sistem Produksi.....	9
BAB III .....	11
METODE-METODE PENCARIAN DALAM KECERDASAN BUATAN.....	11
A.    Pencarian Buta.....	11
1.    Pencarian Breadth-First (Pencarian Luas).....	11
2.    Pencarian Depth-First (Pencarian Dalam) .....	12
3.    Pencarian Uniform Cost (Pencarian Biaya Seragam).....	13
4.    Pencarian Iterative Deepening (Pencarian Dalam yang Dalam dan Bertahap) ....	13
B.    Pencarian Heuristik .....	13
1.    Pencarian Terarah (Informed Search) .....	14
2.    Pencarian Greedy(Greedy Search).....	14
3.    Pencarian A*(A-Star Search).....	15
BAB IV .....	16

REASONING .....	16
A.    Propotional Logic .....	16
1.    Negasi .....	16
2.    Konjungsi.....	16
3.    Disjungsi.....	16
4.    Implikasi .....	17
5.    Biconditional .....	17
B.    First Order Logic (Predicate Calculus) .....	17
1.    Variable .....	17
2.    Konstanta .....	17
3.    Fungsi .....	17
4.    Predikat.....	18
C.    Fuzzy Logic .....	18
1.    Fuzzyness & Probability.....	19
2.    Fuzzy Set .....	19
3.    Fuzzy Logic .....	19
4.    Fuzzy System.....	20
BAB V .....	21
PLANNING( TEKNIK DEKOMPOSISI MASALAH).....	21
A.    Goal Stack Planning (GSP).....	22
B.    Constraint Posting(CP).....	22
BAB VI .....	23
LEARNING .....	23
A.    ID3.....	24
B.    C.45 .....	24
BAB VII.....	26
JARINGAN SYARAF TIRUAN (JST).....	26
A.    Konsep Dasar JST .....	26
1.    Neuron .....	26
2.    Bobot .....	26
3.    Fungsi Aktivasi.....	27
4.    Layer.....	27
5.    Feedforward.....	27

6.	Backpropagation .....	27
B.	Model Syaraf Tiruan (Neuron).....	27
1.	Jaringan Syaraf Tiruan Feedforward (Feedforward Neural Network/FFNN).....	27
2.	Jaringan Syaraf Tiruan Rekurrent (Recurrent Neural Network/RNN).....	28
3.	Jaringan Syaraf Tiruan Konvolusional (Convolutional Neural Network/CNN) ...	28
4.	Jaringan Syaraf Tiruan Long Short-Term Memory (LSTM) .....	28
5.	Jaringan Syaraf Tiruan Gated Recurrent Unit (GRU) .....	28
C.	Aktivasi & Arsitektur Jaringan pada JST .....	28
D.	Supervised Learning & Unsupervised Learning .....	30
1.	Supervised Learning (Pembelajaran Berawasan) .....	30
2.	Unsupervised Learning (Pembelajaran Tanpa Pengawasan).....	30
BAB VIII .....		31
ALGORITMA GENETIKA (AG).....		31
A.	Komponen-komponen AG .....	31
1.	Populasi .....	31
2.	Genotype.....	31
3.	Fungsi Fitness .....	31
4.	Seleksi.....	31
5.	Rekombinasi (Crossover) .....	31
6.	Mutasi .....	32
7.	Penggantian Generasi .....	32
BAB IX .....		33
COGNITIF SCIENCE .....		33
A.	Persepsi.....	33
B.	Pemrosesan Bahasa .....	34
C.	Memori .....	35
D.	Perhatian.....	36
E.	Pengambilan Keputusan dan Kognisi .....	36
F.	Neuropsikologi.....	37

## **BAB I**

### **PENDAHULUAN**

#### **A. Defenisi Kecerdasan Buatan**

Kecerdasan buatan adalah **kecerdasan yang ditanamkan pada suatu sistem yang bisa disetting atau diatur dalam konteks ilmiah** atau bisa disebut juga intelegensi artifisial (bahasa Inggris: Artificial Intelligence) atau dapat disingkat AI, dan didefinisikan sebagai kecerdasan entitas ilmiah.

Kecerdasan Buatan (AI) adalah bidang ilmu komputer yang dikhususkan untuk memecahkan masalah kognitif yang umumnya terkait dengan kecerdasan manusia, seperti pembelajaran, pemecahan masalah, dan pengenalan pola.

#### **B. Teknik Penyelesaian Kasus dalam Kecerdasan Buatan**

##### **1. Logika berpredikat (Predicate Logic)**

Logika predikat adalah cara untuk menyelesaikan argumen yang tidak dapat diselesaikan dengan logika proposisi. Logika predikat adalah cabang logika matematika yang membahas tentang pernyataan dalam bentuk predikat atau fungsi yang menghubungkan objek atau variabel dengan sifat atau hubungan tertentu. Predikat dalam logika predikat digunakan untuk menyatakan properti, hubungan, atau keberadaan suatu objek dalam suatu domain.

Logika predikat memperluas logika proposisional dengan memperkenalkan konsep variabel dan kuantor. Variabel digunakan untuk menggeneralisasi pernyataan dan memperkenalkan abstraksi dalam logika predikat. Kuantor, seperti kuantor universal ( $\forall$ ) dan kuantor eksistensial ( $\exists$ ), digunakan untuk menyatakan pernyataan umum atau pernyataan yang mengandung variabel.

- 1) Dalam logika predikat, terdapat beberapa konsep penting, antara lain:
- 2) Fungsi predikat: Fungsi predikat adalah fungsi yang menghubungkan objek dengan sifat atau hubungan tertentu. Contohnya, " $P(x)$ " bisa berarti "x adalah bilangan prima" atau "x adalah manusia".
- 3) Variabel: Variabel digunakan untuk menggantikan objek dalam pernyataan. Variabel memungkinkan generalisasi dan abstraksi dalam logika predikat. Contohnya, dalam pernyataan " $P(x)$ ", x adalah variabel yang dapat digantikan dengan objek apa pun.
- 4) Kuantor universal ( $\forall$ ): Kuantor universal digunakan untuk menyatakan bahwa suatu pernyataan berlaku untuk semua objek dalam domain tertentu. Misalnya, " $\forall x P(x)$ " berarti "untuk setiap x,  $P(x)$  benar".

- 5) Kuantor eksistensial ( $\exists$ ): Kuantor eksistensial digunakan untuk menyatakan bahwa ada setidaknya satu objek dalam domain tertentu yang memenuhi suatu pernyataan. Misalnya, " $\exists x P(x)$ " berarti "ada x sedemikian sehingga  $P(x)$  benar".
- 6) Logika predikat memiliki aplikasi luas dalam matematika, ilmu komputer, dan filosofi. Ia digunakan untuk mempelajari struktur bahasa alami, penalaran formal, pemrograman logika, dan teori himpunan, di antara banyak bidang lainnya.

## 2. Pohon Keputusan (Decision Tree)

Pohon Keputusan (Decision Tree) adalah salah satu metode atau algoritma dalam data mining dan pembelajaran mesin yang digunakan untuk mengambil keputusan atau membuat prediksi berdasarkan pemodelan struktur berhierarki dalam bentuk pohon. Pohon keputusan menggambarkan serangkaian keputusan dan konsekuensi yang mungkin terjadi berdasarkan fitur-fitur atau atribut-atribut yang diberikan.

Struktur pohon keputusan terdiri dari simpul atau node yang mewakili keputusan, cabang atau edge yang mewakili konsekuensi atau hasil dari keputusan tersebut, dan daun atau leaf node yang mewakili kelas atau nilai prediksi. Pada setiap simpul dalam pohon, dilakukan pemilihan atribut yang paling informatif untuk membagi data menjadi subkelompok yang lebih homogen.

Proses pembentukan pohon keputusan dimulai dengan simpul akar yang merepresentasikan seluruh dataset. Pada setiap tahap, atribut atau fitur yang paling informatif dipilih untuk membagi dataset menjadi subkelompok yang lebih homogen. Pemilihan atribut ini umumnya didasarkan pada metrik seperti Information Gain, Gini Index, atau Chi-square, yang mengukur tingkat ketidakmurnian atau impurity dari dataset.

Setelah atribut pemisah dipilih, dataset akan dibagi menjadi subkelompok berdasarkan nilai-nilai atribut tersebut. Proses ini diulangi secara rekursif untuk setiap subkelompok sampai mencapai kondisi berhenti, misalnya ketika seluruh data dalam subkelompok sudah termasuk dalam satu kelas atau ketika tidak ada atribut lagi yang tersisa untuk pemisahan.

Pohon keputusan dapat digunakan untuk berbagai jenis tugas, seperti klasifikasi dan regresi. Dalam klasifikasi, pohon keputusan digunakan untuk memprediksi kelas atau label dari suatu objek berdasarkan atribut-atributnya. Dalam regresi, pohon keputusan digunakan untuk memprediksi nilai kontinu berdasarkan atribut-atributnya.

- 1) Keuntungan dari penggunaan pohon keputusan antara lain:
- 2) Mudah dipahami dan diinterpretasikan.
- 3) Tidak memerlukan persyaratan khusus pada data, seperti normalisasi atau penghilangan outlier.

- 4) Mampu menangani data numerik dan kategorikal.
- 5) Mampu mengidentifikasi atribut-atribut penting dalam data.

Namun, pohon keputusan juga memiliki beberapa kelemahan, seperti cenderung overfitting pada data yang kompleks atau noisy, sensitif terhadap perubahan kecil pada data, dan tidak efisien untuk data dengan jumlah atribut yang sangat besar.

Untuk memperbaiki kelemahan pohon keputusan, ada beberapa teknik yang digunakan, seperti penggunaan ensemble learning (misalnya, Random Forest dan Gradient Boosting) dan pruning (memangkas atau mengurangi struktur pohon yang kompleks).

### **3. Jaringan Syaraf Tiruan (Neural Networks)**

Jaringan Syaraf Tiruan (Neural Networks) adalah model komputasi yang terinspirasi oleh struktur dan fungsi jaringan saraf dalam sistem saraf manusia. Jaringan syaraf tiruan terdiri dari kumpulan unit pemroses sederhana yang disebut neuron buatan atau node, yang saling terhubung melalui koneksi yang memiliki bobot atau kekuatan tertentu.

Struktur jaringan syaraf tiruan biasanya terdiri dari tiga jenis layer: input layer, hidden layer, dan output layer. Input layer menerima masukan dari data yang ingin diproses. Hidden layer merupakan layer di antara input layer dan output layer yang bertanggung jawab untuk melakukan pemrosesan dan ekstraksi fitur dari masukan. Output layer menghasilkan keluaran atau prediksi berdasarkan hasil pemrosesan dalam hidden layer.

Setiap koneksi antara neuron memiliki bobot yang menunjukkan kekuatan koneksi tersebut. Selama proses pembelajaran, bobot-bobot ini akan diubah dan disesuaikan agar jaringan syaraf tiruan dapat menghasilkan keluaran yang sesuai dengan target yang diinginkan. Proses ini biasanya melibatkan pemilihan fungsi aktivasi yang digunakan untuk menghitung keluaran setiap neuron, serta penggunaan algoritma pembelajaran seperti backpropagation.

Backpropagation adalah salah satu algoritma pembelajaran yang umum digunakan dalam jaringan syaraf tiruan. Algoritma ini bekerja dengan menghitung gradien atau turunan dari fungsi kesalahan (loss function) terhadap bobot-bobot dalam jaringan, dan kemudian mengupdate bobot-bobot tersebut menggunakan metode penurunan gradien (gradient descent) untuk mengoptimalkan kinerja jaringan.

Jaringan syaraf tiruan dapat digunakan untuk berbagai jenis tugas, termasuk klasifikasi, regresi, pengenalan pola, dan pemrosesan bahasa alami. Dalam beberapa tahun terakhir, dengan kemajuan dalam komputasi dan peningkatan jumlah data yang tersedia, jaringan syaraf tiruan dalam bentuk yang lebih dalam dan kompleks seperti Jaringan Syaraf Tiruan Konvolusional (Convolutional Neural Networks) dan Jaringan

Syaraf Tiruan Rekurent (Recurrent Neural Networks) telah menjadi sangat populer dan berhasil dalam banyak aplikasi seperti pengenalan wajah, pengenalan suara, dan pengenalan teks.

Keuntungan dari penggunaan jaringan syaraf tiruan antara lain:

- 1) Mampu menangani masalah yang kompleks dan non-linear.
- 2) Mampu melakukan pemrosesan dan ekstraksi fitur secara otomatis.
- 3) Toleran terhadap data yang berisikan noise atau tidak lengkap.
- 4) Mampu menggeneralisasi dan melakukan prediksi pada data yang belum pernah dilihat sebelumnya.

Namun, jaringan syaraf tiruan juga memiliki beberapa kelemahan, seperti sulit untuk diinterpretasikan dan menjelaskan hasil prediksi yang diberikan, rentan terhadap overfitting pada data yang terlalu kompleks atau terlalu sedikit, serta memerlukan jumlah data yang signifikan untuk pelatihan yang efektif.

#### **4. Algoritma Genetika (Genetic Algorithms)**

Algoritma Genetika (Genetic Algorithms) adalah algoritma yang terinspirasi oleh proses evolusi dalam genetika dan digunakan untuk mencari solusi optimal atau pendekatan terbaik dalam masalah yang kompleks. Algoritma ini berdasarkan pada konsep seleksi alam dan pewarisan genetik yang terjadi dalam populasi organisme hidup.

Proses dalam algoritma genetika melibatkan representasi solusi sebagai individu dalam populasi, yang memiliki sifat-sifat yang diwariskan dari generasi sebelumnya. Setiap individu dalam populasi mewakili solusi potensial dalam ruang pencarian. Individu-individu ini dievaluasi berdasarkan suatu fungsi objektif yang mengukur kualitas solusi.

- 1) Berikut adalah langkah-langkah umum dalam algoritma genetika:
- 2) Inisialisasi: Populasi awal individu dibangkitkan secara acak. Setiap individu direpresentasikan sebagai kromosom yang terdiri dari gen-gen, yang mengkodekan atribut-atribut atau variabel-variabel solusi.
- 3) Evaluasi: Setiap individu dalam populasi dievaluasi menggunakan fungsi objektif. Fungsi objektif ini menentukan seberapa baik kualitas solusi yang diwakili oleh individu tersebut.
- 4) Seleksi: Individu-individu dengan kualitas solusi yang lebih baik memiliki peluang yang lebih besar untuk dipilih sebagai orang tua untuk reproduksi. Konsep seleksi alam digunakan di sini, di mana individu-individu yang memiliki

kualitas solusi yang lebih baik memiliki kemungkinan lebih tinggi untuk bertahan dan menghasilkan keturunan.

- 5) Rekombinasi (Crossover): Proses rekombinasi atau crossover dilakukan untuk menghasilkan keturunan baru dengan menggabungkan informasi genetik dari dua orang tua yang dipilih. Rekombinasi ini dilakukan dengan cara membagi kromosom kedua orang tua pada titik-titik pemotongan tertentu, dan menukar sebagian gen-gen antara kedua orang tua untuk menghasilkan keturunan baru.
- 6) Mutasi: Proses mutasi dilakukan pada keturunan dengan memperkenalkan perubahan kecil secara acak dalam gen-gen mereka. Tujuan dari mutasi adalah untuk menjaga keragaman genetik dalam populasi dan mencegah terjebak pada solusi lokal yang suboptimal.
- 7) Penggantian: Individu-individu yang dihasilkan dari rekombinasi dan mutasi digunakan untuk menggantikan individu-individu yang lebih buruk dalam populasi awal. Hal ini dilakukan untuk memperbaiki populasi dari generasi ke generasi.

Proses seleksi, rekombinasi, mutasi, dan penggantian ini diulang secara iteratif hingga mencapai kriteria penghentian tertentu, seperti jumlah generasi yang ditentukan atau mencapai solusi yang memadai.

Algoritma genetika memiliki keuntungan dalam menangani masalah optimasi kompleks yang memiliki ruang pencarian yang besar dan tidak terstruktur. Mereka juga mampu menemukan solusi yang baik bahkan dalam kondisi ketidakpastian atau keberagaman. Namun, mereka juga memiliki beberapa kelemahan, seperti waktu komputasi yang lama untuk masalah yang sangat kompleks, dan kemungkinan terjebak pada solusi lokal yang suboptimal.

## 5. Pembelajaran Mesin (Machine Learning)

Pembelajaran Mesin (Machine Learning) adalah cabang dari kecerdasan buatan (Artificial Intelligence) yang berkaitan dengan pengembangan teknik dan algoritma yang memungkinkan komputer untuk belajar dari data dan pengalaman, tanpa perlu secara eksplisit diprogram secara terperinci. Tujuan utama dari pembelajaran mesin adalah mengembangkan model atau sistem yang dapat melakukan prediksi, klasifikasi, atau pengambilan keputusan yang baik berdasarkan pola atau informasi yang ada dalam data.

Pembelajaran mesin melibatkan beberapa konsep dan teknik, di antaranya:

- 1) Data: Pembelajaran mesin menggunakan data sebagai input untuk melatih model atau sistem. Data ini bisa berupa atribut-atribut (fitur) yang menggambarkan objek atau situasi tertentu, serta label atau target yang ingin diprediksi atau diklasifikasikan. Data ini biasanya dibagi menjadi dua set, yaitu set pelatihan

(training set) yang digunakan untuk melatih model, dan set pengujian (testing set) yang digunakan untuk menguji dan mengevaluasi kinerja model.

- 2) Model: Model dalam pembelajaran mesin adalah representasi matematis atau komputasional dari suatu masalah atau sistem. Model ini berfungsi untuk mengekstrak pola atau informasi dari data dan menghasilkan prediksi atau keputusan. Contoh model yang umum digunakan dalam pembelajaran mesin termasuk pohon keputusan (decision tree), jaringan syaraf tiruan (neural networks), dan mesin vektor pendukung (support vector machines).
- 3) Pelatihan (Training): Proses pelatihan melibatkan penggunaan algoritma pembelajaran untuk menyesuaikan parameter atau bobot model berdasarkan data pelatihan. Tujuan pelatihan adalah mengoptimalkan kinerja model sehingga dapat memberikan prediksi yang akurat atau keputusan yang baik.
- 4) Evaluasi dan Validasi: Setelah melatih model, penting untuk menguji dan mengevaluasi kinerja model tersebut. Evaluasi dilakukan dengan menggunakan set pengujian yang terpisah dari set pelatihan. Metrik evaluasi seperti akurasi, presisi, recall, atau mean squared error digunakan untuk mengukur sejauh mana model dapat memprediksi atau mengklasifikasikan dengan benar.
- 5) Umumkan atau Penerapan: Setelah model dilatih dan dievaluasi dengan baik, model tersebut dapat digunakan untuk memprediksi atau mengklasifikasikan data baru. Model juga dapat digunakan dalam lingkungan produksi untuk memecahkan masalah nyata atau memberikan rekomendasi.

Pembelajaran mesin memiliki berbagai jenis pendekatan, termasuk pembelajaran terawasi (supervised learning) di mana model dilatih menggunakan data berlabel, pembelajaran tak terawasi (unsupervised learning) di mana model mencari pola atau struktur dalam data yang tidak berlabel, dan pembelajaran penguatan (reinforcement learning) di mana model belajar melalui interaksi dengan lingkungan dan mendapatkan umpan balik dalam bentuk hadiah atau hukuman.

Penerapan pembelajaran mesin telah berhasil dalam berbagai bidang, termasuk pengenalan wajah, pengenalan suara, pemrosesan bahasa alami, pemrosesan citra, robotika, pengenalan pola, dan analisis data.

## 6. Pemrosesan Bahasa Alami (Natural Language Processing/NLP)

Pemrosesan Bahasa Alami (Natural Language Processing/NLP) adalah cabang dari kecerdasan buatan (Artificial Intelligence) yang berfokus pada interaksi antara manusia dan komputer melalui pemahaman dan penghasilan bahasa manusia yang alami. Tujuan utama dari NLP adalah memungkinkan komputer untuk memahami, memproses, dan menghasilkan bahasa manusia dalam bentuk yang dapat dimengerti.

NLP melibatkan berbagai teknik dan algoritma untuk memahami dan memproses bahasa manusia, termasuk:

- 1) Pemahaman Bahasa Alami: Proses pemahaman bahasa alami melibatkan analisis struktur, semantik, dan makna dari teks atau kalimat dalam bahasa manusia. Ini meliputi pemecahan kalimat menjadi kata-kata, analisis sintaksis (pemahaman struktur tata bahasa), serta pemahaman semantik (pengenalan makna kata-kata dan hubungan antara mereka).
- 2) Pengolahan Teks: Pengolahan teks melibatkan berbagai teknik untuk mengolah dan mengelompokkan teks dalam bahasa manusia. Ini meliputi tokenisasi (pemisahan teks menjadi kata-kata atau token), stemming (penghapusan afiks dari kata-kata), lemmatisasi (pengembalian kata-kata ke bentuk dasar), pemodelan n-gram (mengidentifikasi urutan kata), dan lain-lain.
- 3) Klasifikasi dan Kategorisasi Teks: Teknik ini melibatkan pengelompokan atau klasifikasi teks ke dalam kategori yang sesuai berdasarkan kontennya. Algoritma pembelajaran mesin seperti Naive Bayes, Support Vector Machines, atau Jaringan Syaraf Tiruan sering digunakan untuk membangun model yang dapat mengklasifikasikan teks.
- 4) Penerjemahan Mesin: Penerjemahan mesin adalah proses mengubah teks dalam satu bahasa menjadi teks dalam bahasa lain secara otomatis. Metode penerjemahan mesin melibatkan penggunaan model statistik, metode berbasis aturan, atau pendekatan berbasis jaringan syaraf tiruan.
- 5) Pemrosesan Bicara: Pemrosesan bicara melibatkan konversi suara manusia menjadi teks atau sebaliknya. Teknik pengenal suara digunakan untuk mengenali dan mentranskripsi ucapan menjadi teks, sedangkan sintesis suara digunakan untuk menghasilkan suara manusia berdasarkan teks yang diberikan.
- 6) Ekstraksi Informasi: Ekstraksi informasi melibatkan identifikasi dan ekstraksi informasi yang relevan dari teks, seperti entitas bernama (nama orang, tempat, organisasi), hubungan antar entitas, dan fakta-fakta penting lainnya.

Pemrosesan Bahasa Alami memiliki banyak aplikasi, termasuk pemrosesan teks massal, asisten virtual, analisis sentimen, chatbot, sistem tanya jawab, pengenalan ucapan, dan banyak lagi. NLP terus berkembang dengan peningkatan dalam teknik-teknik seperti pembelajaran mendalam (deep learning), pemrosesan bahasa alami berbasis grafik, dan penggunaan model bahasa yang besar seperti BERT (Bidirectional Encoder Representations from Transformers).

## **BAB II**

### **RUANG MASALAH & SISTEM PRODUKSI**

#### **A. Ruang Masalah**

Dalam kecerdasan buatan (Artificial Intelligence), ruang masalah merujuk pada kumpulan semua kemungkinan solusi atau keadaan yang relevan untuk suatu masalah yang ingin dipecahkan oleh sistem AI. Ruang masalah mencakup semua variabel, konstrain, dan kemungkinan keadaan yang relevan yang harus dipertimbangkan saat mencari solusi.

Ruang masalah dapat bervariasi tergantung pada jenis masalah yang ingin diselesaikan. Berikut adalah beberapa contoh ruang masalah dalam AI:

1. Ruang Masalah Pencarian: Ruang masalah pencarian berkaitan dengan mencari solusi optimal atau pendekatan terbaik melalui ruang kemungkinan keadaan. Contohnya adalah masalah jalan terpendek dalam peta, di mana ruang masalah mencakup semua rute yang mungkin antara dua titik.
2. Ruang Masalah Klasifikasi: Ruang masalah klasifikasi melibatkan pemisahan objek atau data ke dalam kelas-kelas yang berbeda berdasarkan atribut-atribut tertentu. Ruang masalah mencakup semua kemungkinan kombinasi atribut dan label kelas yang relevan.
3. Ruang Masalah Regresi: Ruang masalah regresi terkait dengan memodelkan hubungan antara variabel input dan output kontinu. Ruang masalah mencakup semua kemungkinan kombinasi nilai-nilai input dan output yang relevan.
4. Ruang Masalah Pengenalan Pola: Ruang masalah pengenalan pola melibatkan pengenalan atau klasifikasi pola yang kompleks dalam data. Ruang masalah mencakup semua kemungkinan kombinasi pola dan atribut yang relevan dalam data.
5. Ruang Masalah Perencanaan: Ruang masalah perencanaan berkaitan dengan mencari urutan tindakan yang optimal untuk mencapai tujuan tertentu. Ruang masalah mencakup semua kemungkinan urutan tindakan yang mungkin dan keadaan yang relevan selama perencanaan.
6. Ruang Masalah Pemrosesan Bahasa Alami: Ruang masalah dalam pemrosesan bahasa alami mencakup semua kemungkinan konstruksi kalimat, pola kata, dan makna yang relevan dalam bahasa manusia.

Selain itu, ruang masalah juga dapat bergantung pada faktor-faktor seperti dimensi data, kompleksitas, dan jenis informasi yang diperlukan. Memahami dan memodelkan

ruang masalah yang tepat adalah langkah penting dalam merancang dan mengimplementasikan solusi AI yang efektif.

## B. Sistem Produksi

Sistem produksi adalah salah satu paradigma yang digunakan dalam bidang kecerdasan buatan (Artificial Intelligence) untuk memodelkan pengetahuan dan perilaku berbasis aturan. Sistem produksi terdiri dari aturan-aturan produksi yang terdiri dari kondisi dan aksi, yang digunakan untuk mengambil keputusan atau menghasilkan keluaran berdasarkan kondisi yang ada.

Sebuah sistem produksi terdiri dari beberapa komponen utama:

1. Basis Pengetahuan (Knowledge Base): Basis pengetahuan berisi aturan-aturan produksi yang mewakili pengetahuan dan kebijakan domain yang ingin diterapkan. Aturan-aturan ini terdiri dari kondisi (juga dikenal sebagai klausul IF) yang harus terpenuhi dan tindakan (juga dikenal sebagai klausul THEN) yang harus diambil jika kondisi tersebut terpenuhi.
2. Mesin Inferensi (Inference Engine): Mesin inferensi bertanggung jawab untuk menerapkan aturan-aturan produksi dalam basis pengetahuan. Ini adalah komponen yang mengevaluasi kondisi-kondisi dalam aturan-aturan produksi dan mengambil tindakan yang sesuai berdasarkan fakta-fakta yang ada.
3. Basis Fakta (Fact Base): Basis fakta berisi informasi aktual yang ada pada suatu saat dalam sistem. Ini mencerminkan keadaan atau situasi yang sedang diamati dan dapat berubah seiring waktu. Basis fakta ini digunakan oleh mesin inferensi untuk mengevaluasi kondisi-kondisi dalam aturan-aturan produksi.
4. Antarmuka Pengguna: Antarmuka pengguna memungkinkan interaksi antara pengguna dan sistem produksi. Pengguna dapat memasukkan input, mengubah basis pengetahuan atau basis fakta, dan melihat output atau hasil dari sistem produksi.

Proses kerja sistem produksi adalah sebagai berikut:

1. Evaluasi Kondisi: Mesin inferensi mengevaluasi kondisi-kondisi dalam aturan-aturan produksi dengan memeriksa kecocokan dengan basis fakta yang ada. Aturan-aturan produksi yang memenuhi kondisi akan diaktifkan.
2. Pemilihan Aksi: Setelah aturan-aturan produksi yang teraktifkan ditentukan, mesin inferensi memilih tindakan atau aksi yang sesuai untuk dijalankan berdasarkan kondisi-kondisi yang terpenuhi.
3. Eksekusi Aksi: Tindakan yang dipilih kemudian dieksekusi, menghasilkan perubahan pada basis fakta atau menghasilkan output yang diinginkan.

Sistem produksi dapat digunakan dalam berbagai aplikasi, seperti sistem pakar, sistem diagnosis, sistem rekomendasi, atau sistem kontrol. Mereka memungkinkan representasi pengetahuan yang berbasis aturan dan kemampuan untuk membuat keputusan atau menghasilkan keluaran berdasarkan kondisi yang ada.

## **BAB III**

### **METODE-METODE PENCARIAN DALAM KECERDASAN BUATAN**

#### **A. Pencarian Buta**

Pencarian buta (blind search) dalam kecerdasan buatan (Artificial Intelligence) merujuk pada strategi pencarian yang digunakan ketika informasi tentang lingkungan atau tujuan tidak tersedia. Dalam pencarian buta, algoritma hanya memiliki akses terbatas pada informasi tentang keadaan saat ini dan tidak memiliki pengetahuan sebelumnya tentang lingkungan atau cara mencapai tujuan.

Ada beberapa strategi pencarian buta yang umum digunakan, termasuk:

##### **1. Pencarian Breadth-First (Pencarian Luas)**

Pencarian BF (Breadth-First Search) atau Pencarian Luas adalah salah satu strategi pencarian buta yang digunakan dalam kecerdasan buatan untuk menemukan solusi dalam ruang pencarian. Pencarian BF mengeksplorasi simpul-simpul secara bertingkat, mulai dari simpul awal, kemudian ke simpul-simpul tetangganya, dan seterusnya, hingga menemukan solusi yang diinginkan.

Berikut adalah langkah-langkah umum dalam pencarian BF:

- 1) Inisialisasi: Simpul awal atau keadaan awal diberi label dan dimasukkan ke dalam antrian.
- 2) Ekspansi: Simpul pertama dalam antrian diekstrak (dequeue) dan diperiksa. Jika simpul tersebut adalah solusi yang diinginkan, pencarian selesai. Jika tidak, simpul tersebut diperluas dengan menghasilkan simpul-simpul tetangga dan menambahkannya ke dalam antrian.
- 3) Antrian: Simpul-simpul tetangga yang baru dihasilkan dimasukkan ke dalam antrian dengan mempertahankan urutan pencarian luas. Ini berarti simpul-simpul yang pertama kali dimasukkan akan dieksplorasi sebelum simpul-simpul yang dimasukkan kemudian.
- 4) Penandaan: Setelah simpul diekstrak dari antrian, biasanya simpul tersebut ditandai sebagai "dieksplorasi" atau "dikunjungi" untuk menghindari pengulangan dalam proses pencarian.
- 5) Pengulangan: Langkah 2 hingga 4 diulangi sampai solusi ditemukan atau antrian kosong, yang menunjukkan bahwa tidak ada solusi yang dapat dicapai.

Keuntungan dari pencarian BF adalah bahwa itu akan menemukan solusi terpendek dalam ruang pencarian, karena ia menjelajahi simpul-simpul secara bertingkat. Namun, kelemahan utama pencarian BF adalah bahwa itu dapat menghabiskan banyak memori, terutama jika ruang pencarian sangat besar.

Pencarian BF sering digunakan dalam masalah pencarian rute terpendek, graf, atau struktur data dengan kompleksitas rendah.

## 2. Pencarian Depth-First (Pencarian Dalam)

Pencarian DF (Depth-First Search) atau Pencarian Dalam adalah salah satu strategi pencarian buta yang digunakan dalam kecerdasan buatan untuk menemukan solusi dalam ruang pencarian. Pencarian DF secara rekursif menjelajahi jalur tertentu hingga mencapai simpul tujuan atau simpul tanpa anak, dan kemudian mundur ke simpul sebelumnya untuk mencoba jalur alternatif.

Berikut adalah langkah-langkah umum dalam pencarian DF:

- 1) Inisialisasi: Simpul awal atau keadaan awal diberi label dan dimasukkan ke dalam tumpukan.
- 2) Ekspansi: Simpul pertama dalam tumpukan diekstrak (pop) dan diperiksa. Jika simpul tersebut adalah solusi yang diinginkan, pencarian selesai. Jika tidak, simpul tersebut diperluas dengan menghasilkan simpul-simpul tetangga dan menambahkannya ke dalam tumpukan.
- 3) Tumpukan: Simpul-simpul tetangga yang baru dihasilkan dimasukkan ke dalam tumpukan, sehingga menjadi prioritas dalam pencarian. Ini berarti simpul-simpul yang terakhir dimasukkan akan dieksplorasi terlebih dahulu sebelum simpul-simpul yang sebelumnya dimasukkan.
- 4) Penandaan: Setelah simpul diekstrak dari tumpukan, biasanya simpul tersebut ditandai sebagai "dieksplorasi" atau "dikunjungi" untuk menghindari pengulangan dalam proses pencarian.
- 5) Pengulangan: Langkah 2 hingga 4 diulangi secara rekursif sampai solusi ditemukan atau tumpukan kosong, yang menunjukkan bahwa tidak ada solusi yang dapat dicapai.

Keuntungan dari pencarian DF adalah bahwa ia dapat menemukan solusi dalam ruang pencarian yang dalam dengan menggunakan memori yang relatif sedikit. Namun, kelemahan utama pencarian DF adalah bahwa ia mungkin terjebak dalam lingkaran tak terbatas atau jalur yang sangat panjang tanpa solusi.

Pencarian DF sering digunakan dalam masalah seperti penelusuran graf, pemecahan teka-teki, dan eksplorasi ruang keadaan.

### **3. Pencarian Uniform Cost (Pencarian Biaya Seragam)**

Strategi ini memilih simpul dengan biaya terendah untuk dieksplorasi terlebih dahulu. Ini berguna saat mencari solusi dengan biaya minimal, seperti mencari rute terpendek dalam jaringan jalan.

Pencarian Uniform Cost (Pencarian Biaya Seragam) adalah strategi pencarian yang digunakan dalam kecerdasan buatan untuk menemukan solusi dengan biaya minimal. Pencarian ini mempertimbangkan biaya yang terkait dengan setiap aksi atau perpindahan antar-simpul dalam ruang pencarian.

Pencarian Uniform Cost memastikan bahwa solusi dengan biaya minimal ditemukan, karena simpul dengan biaya yang lebih rendah dieksplorasi lebih awal. Ini sangat berguna dalam masalah seperti mencari rute terpendek dalam jaringan jalan atau mencari solusi dengan biaya terendah.

### **4. Pencarian Iterative Deepening (Pencarian Dalam yang Dalam dan Bertahap)**

Strategi ini adalah kombinasi dari pencarian depth-first dan breadth-first. Algoritma mulai dengan pencarian depth-first dengan batasan kedalaman terbatas, dan kemudian secara bertahap meningkatkan batasan kedalaman hingga menemukan solusi.

Pencarian buta memiliki kelemahan utama yaitu kurangnya informasi heuristik atau pengetahuan sebelumnya, yang dapat mengarah pada pencarian yang lambat atau terjebak dalam ruang pencarian yang besar. Namun, mereka sering digunakan sebagai langkah awal dalam pencarian solusi sebelum menggunakan strategi pencarian yang lebih cerdas atau informasi pengetahuan tambahan yang tersedia.

## **B. Pencarian Heuristik**

Pencarian heuristik adalah strategi pencarian yang menggunakan informasi heuristik atau pengetahuan sebelumnya tentang lingkungan atau masalah yang sedang diselesaikan. Heuristik adalah aturan praktis atau aturan berbasis pengalaman yang digunakan untuk membantu dalam pengambilan keputusan atau pencarian solusi yang efisien.

Dalam pencarian heuristik, algoritma menggunakan fungsi heuristik untuk mengestimasi biaya atau nilai dari setiap langkah atau simpul dalam ruang pencarian. Fungsi heuristik ini memberikan indikasi tentang seberapa "baik" atau "dekat" sebuah simpul dengan solusi yang diinginkan. Algoritma kemudian menggunakan informasi ini untuk memandu pencarian, fokus pada simpul-simpul yang lebih menjanjikan atau berpotensi lebih dekat dengan solusi.

Pencarian heuristik dapat membantu dalam mengurangi waktu pencarian dan mengarahkan algoritma ke solusi yang lebih efisien. Namun, perlu diperhatikan bahwa keefektifan pencarian heuristik tergantung pada kualitas dan kecocokan heuristik yang

digunakan dengan masalah yang sedang diselesaikan. Heuristik yang buruk atau tidak relevan dapat menghasilkan solusi yang tidak optimal atau bahkan tidak dapat ditemukan.

## 1. Pencarian Terarah (Informed Search)

Pencarian terarah (informed search) adalah strategi pencarian yang menggunakan informasi heuristik atau pengetahuan sebelumnya untuk memandu eksplorasi dalam ruang pencarian. Strategi ini didasarkan pada perkiraan atau estimasi tentang seberapa dekat setiap simpul dengan solusi yang diinginkan.

Salah satu algoritma pencarian terarah yang populer adalah A\* (A-Star). Algoritma A\* menggabungkan pencarian biaya seragam (uniform cost) dengan estimasi heuristik. Pada setiap langkah, algoritma A\* mempertimbangkan biaya sejauh ini dari simpul awal ke simpul saat ini (dikenal sebagai  $g(n)$ ), serta estimasi biaya dari simpul saat ini ke simpul tujuan (dikenal sebagai  $h(n)$ ). Total nilai  $f(n) = g(n) + h(n)$  digunakan untuk menentukan urutan eksplorasi simpul berikutnya. Algoritma A\* berusaha untuk memilih simpul dengan nilai  $f(n)$  yang lebih rendah, mengarahkan pencarian ke simpul yang diharapkan akan mendekati solusi lebih cepat.

Pencarian terarah memberikan keuntungan dalam efisiensi pencarian karena memandu algoritma untuk fokus pada simpul-simpul yang lebih menjanjikan atau berpotensi lebih dekat dengan solusi. Dengan menggunakan informasi heuristik, pencarian terarah dapat menghindari eksplorasi yang tidak perlu dalam ruang pencarian.

## 2. Pencarian Greedy(Greedy Search)

Pencarian greedy adalah strategi pencarian yang memilih langkah terbaik pada setiap langkah berdasarkan kriteria lokal, tanpa mempertimbangkan konsekuensi jangka panjang atau tujuan akhir secara menyeluruh. Algoritma pencarian greedy memilih simpul atau tindakan yang paling menjanjikan pada setiap langkah berdasarkan penilaian heuristik atau sinyal lokal, tanpa melakukan pencarian lanjutan atau mempertimbangkan konsekuensi dari pilihan tersebut.

Dalam pencarian greedy, algoritma berfokus pada tujuan sementara atau pilihan terbaik saat ini, tanpa memperhatikan langkah-langkah selanjutnya atau solusi keseluruhan. Pencarian greedy cenderung memaksimalkan keuntungan sejauh mungkin pada setiap langkah, tanpa mempertimbangkan risiko atau kemungkinan menghasilkan solusi yang tidak optimal dalam jangka panjang.

Keuntungan dari pencarian greedy adalah kecepatan eksekusi yang relatif tinggi, karena hanya mempertimbangkan penilaian lokal pada setiap langkah. Pencarian greedy juga dapat memberikan solusi yang memadai dalam beberapa kasus sederhana atau ketika waktu eksekusi menjadi faktor yang kritis.

Namun, kelemahan utama pencarian greedy adalah ketidakmampuannya untuk menjamin solusi yang optimal atau global. Karena fokus pada pilihan terbaik saat ini, algoritma ini mungkin mengabaikan alternatif yang lebih baik dalam jangka panjang. Pencarian greedy juga dapat terjebak dalam optimum lokal, di mana solusi yang ditemukan tidak mencapai solusi terbaik secara keseluruhan.

Pencarian greedy sering digunakan dalam situasi di mana solusi yang memadai dalam waktu singkat lebih diutamakan daripada solusi optimal. Namun, perlu diingat bahwa dalam banyak kasus, pencarian greedy harus digabungkan dengan strategi pencarian lain atau dianalisis secara kritis untuk memastikan kualitas solusi yang memadai.

### 3. Pencarian A\*(A-Star Search)

Pencarian A\* (A-Star Search) adalah algoritma pencarian terarah yang menggabungkan pencarian biaya seragam (uniform cost search) dengan estimasi heuristik. Algoritma A\* digunakan untuk menemukan jalur terpendek atau solusi optimal dalam ruang pencarian.

Pencarian A\* mempertimbangkan dua faktor penting untuk mengevaluasi simpul-simpul dalam ruang pencarian:

- 1) Biaya Sejauh Ini ( $g(n)$ ): Ini adalah biaya aktual yang diperlukan untuk mencapai simpul saat ini dari simpul awal. Pencarian biaya seragam (uniform cost search) digunakan untuk menghitung biaya ini dengan menjaga jejak biaya terkini saat melakukan eksplorasi.
- 2) Estimasi Biaya Ke Tujuan ( $h(n)$ ): Ini adalah estimasi heuristik tentang biaya yang diperlukan untuk mencapai simpul tujuan dari simpul saat ini. Estimasi ini dapat berdasarkan informasi domain, aturan heuristik, atau pengetahuan sebelumnya tentang masalah yang sedang diselesaikan. Estimasi ini tidak boleh melebihi biaya sebenarnya yang diperlukan untuk mencapai simpul tujuan.

Total nilai  $f(n) = g(n) + h(n)$  digunakan sebagai fungsi penilaian untuk memilih simpul berikutnya yang akan dieksplorasi. Algoritma A\* berusaha meminimalkan nilai  $f(n)$ , sehingga menjelajahi simpul-simpul yang diharapkan lebih dekat dengan solusi.

Pencarian A\* menggabungkan keuntungan dari pencarian biaya seragam (uniform cost search) dan heuristik, sehingga dapat menemukan solusi optimal dalam ruang pencarian dengan efisien. Namun, kualitas heuristik yang digunakan sangat penting untuk memastikan efektivitas pencarian A\*. Heuristik yang buruk

## **BAB IV**

## **REASONING**

### **A. Propotional Logic**

Logika proposisional, juga dikenal sebagai logika proposisi atau logika kalimat, adalah cabang logika matematika yang mempelajari hubungan logis antara proposisi atau pernyataan. Logika proposisional berfokus pada pengolahan simbol-simbol logis, di mana proposisi dinyatakan dengan menggunakan simbol-simbol logis dan dihubungkan melalui operator logis.

#### **1. Negasi**

Negasi (NOT) digunakan untuk membalik nilai kebenaran sebuah proposisi. Dalam logika proposisional, negasi dinyatakan dengan menggunakan simbol ' $\sim$ ' atau ' $\neg$ '. Jika  $p$  adalah sebuah proposisi, maka negasi dari  $p$  ditulis sebagai  $\sim p$  atau  $\neg p$ . Jika  $p$  bernilai True, maka  $\sim p$  atau  $\neg p$  akan bernilai False, dan sebaliknya.

Contoh: Jika  $p$  adalah proposisi "Saya sedang belajar", maka negasi dari  $p$  adalah  $\sim p$  atau  $\neg p$  yang berarti "Saya tidak sedang belajar".

#### **2. Konjungsi**

Konjungsi (AND) digunakan untuk menghubungkan dua proposisi dan menghasilkan proposisi baru yang bernilai True hanya jika kedua proposisi yang terhubung juga bernilai True. Dalam logika proposisional, konjungsi dinyatakan dengan menggunakan simbol ' $\wedge$ ' atau ' $\Lambda$ '.

Contoh: Jika  $p$  adalah proposisi "Saya sedang belajar" dan  $q$  adalah proposisi "Saya sedang membaca", maka  $p \wedge q$  adalah konjungsi dari  $p$  dan  $q$  yang berarti "Saya sedang belajar dan saya sedang membaca".

#### **3. Disjungsi**

Disjungsi (OR): Disjungsi digunakan untuk menghubungkan dua proposisi dan menghasilkan proposisi baru yang bernilai True jika salah satu atau kedua proposisi yang terhubung bernilai True. Dalam logika proposisional, disjungsi dinyatakan dengan menggunakan simbol ' $\vee$ ' atau ' $\vee$ '.

Contoh: Jika  $p$  adalah proposisi "Saya sedang belajar" dan  $q$  adalah proposisi "Saya sedang membaca", maka  $p \vee q$  adalah disjungsi dari  $p$  dan  $q$  yang berarti "Saya sedang belajar atau saya sedang membaca".

#### **4. Implikasi**

Implikasi (Imply): Implikasi menghubungkan dua proposisi dan menghasilkan proposisi baru yang menyatakan bahwa jika proposisi pertama bernilai True, maka proposisi kedua juga bernilai True. Namun, jika proposisi pertama bernilai False, maka tidak ada konsekuensi yang dapat ditarik tentang nilai proposisi kedua. Dalam logika proposisional, implikasi dinyatakan dengan menggunakan simbol ' $\rightarrow$ '.

Contoh: Jika p adalah proposisi "Jika hujan, maka jalanan basah", dan q adalah proposisi "Jalanan basah", maka  $p \rightarrow q$  berarti "Jika hujan, maka jalanan basah".

#### **5. Biconditional**

Biconditional (If and only if): Biconditional menghubungkan dua proposisi dan menghasilkan proposisi baru yang bernilai True jika kedua proposisi yang terhubung memiliki nilai kebenaran yang sama (baik keduanya True atau keduanya False). Dalam logika proposisional, biconditional dinyatakan dengan menggunakan simbol ' $\leftrightarrow$ '.

Contoh: Jika p adalah proposisi "Saya sedang belajar" dan q adalah proposisi "Saya sedang membaca", maka  $p \leftrightarrow q$  berarti "Saya sedang belajar jika dan hanya jika saya sedang membaca".

### **B. First Order Logic (Predicate Calculus)**

First Order Logic (FOL), juga dikenal sebagai Logika Predikat atau Kalkulus Predikat, adalah sistem logika yang digunakan untuk merepresentasikan pengetahuan dalam bentuk pernyataan yang melibatkan predikat, variabel, dan kuantifikasi kuantor. FOL merupakan perluasan dari logika proposisional yang memungkinkan penjelasan yang lebih rinci dan formal tentang relasi antarobjek dalam domain.

#### **1. Variable**

Variabel digunakan untuk mewakili objek atau entitas dalam domain yang dianalisis. Variabel sering dilambangkan dengan huruf kecil seperti x, y, z.

#### **2. Konstanta**

Konstanta adalah simbol yang merepresentasikan objek yang tetap dalam domain. Contoh konstanta adalah "John" atau "Mary".

#### **3. Fungsi**

Fungsi adalah simbol atau ungkapan yang menggambarkan hubungan antara objek dalam domain dan menghasilkan nilai baru. Fungsi sering dilambangkan dengan huruf kecil atau kata kerja seperti "FatherOf(x)" yang berarti "x adalah ayah dari seseorang".

#### 4. Predikat

Predikat adalah simbol atau ungkapan yang menggambarkan sifat atau hubungan antara objek dalam domain. Predikat sering dilambangkan dengan huruf besar atau kata kerja seperti "Man(x)" yang berarti "x adalah seorang pria".

Contoh pernyataan dalam FOL:

- $\forall x \text{ Man}(x) \rightarrow \text{Mortal}(x)$  (Untuk setiap x, jika x adalah seorang pria, maka x adalah makhluk fana)
- $\exists x \text{ FatherOf}(x, \text{John})$  (Ada seseorang yang adalah ayah dari John)

FOL digunakan secara luas dalam kecerdasan buatan, pemrosesan bahasa alami, pemrograman logika, dan pemahaman pengetahuan. Ini memberikan kerangka kerja formal untuk merepresentasikan pengetahuan, melakukan penalaran, dan menghasilkan kesimpulan logis. Dengan FOL, kita dapat mengartikan dan menganalisis pernyataan dengan lebih rinci dan akurat dalam domain yang kompleks.

### C. Fuzzy Logic

Logika Fuzzy adalah cabang logika matematika yang digunakan untuk memodelkan dan mengolah ketidakpastian dan keambiguan dalam penalaran. Berbeda dengan logika klasik yang hanya mengakui nilai kebenaran yang mutlak yaitu benar (True) atau salah (False), logika fuzzy mengizinkan nilai kebenaran yang dapat berada di antara dua nilai tersebut, seperti "sedikit benar" atau "cukup benar".

Dalam logika fuzzy, variabel dan pernyataan dinyatakan dalam bentuk fungsi keanggotaan yang memetakan nilai-nilai ke dalam interval antara 0 dan 1. Fungsi keanggotaan ini menunjukkan sejauh mana suatu elemen atau variabel memenuhi suatu sifat atau kondisi tertentu. Nilai keanggotaan ini mencerminkan tingkat kebenaran dari pernyataan tersebut.

Logika fuzzy juga memperkenalkan operasi-operasi logika fuzzy seperti konjungsi fuzzy, disjungsi fuzzy, negasi fuzzy, dan implikasi fuzzy. Ketika operasi-operasi logika fuzzy diterapkan pada variabel fuzzy, nilai kebenaran hasil operasi tersebut dapat menggambarkan tingkat ketidakpastian atau keambiguan dalam penalaran.

Keuntungan utama logika fuzzy adalah kemampuannya untuk mengatasi ketidakpastian dan keambiguan dalam pemodelan dan penalaran. Ini memungkinkan logika fuzzy digunakan dalam berbagai aplikasi yang melibatkan penalaran berbasis aturan, pengendalian otomatis, pengambilan keputusan, dan sistem pendukung keputusan.

Contoh penggunaan logika fuzzy adalah dalam sistem pengendalian kecepatan kendaraan berdasarkan kondisi lalu lintas. Dalam sistem logika fuzzy ini, variabel seperti "kecepatan" dan "jarak" dinyatakan dalam bentuk himpunan fuzzy, dan aturan-aturan fuzzy didefinisikan untuk mengendalikan kecepatan kendaraan berdasarkan kombinasi kecepatan dan jarak. Hal ini memungkinkan sistem pengendalian untuk mengambil

keputusan yang lebih adaptif dan responsif terhadap kondisi lalu lintas yang berubah-ubah.

Logika fuzzy memberikan kerangka kerja yang kuat untuk mengolah ketidakpastian dan keambiguan dalam penalaran dan pengambilan keputusan. Dalam banyak kasus di mana lingkungan atau data yang dihadapi tidak dapat diukur secara akurat atau tegas, logika fuzzy dapat memberikan solusi yang lebih fleksibel dan efektif.

### **1. Fuzziness & Probability**

Fuzziness dan probabilitas adalah dua konsep yang berbeda dalam konteks ketidakpastian. Fuzziness mengacu pada tingkat ketidakpastian atau keambiguan dalam suatu konsep atau pernyataan, sedangkan probabilitas mengukur tingkat keyakinan atau peluang suatu peristiwa terjadi. Fuzziness berkaitan dengan logika fuzzy, sementara probabilitas berkaitan dengan teori probabilitas. Dalam logika fuzzy, nilai kebenaran adalah kontinum antara benar dan salah, sedangkan dalam probabilitas, nilai peluang adalah antara 0 dan 1. Meskipun ada beberapa persamaan konseptual antara fuzziness dan probabilitas, keduanya digunakan dalam konteks yang berbeda dan memiliki pendekatan yang berbeda dalam menghadapi ketidakpastian.

### **2. Fuzzy Set**

Fuzzy set (himpunan fuzzy) adalah konsep dasar dalam logika fuzzy. Sebuah himpunan fuzzy adalah himpunan yang elemen-elemennya memiliki derajat keanggotaan yang berada dalam interval antara 0 dan 1. Derajat keanggotaan ini mencerminkan tingkat keterkaitan elemen dengan himpunan tersebut. Dalam himpunan klasik, sebuah elemen hanya bisa anggota atau bukan anggota dari himpunan, sedangkan dalam himpunan fuzzy, elemen dapat memiliki tingkat keanggotaan yang berbeda dalam himpunan tersebut. Misalnya, dalam himpunan fuzzy "tinggi", tinggi suatu orang dapat memiliki derajat keanggotaan yang berbeda-beda, seperti 0.8 atau 0.5, yang menunjukkan sejauh mana tinggi orang tersebut memenuhi kriteria tinggi.

### **3. Fuzzy Logic**

Fuzzy logic (logika fuzzy) adalah pendekatan logika yang menggunakan konsep nilai kebenaran yang berada dalam rentang kontinum antara benar dan salah. Dalam logika fuzzy, proposisi atau pernyataan dapat memiliki nilai kebenaran yang bukan hanya benar atau salah, tetapi juga berbagai tingkatan di antaranya, seperti "sedikit benar" atau "cukup benar". Logika fuzzy memperluas logika klasik dengan memperkenalkan fungsi keanggotaan dan operasi-operasi logika fuzzy seperti konjungsi fuzzy, disjungsi fuzzy, negasi fuzzy, dan implikasi fuzzy. Fuzzy logic digunakan dalam berbagai aplikasi yang melibatkan penalaran berbasis aturan dan pengambilan keputusan di bawah

#### **4. Fuzzy System**

Fuzzy system (sistem fuzzy) adalah sistem komputasi yang menggunakan prinsip logika fuzzy untuk melakukan penalaran dan pengambilan keputusan. Sistem ini terdiri dari tiga komponen utama: himpunan fuzzy, basis aturan fuzzy, dan mekanisme inferensi fuzzy. Himpunan fuzzy digunakan untuk memodelkan variabel input dan output dalam bentuk himpunan fuzzy dengan fungsi keanggotaan yang sesuai. Basis aturan fuzzy berisi himpunan aturan yang menghubungkan kondisi (input) dengan aksi (output). Mekanisme inferensi fuzzy menggunakan aturan-aturan fuzzy untuk menghasilkan output fuzzy berdasarkan input fuzzy. Fuzzy system digunakan

## **BAB V**

### **PLANNING( TEKNIK DEKOMPOSISI MASALAH)**

Teknik dekomposisi masalah dalam perencanaan (planning) adalah pendekatan yang digunakan untuk memecah masalah perencanaan yang kompleks menjadi submasalah yang lebih kecil dan lebih mudah dipecahkan. Dalam teknik dekomposisi, masalah perencanaan yang kompleks dipecah menjadi serangkaian tugas atau subtujuan yang lebih sederhana, yang kemudian dapat dipecahkan secara terpisah dan digabungkan untuk mencapai tujuan akhir.

Berikut adalah langkah-langkah umum dalam teknik dekomposisi masalah perencanaan:

- 1) Identifikasi Tujuan Akhir: Tentukan tujuan akhir yang ingin dicapai dalam perencanaan. Ini bisa berupa tujuan akhir tunggal atau sekelompok tujuan terkait.
- 2) Identifikasi Tugas Subordinat: Identifikasi tugas-tugas yang diperlukan untuk mencapai tujuan akhir. Tugas-tugas ini harus saling terkait dan berkontribusi dalam mencapai tujuan akhir. Tugas-tugas ini juga dapat memiliki tujuan mereka sendiri yang harus dicapai.
- 3) Analisis Ketergantungan dan Hubungan Antara Tugas: Identifikasi ketergantungan dan hubungan antara tugas-tugas yang telah diidentifikasi. Identifikasi tugas-tugas yang harus diselesaikan sebelum tugas lain dapat dimulai, serta tugas-tugas yang dapat dilakukan secara independen.
- 4) Prioritaskan dan Atur Urutan Tugas: Prioritaskan tugas-tugas yang telah diidentifikasi berdasarkan ketergantungan dan urgensi. Tentukan urutan tugas yang harus diselesaikan terlebih dahulu, dan urutan tugas yang dapat dilakukan secara paralel atau bergantian.
- 5) Pecahkan Tugas Menjadi Subtugas: Jika tugas yang diidentifikasi masih terlalu kompleks, pecahkan tugas tersebut menjadi subtugas yang lebih kecil dan lebih mudah dipecahkan. Teruslah melakukan dekomposisi hingga mencapai tingkat kecil yang dapat dipecahkan dengan mudah.
- 6) Peleburan Subtugas: Setelah subtugas dipecahkan, lakukan langkah-langkah untuk menggabungkan hasil dari subtugas-subtugas tersebut sehingga mencapai tujuan akhir. Ini melibatkan penggabungan hasil, penyesuaian, dan koordinasi antara subtugas-subtugas tersebut.

Teknik dekomposisi masalah dalam perencanaan membantu mengurangi kompleksitas perencanaan dan memfasilitasi pemecahan masalah secara bertahap. Dengan memecah masalah perencanaan yang besar menjadi submasalah yang lebih kecil, lebih mudah untuk merancang strategi dan merencanakan langkah-langkah yang diperlukan untuk mencapai tujuan akhir.

## **A. Goal Stack Planning (GSP)**

Goal Stack Planning (GSP) adalah salah satu pendekatan dalam perencanaan yang mengorganisir tugas-tugas perencanaan dalam bentuk tumpukan tujuan (goal stack). Pada awalnya, tujuan akhir yang ingin dicapai ditempatkan di puncak tumpukan tujuan. Kemudian, perencanaan dilakukan dengan menguraikan tujuan tersebut menjadi tujuan-tujuan yang lebih kecil dan lebih spesifik.

Proses GSP dimulai dengan mempopulasi tumpukan tujuan dengan tujuan akhir. Selanjutnya, perencanaan dilakukan dengan mencari tindakan atau subtujuan yang dapat memenuhi tujuan yang ada di puncak tumpukan tujuan. Jika tindakan atau subtujuan tersebut tidak dapat langsung dilakukan, maka tujuan tersebut dipindahkan ke tumpukan tujuan yang lebih rendah. Proses ini berlanjut hingga semua tujuan dalam tumpukan tujuan terpenuhi.

Keuntungan dari GSP adalah fleksibilitasnya dalam menghadapi situasi yang kompleks dan kemampuannya untuk mengatasi ketergantungan antara tujuan. Dengan menggunakan tumpukan tujuan, GSP memungkinkan perencanaan yang adaptif dan memungkinkan perubahan prioritas dalam mencapai tujuan akhir.

## **B. Constraint Posting(CP)**

Constraint Posting (CP) adalah pendekatan dalam perencanaan yang menangani keterbatasan atau kendala yang ada dalam mencapai tujuan. Dalam CP, kendala-kendala ini dinyatakan secara eksplisit dan diberlakukan dalam proses perencanaan.

Proses CP dimulai dengan mendefinisikan kendala-kendala yang ada dalam mencapai tujuan akhir. Kendala-kendala ini dapat berupa keterbatasan sumber daya, batasan waktu, atau batasan lain yang relevan. Kemudian, kendala-kendala ini diungkapkan dan diterapkan dalam perencanaan.

Dalam CP, tindakan-tindakan atau langkah-langkah yang tidak memenuhi kendala-kendala yang ada dapat dihapus dari pertimbangan perencanaan. Hal ini memungkinkan fokus pada tindakan-tindakan yang memenuhi kendala-kendala yang telah ditetapkan.

Keuntungan dari CP adalah kemampuannya dalam mempertimbangkan keterbatasan-keterbatasan yang ada secara langsung dalam perencanaan. Dengan menetapkan kendala-kendala tersebut, CP dapat membantu memastikan bahwa perencanaan yang dihasilkan mematuhi keterbatasan yang ada, sehingga menghasilkan rencana yang lebih realistik dan memungkinkan untuk diimplementasikan dalam praktik.

Pendekatan GSP dan CP dapat digunakan secara terpisah atau kombinasi tergantung pada karakteristik masalah perencanaan yang dihadapi dan preferensi pengguna.

## **BAB VI**

### **LEARNING**

Dalam konteks pembelajaran (learning) adalah proses memperoleh pengetahuan, keterampilan, sikap, atau kompetensi melalui studi, pengalaman, atau pengajaran. Dalam konteks kecerdasan buatan (artificial intelligence/AI), pembelajaran merujuk pada kemampuan sistem komputer atau mesin untuk meningkatkan kinerjanya dalam suatu tugas berdasarkan pengalaman atau data.

Ada beberapa jenis pembelajaran dalam AI:

- 1) Pembelajaran Berbimbing (Supervised Learning): Dalam pembelajaran berbimbing, model pembelajaran mesin dilatih menggunakan dataset berlabel, di mana setiap contoh data terkait dengan target atau keluaran yang diketahui. Model ini belajar untuk memetakan fitur input ke keluaran yang benar dengan melakukan generalisasi dari contoh-contoh berlabel. Pembelajaran berbimbing umumnya digunakan untuk tugas-tugas seperti klasifikasi dan regresi.
- 2) Pembelajaran Tanpa Bimbingan (Unsupervised Learning): Pembelajaran tanpa bimbingan melibatkan pelatihan model pada dataset tak berlabel, di mana target atau keluaran tidak diketahui. Model ini belajar untuk menemukan pola, hubungan, atau struktur dalam data tanpa panduan eksplisit. Klasterisasi dan reduksi dimensionalitas adalah contoh tugas pembelajaran tanpa bimbingan.
- 3) Pembelajaran Penguatan (Reinforcement Learning): Pembelajaran penguatan melibatkan agen yang belajar berinteraksi dengan lingkungan dan membuat keputusan berdasarkan percobaan dan kesalahan. Agen menerima umpan balik berupa hadiah atau hukuman berdasarkan tindakannya, dan tujuannya adalah memaksimalkan hadiah akumulatif dari waktu ke waktu. Pembelajaran penguatan umumnya digunakan dalam tugas-tugas seperti bermain game, robotika, dan optimisasi.
- 4) Pembelajaran Semi-Bimbingan (Semi-Supervised Learning): Pembelajaran semi-bimbingan menggabungkan elemen-elemen pembelajaran berbimbing dan tanpa bimbingan. Ini menggunakan sejumlah kecil data berlabel dan sejumlah besar data tak berlabel untuk melatih model. Data tak berlabel membantu dalam menangkap struktur atau distribusi data yang mendasarinya, sementara data berlabel memberikan panduan dalam mempelajari tugas spesifik.
- 5) Pembelajaran Transfer (Transfer Learning): Pembelajaran transfer melibatkan memanfaatkan pengetahuan atau representasi yang telah dipelajari dari satu tugas atau domain untuk meningkatkan kinerja pada tugas atau domain terkait lainnya. Ini memungkinkan penggunaan kembali model atau fitur yang telah dilatih sebelumnya, yang dapat secara signifikan mengurangi kebutuhan akan data berlabel dan waktu pelatihan.

- 6) Pembelajaran Dalam Jaringan (Deep Learning): Pembelajaran dalam jaringan adalah subbidang pembelajaran mesin yang berfokus pada melatih jaringan saraf buatan dengan banyak lapisan untuk mempelajari representasi hierarkis data. Pembelajaran dalam jaringan telah mencapai keberhasilan signifikan dalam berbagai

## A. ID3

ID3 (Iterative Dichotomiser 3) adalah sebuah algoritma pembelajaran mesin yang digunakan untuk membangun pohon keputusan dari dataset berlabel. Algoritma ID3 menggunakan pendekatan top-down untuk membagi dataset menjadi subgrup yang lebih kecil berdasarkan fitur-fitur yang ada. Algoritma ini mengambil langkah-langkah berikut dalam membangun pohon keputusan:

- 1) Memilih Fitur Terbaik: ID3 memilih fitur yang paling informatif untuk membagi dataset menjadi subgrup yang lebih homogen. Ini dilakukan dengan menggunakan metrik seperti Gain Information (Informasi Keuntungan) atau Gini Index (Indeks Gini).
- 2) Membuat Node dalam Pohon: Setelah fitur terbaik dipilih, ID3 membuat node dalam pohon keputusan yang mewakili fitur tersebut.
- 3) Memisahkan Dataset: ID3 memisahkan dataset berdasarkan nilai-nilai fitur yang terkait dengan node saat ini. Setiap nilai fitur menjadi cabang dari node tersebut.
- 4) Rekursif: ID3 melakukan langkah-langkah di atas secara rekursif pada setiap subgrup data yang dihasilkan dari pemisahan. Proses ini berlanjut hingga semua data terbagi secara homogen atau hingga kriteria berhenti lainnya terpenuhi.
- 5) Menentukan Label di Daun: Ketika semua cabang dan subgrup telah ditentukan, ID3 menentukan label atau kelas yang akan diberikan pada daun pohon keputusan. Ini biasanya didasarkan pada mayoritas kelas dari data yang ada di subgrup tersebut.

Kelemahan ID3 adalah rentan terhadap overfitting, di mana pohon keputusan dapat terlalu kompleks dan tidak dapat umum, serta sensitif terhadap atribut dengan banyak nilai. Untuk mengatasi beberapa kelemahan ini, algoritma-algoritma berbasis ID3 yang lebih canggih seperti C4.5 dan CART (Classification and Regression Trees) dikembangkan.

## B. C.45

C4.5 adalah pengembangan dari algoritma ID3 yang memiliki beberapa perbaikan dan peningkatan. C4.5 juga membangun pohon keputusan berdasarkan dataset berlabel, namun memiliki beberapa fitur tambahan:

- 1) Penanganan Data Kategorikal dan Kontinu: C4.5 mampu menangani baik fitur kategorikal maupun kontinu dalam dataset. Itu menggunakan pengelompokan untuk fitur kontinu dan mempertimbangkan ambang batas untuk membagi data menjadi subgrup.
- 2) Penggunaan Gain Ratio: C4.5 menggunakan metrik Gain Ratio sebagai pemilih fitur terbaik untuk membagi dataset. Gain Ratio memperhitungkan jumlah dan keunikan nilai-nilai fitur untuk menghindari bias terhadap fitur dengan banyak nilai.
- 3) Penanganan Missing Values: C4.5 dapat mengatasi data yang hilang atau tidak lengkap dengan melakukan penanganan khusus untuk nilai-nilai yang hilang.
- 4) Pemangkasan Pohon (Pruning): C4.5 menyediakan mekanisme pemangkasan pohon untuk mengurangi overfitting. Ini memotong cabang-cabang yang tidak memberikan peningkatan yang signifikan dalam klasifikasi.

C4.5 adalah salah satu algoritma yang populer digunakan untuk membangun pohon keputusan dalam pembelajaran mesin.

## **BAB VII**

### **JARINGAN SYARAF TIRUAN (JST)**

Jaringan Syaraf Tiruan (Artificial Neural Network/ANN) adalah model komputasi yang terinspirasi oleh struktur dan fungsi jaringan saraf dalam sistem saraf manusia. Jaringan syaraf tiruan terdiri dari banyak unit pemrosesan sederhana yang disebut neuron, yang terhubung melalui koneksi bobot. Neuron-neuron ini bekerja secara paralel untuk memproses informasi dan mempelajari pola-pola yang terkandung dalam data.

Struktur dasar dari jaringan syaraf tiruan terdiri dari tiga jenis layer, yaitu:

- 1) Input Layer: Layer ini menerima input data yang akan diproses oleh jaringan. Setiap neuron dalam input layer mewakili fitur-fitur atau atribut-atribut dari data yang masuk.
- 2) Hidden Layer: Hidden layer adalah layer yang berada di antara input layer dan output layer. Layer ini bertanggung jawab untuk melakukan pemrosesan komputasi dan pembelajaran di dalam jaringan. Jumlah dan ukuran hidden layer dapat bervariasi tergantung pada kompleksitas masalah yang ingin dipecahkan.
- 3) Output Layer: Output layer adalah layer terakhir dari jaringan syaraf tiruan yang menghasilkan output atau prediksi berdasarkan hasil pemrosesan jaringan. Jumlah neuron dalam output layer sesuai dengan jumlah kelas atau nilai-nilai yang ingin diprediksi.

#### **A. Konsep Dasar JST**

Konsep dasar Jaringan Syaraf Tiruan (JST) melibatkan beberapa elemen utama yang membentuk struktur dan fungsi jaringan tersebut. Berikut adalah konsep dasar yang perlu dipahami dalam JST:

##### **1. Neuron**

Neuron adalah unit pemrosesan dasar dalam JST, yang mirip dengan neuron dalam sistem saraf manusia. Setiap neuron menerima input dari neuron sebelumnya, memproses informasi dengan menggunakan fungsi aktivasi, dan menghasilkan output. Output neuron ini kemudian menjadi input untuk neuron-neuron berikutnya.

##### **2. Bobot**

Bobot adalah parameter numerik yang terhubung ke setiap koneksi antara neuron. Bobot mengatur kontribusi relatif dari input yang diterima oleh neuron dalam pemrosesan informasi. Bobot ini diatur selama proses pelatihan jaringan untuk mempengaruhi pembelajaran dan hasil prediksi.

### **3. Fungsi Aktivasi**

Fungsi aktivasi diterapkan pada setiap neuron untuk mengubah nilai input menjadi output yang berbeda. Fungsi aktivasi ini memungkinkan jaringan untuk memodelkan hubungan non-linear antara input dan output. Contoh fungsi aktivasi yang umum digunakan adalah fungsi sigmoid, fungsi ReLU (Rectified Linear Unit), dan fungsi tangen hiperbolik.

### **4. Layer**

JST terdiri dari beberapa layer yang terhubung satu sama lain. Layer pertama disebut input layer, menerima input data. Layer terakhir disebut output layer, menghasilkan output atau prediksi. Layer di antara input dan output layer disebut hidden layer. Setiap neuron dalam layer terhubung ke neuron dalam layer sebelumnya dan layer setelahnya.

### **5. Feedforward**

Feedforward adalah proses mengirimkan input melalui jaringan dari input layer ke output layer. Informasi mengalir maju melalui koneksi-koneksi antar neuron hingga mencapai output layer. Selama feedforward, setiap neuron melakukan komputasi berdasarkan bobot dan fungsi aktivasinya sendiri.

### **6. Backpropagation**

Backpropagation adalah algoritma yang digunakan untuk melatih jaringan syaraf tiruan. Dalam algoritma ini, kesalahan atau perbedaan antara output yang dihasilkan oleh jaringan dan output yang diharapkan dikembalikan ke belakang melalui jaringan. Bobot-bobot yang menghubungkan neuron-neuron diperbarui berdasarkan kesalahan ini untuk meningkatkan kinerja jaringan.

Dengan konsep dasar ini, JST dapat memodelkan hubungan kompleks antara input dan output, dan dapat belajar dari data untuk membuat prediksi atau pengambilan keputusan. JST telah berhasil diterapkan dalam berbagai aplikasi AI, seperti pengenalan pola, klasifikasi, prediksi, dan pengenalan suara.

## **B. Model Syaraf Tiruan (Neuron)**

Ada beberapa model jaringan syaraf tiruan (JST) yang umum digunakan dalam bidang kecerdasan buatan. Berikut ini adalah beberapa model JST yang populer:

### **1. Jaringan Syaraf Tiruan Feedforward (Feedforward Neural Network/FFNN)**

Model ini terdiri dari layer-layer neuron yang saling terhubung secara sekuensial. Informasi mengalir maju dari input layer melalui hidden layer(s) ke output layer tanpa ada koneksi mundur. JST feedforward adalah model paling dasar dalam JST dan sering digunakan dalam berbagai tugas pembelajaran mesin.

## **2. Jaringan Syaraf Tiruan Rekurrent (Recurrent Neural Network/RNN)**

Model ini memiliki koneksi mundur yang memungkinkan aliran informasi berjalan ke arah belakang dalam jaringan. Dalam RNN, setiap neuron menerima input saat ini serta input yang dihasilkan pada langkah sebelumnya. Model ini cocok untuk memproses data berurutan, seperti teks atau suara, di mana konteks sebelumnya penting.

## **3. Jaringan Syaraf Tiruan Konvolusional (Convolutional Neural Network/CNN)**

Model ini dirancang khusus untuk memproses data grid, seperti gambar. CNN menggunakan operasi konvolusi untuk mengekstraksi fitur-fitur penting dari data dan menggunakan layer-layer konvolusi, pooling, dan fully connected untuk menghasilkan prediksi. Model ini sangat sukses dalam tugas-tugas pengenalan gambar dan penglihatan komputer.

## **4. Jaringan Syaraf Tiruan Long Short-Term Memory (LSTM)**

Model LSTM adalah jenis RNN yang dirancang khusus untuk mengatasi masalah peringatan yang terjadi dalam RNN tradisional. LSTM menggunakan unit memori khusus yang memungkinkan jaringan untuk mengingat informasi dalam jangka panjang, sehingga cocok untuk memproses data yang memerlukan pemahaman konteks yang lebih luas, seperti teks atau suara.

## **5. Jaringan Syaraf Tiruan Gated Recurrent Unit (GRU)**

Model GRU adalah variasi lain dari RNN yang juga dirancang untuk mengatasi masalah peringatan dalam RNN tradisional. GRU memiliki arsitektur yang lebih sederhana daripada LSTM, tetapi masih mampu mengingat informasi jangka panjang dalam urutan data.

Setiap model JST memiliki kekuatan dan kelemahan masing-masing dan lebih cocok untuk tugas-tugas tertentu. Pemilihan model yang tepat tergantung pada jenis data yang dihadapi dan jenis tugas yang ingin diselesaikan.

## **C. Aktivasi & Arsitektur Jaringan pada JST**

Aktivasi dalam jaringan syaraf tiruan (JST) mengacu pada fungsi matematis yang diterapkan pada output neuron untuk memodulasi sinyal yang diteruskan ke neuron selanjutnya. Aktivasi tersebut memungkinkan JST untuk mempelajari pola-pola kompleks dalam data dengan memperkenalkan non-linearitas. Beberapa fungsi aktivasi yang umum digunakan dalam JST antara lain:

- 1) Fungsi Sigmoid: Fungsi sigmoid menghasilkan output dalam rentang 0 hingga 1. Ini adalah fungsi aktivasi yang paling umum digunakan dalam JST. Fungsi sigmoid dapat digunakan di layer hidden dan output dalam JST. Formula fungsi sigmoid adalah:

$$f(x) = 1 / (1 + \exp(-x))$$

- 2) Fungsi ReLU (Rectified Linear Unit): Fungsi ReLU menghasilkan output 0 untuk input negatif dan output input untuk input positif. Ini adalah fungsi aktivasi yang sederhana tetapi efektif dan sering digunakan dalam layer-layer hidden dalam JST. Formula fungsi ReLU adalah:

$$f(x) = \max(0, x)$$

- 3) Fungsi Tanh (Tangent Hyperbolic): Fungsi tanh menghasilkan output dalam rentang -1 hingga 1. Ini mirip dengan fungsi sigmoid tetapi memiliki rentang yang sedikit lebih besar. Fungsi tanh digunakan dalam layer-layer hidden dalam JST. Formula fungsi tanh adalah:

$$f(x) = (\exp(x) - \exp(-x)) / (\exp(x) + \exp(-x))$$

Arsitektur jaringan pada JST merujuk pada struktur dan hubungan antara neuron-neuron dalam jaringan. Ada berbagai arsitektur JST yang dapat digunakan, tergantung pada jenis tugas dan data yang sedang diproses. Beberapa arsitektur JST yang umum digunakan adalah:

- 1) Jaringan Syaraf Tiruan Feedforward (Feedforward Neural Network/FFNN): Ini adalah arsitektur dasar JST, di mana informasi mengalir maju dari input layer melalui layer-layer hidden ke output layer tanpa ada koneksi mundur. Arsitektur ini cocok untuk banyak tugas pembelajaran mesin seperti klasifikasi dan regresi.
- 2) Jaringan Syaraf Tiruan Rekurrent (Recurrent Neural Network/RNN): Arsitektur RNN memiliki koneksi mundur yang memungkinkan aliran informasi berjalan ke arah belakang dalam jaringan. Ini memungkinkan JST untuk memproses data berurutan dan mempertahankan informasi kontekstual dari langkah sebelumnya. Arsitektur ini berguna untuk tugas-tugas seperti pengenalan ucapan dan penerjemahan mesin.
- 3) Jaringan Syaraf Tiruan Konvolusional (Convolutional Neural Network/CNN): Arsitektur CNN dirancang khusus untuk memproses data grid, seperti gambar. Arsitektur ini menggunakan operasi konvolusi untuk mengekstraksi fitur-fitur penting dari data dan mempelajari representasi spasial dari data. CNN sangat efektif dalam tugas-tugas pengenalan gambar dan penglihatan komputer.
- 4) Jaringan Syaraf Tiruan Modular: Arsitektur JST modular terdiri dari beberapa submodule yang saling terhubung. Setiap submodule dapat memiliki arsitektur yang berbeda dan mengambil tanggung jawab pemrosesan data tertentu. Arsitektur modular memungkinkan pembagian tugas dan pemrosesan paralel yang efisien.

Setiap arsitektur JST memiliki kelebihan dan kelemahan masing-masing, dan pemilihan arsitektur yang tepat tergantung pada jenis tugas dan data yang sedang diproses.

## **D. Supervised Learning & Unsupervised Learning**

Supervised Learning dan Unsupervised Learning adalah dua paradigma utama dalam pembelajaran mesin (machine learning) yang digunakan untuk mempelajari pola dan menghasilkan pemodelan prediktif. Berikut adalah penjelasan singkat tentang keduanya:

### **1. Supervised Learning (Pembelajaran Berawasan)**

Dalam supervised learning, algoritma pembelajaran mesin diberi data yang sudah dilabeli atau berlabel. Artinya, setiap sampel data dalam dataset training memiliki atribut atau fitur yang diberi label atau kelas yang diketahui sebelumnya. Tujuan dari supervised learning adalah untuk mempelajari pola atau hubungan antara fitur dan label yang ada, sehingga algoritma dapat membuat prediksi yang akurat pada data baru yang tidak diberi label. Contoh algoritma supervised learning termasuk Decision Trees, Support Vector Machines (SVM), dan Neural Networks.

### **2. Unsupervised Learning (Pembelajaran Tanpa Pengawasan)**

Dalam unsupervised learning, algoritma pembelajaran mesin diberikan data yang tidak dilabeli atau tanpa label. Tidak ada informasi tentang label atau kelas yang diketahui sebelumnya. Tujuan dari unsupervised learning adalah untuk menemukan struktur tersembunyi dalam data, mengidentifikasi pola-pola yang tidak diketahui sebelumnya, atau mengelompokkan data menjadi kelompok-kelompok yang serupa berdasarkan kesamaan fitur. Dalam unsupervised learning, algoritma bekerja secara mandiri untuk menemukan pola atau struktur dalam data tanpa arahan eksternal. Contoh algoritma unsupervised learning termasuk K-means Clustering, Hierarchical Clustering, dan Principal Component Analysis (PCA).

Perbedaan utama antara supervised learning dan unsupervised learning terletak pada ketersediaan informasi label dalam data. Supervised learning menggunakan data dengan label yang diketahui untuk membuat prediksi pada data baru, sedangkan unsupervised learning mengelompokkan atau mengungkapkan pola dalam data tanpa informasi label sebelumnya.

## **BAB VIII**

### **ALGORITMA GENETIKA (AG)**

Algoritma Genetika (Genetic Algorithms) adalah sebuah metode optimisasi dan pencarian heuristik yang terinspirasi oleh proses evolusi dalam alam. Algoritma ini menggunakan konsep seleksi alam, reproduksi, dan mutasi untuk mencari solusi optimal dalam ruang pencarian yang kompleks.

#### **A. Komponen-komponen AG**

##### **1. Populasi**

Populasi dalam algoritma genetika merujuk pada kumpulan individu-individu atau solusi-solusi potensial yang ada pada setiap generasi. Setiap individu dalam populasi mewakili suatu kemungkinan solusi untuk masalah yang sedang diselesaikan.

##### **2. Genotype**

Genotype adalah representasi genetik dari individu dalam populasi. Biasanya, genotype direpresentasikan dalam bentuk urutan gen atau bit yang mengodekan informasi tentang solusi yang diusulkan. Misalnya, dalam masalah optimisasi, genotipe dapat berupa urutan nilai-nilai parameter yang mencirikan solusi.

##### **3. Fungsi Fitness**

Fungsi fitness adalah metrik yang digunakan untuk mengevaluasi sejauh mana suatu individu cocok dengan tujuan yang ditetapkan dalam masalah yang sedang diselesaikan. Fungsi fitness mengukur kualitas atau kinerja individu dalam mencapai solusi yang diinginkan. Nilai fitness biasanya ditentukan berdasarkan kriteria yang telah ditentukan sebelumnya.

##### **4. Seleksi**

Seleksi adalah proses dalam algoritma genetika di mana individu-individu terbaik dipilih dari populasi untuk menjadi orang tua dalam generasi berikutnya. Seleksi biasanya didasarkan pada nilai fitness individu, di mana individu dengan nilai fitness yang lebih tinggi memiliki peluang yang lebih besar untuk dipilih. Prinsip seleksi adalah bahwa individu yang lebih baik dalam mencapai tujuan akan memberikan kontribusi genetik yang lebih baik untuk generasi berikutnya.

##### **5. Rekombinasi (Crossover)**

Rekombinasi, juga dikenal sebagai crossover, adalah proses di mana bagian-bagian genotipe antara dua individu dipertukarkan untuk menghasilkan keturunan baru. Proses ini mensimulasikan persilangan genetik dalam reproduksi seksual alami. Rekombinasi menghasilkan variasi genetik baru dalam populasi dan membantu menjaga keragaman dan eksplorasi ruang pencarian.

## **6. Mutasi**

Mutasi adalah proses acak di mana beberapa gen dalam genotipe individu mengalami perubahan. Tujuan mutasi adalah memperkenalkan variasi genetik baru yang tidak ada dalam populasi saat ini. Mutasi membantu dalam eksplorasi ruang pencarian dan mencegah populasi terjebak dalam minimum lokal. Tingkat mutasi yang tepat penting untuk menjaga keseimbangan antara eksplorasi dan eksloitasi.

## **7. Penggantian Generasi**

Setelah dilakukan seleksi, rekombinasi, dan mutasi, generasi baru individu dibentuk untuk generasi berikutnya. Dalam proses penggantian generasi, individu-individu baru dan beberapa individu terbaik dari generasi sebelumnya digunakan untuk membentuk populasi baru. Dengan demikian, proses evaluasi, seleksi, rekombinasi, dan mutasi diulang dalam setiap generasi hingga kondisi berhenti terpenuhi atau solusi yang memadai ditemukan.

## **BAB IX**

### **COGNITIF SCIENCE**

Ilmu kognitif adalah studi interdisipliner dan ilmiah tentang pikiran dan prosesnya dengan masukan dari linguistik, psikologi, ilmu saraf, filsafat, ilmu komputer/kecerdasan buatan, dan antropologi. Ini mengkaji sifat, tugas, dan fungsi kognisi (dalam arti luas).

Prinsip utama ilmu kognitif adalah bahwa pemahaman lengkap tentang pikiran/otak tidak dapat dicapai hanya dengan mempelajari satu tingkat. Contohnya adalah masalah mengingat nomor telepon dan mengingatnya nanti .

Ilmu kognitif adalah bidang interdisipliner yang dikhawasukan untuk mengeksplorasi sifat proses kognitif seperti persepsi, penalaran, memori, perhatian, bahasa, citra, kontrol motorik, dan pemecahan masalah .

#### **A. Persepsi**

Dalam ilmu kognitif, persepsi merujuk pada proses mental yang melibatkan interpretasi dan pemahaman informasi sensorik yang diterima dari lingkungan. Ini adalah proses bagaimana kita mengorganisasikan dan memberikan makna pada data sensorik yang masuk, seperti penglihatan, pendengaran, sentuhan, bau, dan rasa.

Dalam konteks ilmu kognitif, ada beberapa konsep penting yang terkait dengan persepsi:

- 1) Pengolahan Informasi: Persepsi melibatkan pengolahan informasi di dalam pikiran kita. Data sensorik yang masuk diolah dan dianalisis dalam pikiran untuk memberikan makna dan pemahaman yang lebih dalam tentang lingkungan.
- 2) Pengenalan Pola: Persepsi melibatkan kemampuan untuk mengenali dan mengidentifikasi pola dalam data sensorik. Misalnya, kemampuan kita untuk mengenali wajah orang, mengidentifikasi objek berdasarkan bentuknya, atau memahami pola bunyi untuk mengenali kata-kata.
- 3) Pengorganisasian: Persepsi melibatkan pengorganisasian informasi sensorik ke dalam struktur dan kategori yang bermakna. Misalnya, dalam penglihatan, kita mengorganisasikan elemen visual menjadi objek-objek yang terpisah dan mengidentifikasi relasi antara objek-objek tersebut.
- 4) Interpretasi: Persepsi melibatkan interpretasi informasi sensorik berdasarkan pengetahuan, pengalaman, dan konteks yang ada. Kita menggunakan pengetahuan kita tentang dunia untuk memberikan makna dan pemahaman pada data sensorik yang diterima.
- 5) Konstruksi Subyektif: Persepsi bukanlah refleksi langsung dari dunia luar, tetapi merupakan konstruksi subyektif yang dipengaruhi oleh faktor-faktor seperti

pengetahuan, harapan, motivasi, dan interpretasi kita sendiri. Kita menciptakan representasi mental yang unik berdasarkan persepsi kita terhadap dunia.

Dalam ilmu kognitif, persepsi merupakan area penelitian yang luas dan melibatkan berbagai disiplin ilmu seperti psikologi, neurosains, ilmu komputer, dan filsafat. Tujuan utama penelitian persepsi adalah untuk memahami bagaimana kita mengalami dunia dan bagaimana proses kognitif mempengaruhi persepsi kita.

## B. Pemrosesan Bahasa

Pemrosesan bahasa dalam ilmu kognitif melibatkan studi tentang bagaimana pikiran manusia memahami, menghasilkan, dan menggunakan bahasa. Ini melibatkan pemahaman tentang struktur bahasa, pemrosesan kata dan frasa, pemahaman makna, serta produksi dan pemahaman kalimat.

Berikut adalah beberapa aspek penting dalam pemrosesan bahasa dalam ilmu kognitif:

- 1) Pemahaman Bahasa: Studi pemrosesan bahasa melibatkan pemahaman tentang bagaimana pikiran manusia memahami dan menginterpretasikan kata-kata, frasa, dan kalimat. Ini melibatkan proses pemahaman sintaksis, semantik, dan pragmatik bahasa.
- 2) Pengenalan Kata: Pengenalan kata melibatkan kemampuan untuk mengenali kata-kata dalam percakapan atau teks tertulis. Ini melibatkan pemrosesan informasi seperti bentuk kata, arti kata, dan konteks penggunaannya.
- 3) Pemrosesan Gramatikal: Pemrosesan gramatikal melibatkan pemahaman dan produksi struktur kalimat yang benar. Ini melibatkan pemahaman tentang tata bahasa, sintaksis, dan struktur frasa.
- 4) Pemahaman Makna: Pemahaman makna melibatkan pemrosesan dan interpretasi makna kata, frasa, dan kalimat. Ini melibatkan pemahaman sinonim, antonim, konotasi, dan referensi.
- 5) Produksi Bahasa: Produksi bahasa melibatkan kemampuan untuk menghasilkan kalimat-kalimat yang memenuhi tujuan komunikasi. Ini melibatkan pemilihan kata, struktur kalimat, dan pengaturan frasa yang sesuai dengan konteks dan tujuan komunikasi.
- 6) Kognisi Tertutup dan Terbuka: Pemrosesan bahasa dapat dibagi menjadi dua jenis kognisi: kognisi tertutup dan kognisi terbuka. Kognisi tertutup melibatkan pemahaman dan produksi bahasa berdasarkan aturan dan struktur yang terbatas, seperti dalam tugas tatabahasa. Kognisi terbuka melibatkan pemahaman dan produksi bahasa yang lebih fleksibel dan kreatif, seperti dalam percakapan sehari-hari.

Penelitian dalam pemrosesan bahasa dalam ilmu kognitif melibatkan penggunaan metode dan pendekatan dari berbagai disiplin ilmu, termasuk psikologi kognitif, neurosains kognitif, linguistik, dan ilmu komputer. Tujuannya adalah untuk memahami bagaimana pikiran manusia memproses dan menggunakan bahasa, serta mengembangkan model dan teori yang menjelaskan proses ini secara mendalam.

## C. Memori

Dalam konteks kecerdasan buatan (artificial intelligence/AI), memori merujuk pada kemampuan sistem AI untuk menyimpan dan mengingat informasi yang diperlukan untuk tugas tertentu. Memori dalam AI dapat berupa struktur data dan mekanisme yang digunakan untuk menyimpan pengetahuan, pengalaman, atau informasi penting yang diperlukan oleh sistem untuk mengambil keputusan atau melakukan tugas yang kompleks.

Berikut ini adalah beberapa bentuk memori yang digunakan dalam AI:

- 1) Memori Pengetahuan: Memori pengetahuan atau basis pengetahuan adalah struktur data yang digunakan untuk menyimpan informasi terstruktur tentang dunia atau domain tertentu. Basis pengetahuan dapat berupa basis pengetahuan berbasis aturan (rule-based knowledge base) yang berisi aturan-aturan logika yang menggambarkan hubungan antara entitas dan fakta, atau dapat berupa basis pengetahuan berbasis graf (graph-based knowledge base) yang menggambarkan entitas dan hubungannya dalam bentuk graf.
- 2) Memori Pengalaman: Memori pengalaman adalah mekanisme yang digunakan oleh sistem AI untuk menyimpan pengalaman yang diperoleh selama interaksi dengan lingkungannya. Memori pengalaman digunakan untuk belajar dari pengalaman masa lalu dan menggunakannya untuk pengambilan keputusan atau pemecahan masalah di masa depan. Metode seperti reinforcement learning menggunakan memori pengalaman untuk mengoptimalkan keputusan berdasarkan umpan balik yang diterima dari lingkungan.
- 3) Memori Jangka Pendek dan Jangka Panjang: Dalam konteks AI, memori jangka pendek (short-term memory) adalah komponen yang menyimpan informasi yang relevan dan penting untuk tugas yang sedang dilakukan saat ini. Informasi ini mungkin tidak disimpan dalam waktu yang lama dan dapat segera dilupakan setelah tugas selesai. Di sisi lain, memori jangka panjang (long-term memory) adalah komponen yang digunakan untuk menyimpan informasi jangka panjang dan pengetahuan yang lebih permanen.
- 4) Memori Asosiatif: Memori asosiatif adalah jenis memori yang memungkinkan pencarian dan pemulihan informasi berdasarkan korelasi atau asosiasi antara input dan output. Memori asosiatif dapat digunakan dalam konteks pengenalan pola,

pengenalan suara, dan pemrosesan bahasa alami, di mana informasi yang relevan dapat diambil berdasarkan kesamaan atau hubungan asosiatif.

Memori dalam AI bertujuan untuk mendukung pembelajaran, penalaran, dan pengambilan keputusan yang lebih baik dan efektif oleh sistem AI. Berbagai teknik dan algoritma digunakan untuk mengelola dan memanfaatkan memori dalam AI, termasuk teknik pengindeksan, penyimpanan terdistribusi, dan pengelolaan memori berbasis aturan.

## D. Perhatian

Perhatian adalah proses kognitif yang melibatkan pemilihan, pemuatan, dan pengendalian fokus mental pada stimulus yang relevan dari lingkungan. Dalam ilmu kognitif, perhatian dipelajari untuk memahami bagaimana pikiran manusia memproses informasi, mengatur perilaku, dan mempengaruhi pemrosesan kognitif yang lebih tinggi.

Berikut adalah beberapa aspek penting tentang perhatian dalam ilmu kognitif:

- 1) Selektivitas: Perhatian selektif adalah kemampuan untuk memilih dan memusatkan fokus pada stimulus yang relevan sementara mengabaikan stimulus yang tidak relevan. Ini memungkinkan kita untuk memproses informasi yang penting dan mengabaikan gangguan atau distraksi.
- 2) Pengalihan Perhatian: Pengalihan perhatian adalah kemampuan untuk memindahkan fokus perhatian dari satu stimulus ke stimulus lainnya. Ini terjadi ketika kita beralih dari satu tugas ke tugas lain, atau ketika kita beralih perhatian dari satu bagian lingkungan ke bagian lain yang menarik perhatian.
- 3) Kapasitas Terbatas: Perhatian memiliki kapasitas terbatas, artinya kita tidak dapat secara efektif memproses atau memusatkan perhatian pada semua stimulus sekaligus. Kita cenderung memilih stimulus yang paling relevan atau menarik perhatian, sementara yang lain mungkin diabaikan atau kurang diproses.
- 4) Daur Perhatian: Daur perhatian merujuk pada siklus yang berulang antara pemrosesan stimulus eksternal dan pemrosesan internal. Proses ini melibatkan pemrosesan stimulus, pemantauan, evaluasi, dan pengaturan fokus perhatian berdasarkan tugas dan tujuan yang ada.
- 5) Pengendalian Perhatian: Pengendalian perhatian adalah kemampuan untuk mengendalikan dan mengarahkan fokus perhatian sesuai dengan kebutuhan dan tujuan saat ini. Ini melibatkan kemampuan untuk menahan distraksi, mengabaikan stimulus yang tidak relevan, dan mengatur perhatian dengan fleksibilitas.

## E. Pengambilan Keputusan dan Kognisi

Pengambilan keputusan adalah proses mental yang melibatkan pemilihan tindakan atau pilihan dari berbagai alternatif yang tersedia. Ini melibatkan kognisi, yaitu proses

mental yang melibatkan persepsi, pemahaman, penilaian, evaluasi, dan pemrosesan informasi untuk mempengaruhi keputusan yang diambil.

Berikut adalah beberapa aspek penting tentang pengambilan keputusan dan kognisi:

- 1) Persepsi: Proses persepsi memainkan peran penting dalam pengambilan keputusan. Ini melibatkan interpretasi dan pemahaman informasi yang diterima dari lingkungan. Bagaimana kita melihat, mendengar, dan menginterpretasikan informasi akan mempengaruhi bagaimana kita mempersepsikan situasi dan membuat keputusan.
- 2) Pengetahuan dan Memori: Pengetahuan dan memori memainkan peran penting dalam pengambilan keputusan. Pengetahuan yang kita miliki tentang dunia dan pengalaman sebelumnya membentuk kerangka kerja mental yang digunakan untuk mengevaluasi informasi dan memahami konsekuensi dari keputusan yang mungkin diambil. Memori membantu kita mengakses informasi yang relevan untuk membuat keputusan yang baik.
- 3) Penilaian dan Evaluasi: Proses kognitif seperti penilaian dan evaluasi mempengaruhi pengambilan keputusan. Kita menilai nilai, risiko, dan manfaat dari berbagai alternatif yang tersedia sebelum memilih tindakan yang paling memadai. Evaluasi ini didasarkan pada preferensi pribadi, nilai-nilai, dan tujuan yang dimiliki individu.
- 4) Pemrosesan Informasi: Pengambilan keputusan melibatkan pemrosesan informasi yang ada. Ini meliputi identifikasi masalah, pengumpulan data, analisis, dan sintesis informasi yang relevan untuk membantu dalam memilih opsi yang paling baik. Proses ini melibatkan kognisi seperti perhatian, pemahaman, dan penalaran.
- 5) Ketidakpastian dan Risiko: Banyak keputusan yang dihadapi oleh manusia melibatkan ketidakpastian dan risiko. Kognisi manusia mencoba untuk mengatasi ketidakpastian ini melalui penilaian probabilitas, pengambilan keputusan berdasarkan informasi yang tersedia, dan penyesuaian rencana berdasarkan hasil yang diharapkan.

## F. Neuropsikologi

Neuropsikologi adalah cabang ilmu yang menggabungkan prinsip-prinsip psikologi dengan pengetahuan tentang fungsi otak dan sistem saraf untuk memahami hubungan antara otak, kognisi, dan perilaku. Neuropsikologi mempelajari bagaimana kerusakan atau gangguan pada otak dapat mempengaruhi fungsi kognitif dan perilaku manusia.

Berikut adalah beberapa konsep penting dalam neuropsikologi:

- 1) Fungsi Kognitif: Neuropsikologi mempelajari fungsi kognitif, yaitu kemampuan mental yang terkait dengan pemrosesan informasi, seperti perhatian, memori, bahasa, persepsi, pemecahan masalah, dan pengambilan keputusan. Melalui studi kasus pasien dengan kerusakan otak, neuropsikolog dapat mengidentifikasi hubungan antara kerusakan otak dan perubahan dalam fungsi kognitif.

- 2) Gangguan Neurologis dan Psikiatrik: Neuropsikologi membantu dalam memahami gangguan neurologis dan psikiatrik dengan mempelajari efek dari kerusakan otak atau gangguan saraf pada fungsi kognitif dan perilaku. Ini mencakup studi tentang demensia, stroke, cedera kepala, penyakit Alzheimer, Parkinson, skizofrenia, dan gangguan lainnya.
- 3) Evaluasi Neuropsikologis: Neuropsikolog melakukan evaluasi terhadap individu yang mengalami gangguan neurologis atau psikiatrik untuk mengidentifikasi perubahan dalam fungsi kognitif mereka. Evaluasi ini melibatkan tes dan instrumen khusus yang dirancang untuk mengukur berbagai aspek fungsi kognitif, seperti memori, perhatian, kemampuan berbicara, dan pemecahan masalah.
- 4) Rehabilitasi Neuropsikologis: Neuropsikologi juga berperan dalam proses rehabilitasi pasien dengan gangguan neurologis atau cedera otak. Melalui program rehabilitasi yang disesuaikan, neuropsikolog membantu pasien memulihkan atau mengkompensasi fungsi kognitif yang terganggu, meningkatkan kemandirian, dan mengoptimalkan kualitas hidup mereka.
- 5) Penggunaan Teknologi dan Pemetaan Otak: Teknologi neuroimaging seperti pencitraan resonansi magnetik (MRI) dan elektroensefalogram (EEG) digunakan dalam neuropsikologi untuk memetakan aktivitas otak saat individu melakukan tugas kognitif tertentu. Ini membantu dalam pemahaman tentang hubungan antara aktivitas otak dan fungsi kognitif.

Studi neuropsikologi memberikan wawasan yang berharga tentang hubungan kompleks antara otak, kognisi, dan perilaku. Pengetahuan ini dapat digunakan untuk mengembangkan intervensi rehabilitasi yang lebih efektif, mendiagnosis gangguan neurologis lebih awal, dan memahami mekanisme kognitif yang mendasari perilaku manusia.