

MAKALAH
“PRAKTIKUM ARTIFICIAL INTELLIGENCE”

“Disusun sebagai tugas akhir pada mata kuliah Perancangan dan Analisis Algoritma , dengan
Widya Darwin S.Pd., M.Pd.T”



Disusun Oleh :

Fajri Hidayatul Ihsan
21346007

PROGRAM STUDI S1 TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI PADANG
TAHUN 2022

KATA PENGANTAR

Puji syukur kita hanturkan kehadiran Tuhan Yang Maha Esa, yang telah melimpahkan rahmat dan karunianya kepada kita, sehingga kita dapat menyusun dan menyajikan makalah Perancangan dan Analisis Algoritma ini dengan tepat waktu. Tak lupa pula kita mengucapkan terima kasih kepada berbagai pihak yang telah memberikan dorongan dan motivasi. Sehingga makalah ini dapat tersusun dengan baik.

Makalah ini dibuat sebagai salah satu syarat untuk memenuhi tugas mata kuliah Perancangan dan Analisis Algoritma. 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 juga memohon maaf apabila dalam penulisan makalah ini terdapat kesalahan kata - kata, pengetikan dan kekeliruan, sehingga membingungkan pembaca dalam memahami maksud penulis.

Adapun makalah ini penulis rangkum dari beberapa sumber yang dapat dipercaya yang disajikan penulisnya, dirangkum dalam lembar Daftar Pustaka dengan harapan makalah ini dapat menambah 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, Mei 2023

Fajri Hidayatul Ihsan

BAB I

KECERDASAN BUATAN

A. Pengeritan artificial intelligence

Kecerdasan Buatan (Artificial Intelligence/AI) adalah bidang studi dalam ilmu komputer yang bertujuan untuk mengembangkan sistem atau mesin yang mampu melakukan tugas-tugas yang biasanya memerlukan kecerdasan manusia. Tujuan utama dari kecerdasan buatan adalah menciptakan mesin yang dapat berpikir, belajar, dan mengambil keputusan seperti manusia.

Definisi kecerdasan buatan telah berkembang seiring dengan perkembangan teknologi dan pemahaman tentang bagaimana komputer dan sistem cerdas dapat bekerja. Pada dasarnya, kecerdasan buatan melibatkan pengembangan algoritma dan model matematika yang memungkinkan komputer untuk menganalisis data, memahami konteks, mengambil keputusan, dan melakukan tugas-tugas yang kompleks.

Kecerdasan buatan dapat diterapkan dalam berbagai bidang, termasuk pemrosesan bahasa alami, visi komputer, pengenalan suara, robotika, sistem ekspert, kendaraan otonom, dan banyak lagi. Metode yang digunakan dalam kecerdasan buatan termasuk logika, statistik, pengenalan pola, jaringan saraf tiruan, pemrosesan bahasa alami, dan algoritma pembelajaran mesin.

Kecerdasan buatan dapat digolongkan menjadi dua kategori utama: kecerdasan buatan yang lemah (weak AI) dan kecerdasan buatan yang kuat (strong AI). Kecerdasan buatan yang lemah dirancang untuk menyelesaikan tugas-tugas spesifik dengan tingkat kecerdasan terbatas, sedangkan kecerdasan buatan yang kuat bertujuan untuk menciptakan entitas yang memiliki kecerdasan setara dengan manusia.

Namun, walaupun kecerdasan buatan telah mencapai kemajuan yang signifikan, tetap ada tantangan dalam menciptakan kecerdasan buatan yang mampu menyamai atau melebihi kecerdasan manusia dalam segala aspek. Perkembangan dan penggunaan kecerdasan buatan juga menimbulkan pertimbangan etis dan sosial yang penting, seperti privasi, keamanan, dan dampaknya terhadap lapangan pekerjaan.

Secara umum, kecerdasan buatan merupakan bidang yang terus berkembang dan memiliki potensi besar untuk mengubah cara kita hidup dan bekerja di masa depan.

B. Teknik pencarian

Dalam kecerdasan buatan, terdapat beberapa teknik atau metode yang digunakan untuk menyelesaikan kasus atau masalah. Berikut adalah beberapa teknik penyelesaian kasus yang umum digunakan dalam kecerdasan buatan:

1. Pencarian dan Penelusuran (Search and Exploration):

Teknik ini melibatkan eksplorasi berbagai kemungkinan solusi dalam ruang masalah yang diberikan. Berbagai algoritma pencarian, seperti algoritma DFS (Depth-First Search), BFS (Breadth-First Search), dan A* (A-star), digunakan untuk mencari solusi secara sistematis dengan menggambarkan ruang masalah dan menggerakkan agen atau entitas cerdas untuk mencapai tujuan.

2. Logika dan Pemrograman Logika (Logic and Logic Programming):

Metode ini melibatkan penggunaan logika formal untuk mewakili dan menganalisis masalah. Alat-alat seperti logika proposisional, logika predikat, dan aturan inferensi digunakan untuk memodelkan pengetahuan dan menyelidiki hubungan antara fakta dan aturan dalam sistem cerdas.

3. Pembelajaran Mesin (Machine Learning):

Pembelajaran mesin adalah metode yang menggunakan algoritma untuk melatih mesin agar dapat belajar dari data dan pengalaman untuk membuat keputusan atau melakukan tugas tertentu. Metode pembelajaran mesin termasuk pembelajaran terawasi (supervised learning), pembelajaran tanpa pengawasan (unsupervised learning), dan pembelajaran penguatan (reinforcement learning).

4. Jaringan Saraf Tiruan (Artificial Neural Networks):

Jaringan saraf tiruan adalah model matematika yang terinspirasi oleh struktur dan fungsi jaringan saraf biologis. Jaringan saraf tiruan digunakan untuk memproses data masukan melalui lapisan-lapisan neuron buatan, yang kemudian menghasilkan keluaran yang diinginkan. Metode ini digunakan dalam berbagai tugas seperti pengenalan pola, klasifikasi data, dan prediksi.

5. Logika Kabur (Fuzzy Logic):

Logika kabur digunakan untuk mengatasi ketidakpastian atau ketidakjelasan dalam pemodelan dan pengambilan keputusan. Dalam logika kabur, variabel dapat memiliki nilai yang bukan hanya benar atau salah, tetapi juga memiliki tingkat keanggotaan dalam suatu himpunan. Logika kabur berguna dalam pengambilan keputusan berbasis aturan, kontrol sistem, dan pemodelan yang melibatkan ketidakpastian.

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

NLP melibatkan pemahaman, pengolahan, dan generasi bahasa manusia oleh mesin atau sistem cerdas. Metode NLP termasuk analisis sintaksis, analisis semantik, pemodelan topik, penerjemahan mesin, dan pengenalan ucapan. NLP memungkinkan interaksi yang lebih alami antara manusia dan sistem cerdas.

BAB II

RUANG MASALAH & SISTEM PRODUKSI

A. Ruang masalah

Ruang masalah dalam kecerdasan buatan mencakup berbagai domain dan kasus yang dapat dipecahkan menggunakan teknik-teknik kecerdasan buatan. Beberapa contoh ruang masalah yang umum dalam kecerdasan buatan beserta kondisi yang mungkin mengikutinya adalah sebagai berikut:

1. Pengenalan Pola (Pattern Recognition):

- Ruang Masalah: Pengenalan wajah, pengenalan tulisan tangan, klasifikasi gambar, pengenalan suara.
- Kondisi: Data pelatihan yang mencakup contoh-contoh yang representatif, algoritma pembelajaran mesin yang cocok untuk tugas pengenalan pola, fitur-fitur yang relevan untuk dianalisis.

2. Pemrosesan Bahasa Alami (Natural Language Processing/NLP):

- Ruang Masalah: Penerjemahan mesin, analisis sentimen, pengenalan entitas bernama, sistem pertanyaan dan jawaban.
- Kondisi: Korpus data bahasa yang cukup besar, algoritma pemrosesan bahasa alami yang efektif, pemodelan bahasa yang akurat.

3. Pengenalan Suara (Speech Recognition):

- Ruang Masalah: Pengenalan ucapan, transkripsi otomatis, pengenalan pembicara.
- Kondisi: Dataset suara yang mencakup berbagai aksen dan variasi suara, model akustik dan model bahasa yang baik, teknik pemrosesan sinyal suara yang efektif.

4. Sistem Rekomendasi (Recommendation Systems):

- Ruang Masalah: Rekomendasi produk, rekomendasi konten, rekomendasi teman.
- Kondisi: Data pengguna dan preferensi yang memadai, algoritma rekomendasi yang sesuai, metode pengukuran kepuasan pengguna yang valid.

5. Kendaraan Otonom (Autonomous Vehicles):

- Ruang Masalah: Navigasi otonom, pengenalan objek, pengambilan keputusan dalam lalu lintas.
- Kondisi: Sensor yang andal dan akurat, pemodelan lingkungan yang tepat, sistem pengambilan keputusan yang adaptif.

6. Diagnosis Medis (Medical Diagnosis):

- Ruang Masalah: Diagnosis penyakit, prediksi risiko penyakit, pengklasifikasian gambar medis.

- Kondisi: Data medis yang mencukupi, model pembelajaran mesin yang terlatih dengan baik, kriteria diagnosa yang jelas.

7. Perencanaan dan Penjadwalan (Planning and Scheduling):

- Ruang Masalah: Penjadwalan tugas, perencanaan produksi, penjadwalan transportasi.
- Kondisi: Batasan-batasan dan preferensi yang jelas, algoritma perencanaan yang efisien, representasi yang tepat untuk masalah penjadwalan yang diberikan.

Setiap ruang masalah dalam kecerdasan buatan memiliki kondisi dan persyaratan khusus yang harus dipenuhi untuk mencapai solusi yang baik. Pemahaman yang baik tentang ruang masalah dan kondisi yang mengikutinya adalah kunci untuk merancang dan mengimplementasikan sistem cerdas yang efektif.

Dalam kecerdasan buatan, initial state (keadaan awal) dan goal state (keadaan tujuan) digunakan dalam pemodelan masalah agar sistem cerdas dapat melakukan penyelesaian masalah. Initial state adalah keadaan awal dari sistem atau lingkungan yang dihadapi, sedangkan goal state adalah keadaan yang ingin dicapai atau tujuan yang ingin dicapai oleh sistem cerdas. Berikut adalah contoh initial state dan goal state dalam beberapa kasus kecerdasan buatan:

1. Pencarian Jalan Terpendek:

- Initial State: Posisi awal agen atau entitas cerdas dalam peta atau ruang.
- Goal State: Posisi tujuan yang ingin dicapai oleh agen.

2. Permainan Catur:

- Initial State: Posisi awal bidak pada papan catur.
- Goal State: Keadaan di mana salah satu pemain mencapai kondisi checkmate (raja terjebak) atau tujuan lain dalam permainan.

3. Penerjemahan Mesin:

- Initial State: Teks dalam bahasa asal yang ingin diterjemahkan.
- Goal State: Teks hasil terjemahan dalam bahasa target yang diinginkan.

4. Diagnosis Medis:

- Initial State: Gejala atau informasi awal mengenai kondisi kesehatan pasien.
- Goal State: Diagnosis akurat tentang penyakit atau kondisi yang dialami pasien.

5. Sistem Rekomendasi:

- Initial State: Data profil pengguna dan preferensi yang diketahui.
- Goal State: Rekomendasi produk, konten, atau teman yang sesuai dengan preferensi pengguna.

6. Navigasi Kendaraan Otonom:

- Initial State: Posisi awal dan informasi sensor tentang lingkungan sekitar.
- Goal State: Posisi tujuan yang ingin dicapai oleh kendaraan otonom.

Initial state dan goal state ini membantu dalam merumuskan masalah dan memberikan arah pada sistem cerdas untuk mencapai tujuan yang diinginkan. Algoritma dan teknik kecerdasan buatan digunakan untuk mencari dan merencanakan serangkaian tindakan atau langkah yang diperlukan untuk mencapai goal state dari initial state yang diberikan.

B. Sistem produksi

Sistem produksi (production system) adalah salah satu paradigma dalam kecerdasan buatan yang digunakan untuk memodelkan dan mengimplementasikan sistem cerdas. Sistem produksi terdiri dari aturan-aturan produksi yang terdiri dari kondisi (if) dan tindakan (then). Aturan produksi ini memungkinkan sistem cerdas untuk menganalisis input, membandingkannya dengan kondisi yang ditentukan, dan melakukan tindakan yang sesuai berdasarkan kondisi tersebut.

Berikut adalah komponen-komponen utama dari sistem produksi dalam kecerdasan buatan:

1. Basis Pengetahuan (Knowledge Base): Merupakan kumpulan pengetahuan yang dimiliki oleh sistem cerdas. Basis pengetahuan ini terdiri dari fakta-fakta, aturan-aturan produksi, dan informasi lain yang relevan dengan masalah yang ingin diselesaikan.
2. Mesin Inferensi (Inference Engine): Merupakan komponen yang bertanggung jawab untuk menerapkan aturan-aturan produksi pada basis pengetahuan. Mesin inferensi menganalisis input yang diberikan, membandingkannya dengan kondisi yang ditentukan dalam aturan produksi, dan menghasilkan tindakan yang sesuai berdasarkan kondisi tersebut.
3. Basis Fakta (Fact Base): Merupakan kumpulan fakta atau informasi tentang keadaan saat ini yang digunakan oleh mesin inferensi. Basis fakta berfungsi sebagai input untuk proses inferensi dan terus diperbarui selama sistem cerdas berjalan.
4. Antarmuka Pengguna (User Interface): Merupakan antarmuka yang memungkinkan pengguna untuk berinteraksi dengan sistem cerdas, memberikan input, dan menerima output atau rekomendasi dari sistem tersebut.

Proses kerja sistem produksi dapat dijelaskan sebagai berikut:

1. Input diberikan ke sistem produksi, misalnya dalam bentuk fakta atau pernyataan.
2. Mesin inferensi menganalisis input dan mencocokkan dengan aturan-aturan produksi dalam basis pengetahuan.

3. Jika kondisi dalam aturan produksi terpenuhi, maka tindakan yang sesuai akan dijalankan.
4. Output atau rekomendasi diberikan berdasarkan tindakan yang dihasilkan.
5. Proses berulang terus menerus, di mana input baru akan dianalisis dan diolah berdasarkan aturan produksi yang relevan.

Sistem produksi dalam kecerdasan buatan sangat fleksibel dan dapat digunakan dalam berbagai aplikasi, termasuk sistem rekomendasi, diagnosis medis, kontrol proses, dan banyak lagi. Kelebihan utama dari sistem produksi adalah kemampuannya untuk menerapkan pengetahuan berbasis aturan secara efisien dan mudah dipahami.

Analisa gambaran aturan produksi dari masing-masing ruang masalah dalam kecerdasan buatan akan bervariasi tergantung pada kasus yang spesifik. Berikut adalah contoh umum aturan produksi untuk beberapa ruang masalah kecerdasan buatan:

1. Pencarian Jalan Terpendek:

Aturan Produksi:

- IF saat ini berada di simpul X dan ada simpul Y yang terhubung langsung dengan X THEN pindah ke simpul Y.

2. Permainan Catur:

Aturan Produksi:

- IF dalam kondisi tertentu, ada serangan yang menguntungkan pada lawan THEN lakukan serangan tersebut.
- IF raja dalam kondisi terancam THEN pindahkan raja ke posisi yang lebih aman.
- IF memenuhi kondisi tertentu, ada kesempatan untuk melakukan jalan raja THEN lakukan jalan raja.

3. Penerjemahan Mesin:

Aturan Produksi:

- IF menemukan pola kata yang sesuai dalam bahasa asal THEN terjemahkan kata tersebut ke bahasa target.
- IF menemukan pola kalimat khusus dalam bahasa asal THEN terjemahkan kalimat tersebut ke bahasa target sesuai dengan pola yang telah ditentukan.

4. Diagnosis Medis:

Aturan Produksi:

- IF gejala A, B, dan C terjadi THEN duga adanya penyakit X.
- IF gejala Y dan Z terjadi THEN duga adanya penyakit Y.
- IF gejala A dan B terjadi tetapi gejala C tidak terjadi THEN duga adanya penyakit Z.

5. Sistem Rekomendasi:

Aturan Produksi:

- IF pengguna telah menonton film A dan B THEN rekomendasikan film C yang serupa dengan A dan B.
- IF pengguna memberikan peringkat tinggi untuk genre film X THEN rekomendasikan film terbaru dari genre X.

Dalam setiap ruang masalah, aturan produksi digunakan untuk menghubungkan kondisi-kondisi tertentu dengan tindakan atau rekomendasi yang harus diambil. Aturan-aturan ini dapat diimplementasikan dalam bentuk "IF kondisi THEN tindakan". Tujuan utama dari aturan produksi adalah memberikan panduan kepada sistem cerdas dalam pengambilan keputusan atau memberikan solusi yang sesuai dengan kondisi yang ada.

BAB III & IV

METODE - METODE PENCARIAN DALAM KECERDASAN BUATAN

A. Metode Pencarian Buta

Metode pencarian buta tidak mempertimbangkan informasi tambahan tentang masalah yang sedang diselesaikan, melainkan hanya bergantung pada struktur ruang pencarian. Berikut beberapa metode pencarian buta yang umum digunakan:

- a. Pencarian Breadth-First (Pencarian Lebar):
Metode ini melakukan pencarian secara melebar pada semua simpul yang ada pada tingkat yang sama sebelum melanjutkan ke tingkat berikutnya. Dalam metode ini, simpul-simpul anak dieksplorasi terlebih dahulu sebelum simpul-simpul cucu.
- b. Pencarian Depth-First (Pencarian Dalam):
Metode ini melakukan pencarian secara mendalam pada jalur tertentu sampai tidak ada simpul yang tersisa. Pencarian ini dilakukan dengan memilih simpul anak pertama dan melanjutkan hingga mencapai simpul terminal atau simpul tanpa anak.
- c. Pencarian Iterative Deepening (Pencarian Dalam Terbatas):
Metode ini adalah variasi dari pencarian dalam (depth-first search), tetapi dengan batasan kedalaman yang ditingkatkan pada setiap iterasi. Metode ini berguna saat ruang pencarian sangat besar dan kita tidak memiliki informasi tentang kedalaman solusi yang diharapkan.

B. Metode Pencarian Heuristik

Metode pencarian heuristik menggunakan informasi tambahan yang disebut heuristik untuk membimbing proses pencarian. Heuristik adalah aturan praktis atau strategi yang dirancang untuk mempercepat proses pencarian. Beberapa metode pencarian heuristik yang umum digunakan adalah sebagai berikut:

- a. Pencarian Hill Climbing (Pencarian Naik Bukit):
Metode ini mencoba untuk mencari solusi dengan melakukan gerakan yang mengarah ke peningkatan nilai fungsi heuristik. Namun, metode ini dapat terjebak pada optimum lokal jika tidak ada jalan keluar yang lebih baik.
- b. Pencarian Best-First (Pencarian Terbaik Pertama):
Metode ini memilih simpul yang paling menjanjikan berdasarkan fungsi evaluasi atau heuristik tertentu. Simpul-simpul yang memiliki nilai heuristik terbaik dieksplorasi terlebih dahulu.
- c. Pencarian A*:

Metode ini menggabungkan konsep pencarian terbaik pertama (best-first search) dengan fungsi evaluasi heuristik. Fungsi evaluasi A* adalah kombinasi dari biaya sejauh ini dan perkiraan biaya yang tersisa untuk mencapai tujuan. Metode ini memastikan solusi optimal jika fungsi heuristik memenuhi sifat-sifat tertentu.

Metode pencarian heuristik cenderung lebih efisien daripada metode pencarian buta karena penggunaan informasi tambahan. Namun, metode heuristik juga dapat memberikan solusi suboptimal jika heuristiknya tidak memadai atau jika terjebak pada optimum lokal. Pilihan metode pencarian tergantung pada sifat masalah dan ketersediaan informasi.

BAB V

REASONING : PROPOSITIONAL LOGIC

A. PROPOSITIONAL LOGIC

Proposisi logika, juga dikenal sebagai logika sentensial atau kalkulus proposisi, adalah cabang logika formal yang berurusan dengan proposisi, yaitu pernyataan yang dapat benar atau salah. Dalam proposisi logika, proposisi ini direpresentasikan menggunakan simbol dan operator untuk membentuk ekspresi logika. Fokus utama dari proposisi logika adalah menganalisis hubungan antara ekspresi logika ini dan menentukan nilai kebenaran mereka berdasarkan aturan dan prinsip tertentu.

Berikut adalah beberapa komponen dan konsep utama dalam proposisi logika:

- a. **Proposisi:** Proposisi adalah blok dasar dari proposisi logika. Mereka mewakili pernyataan yang dapat benar atau salah. Misalnya, "Sedang hujan" atau " $2 + 2 = 5$ " dapat dianggap sebagai proposisi.
- b. **Operator Logika:** Operator logika digunakan untuk menggabungkan atau memanipulasi proposisi. Tiga operator logika utama dalam proposisi logika adalah:
 - **Negasi (\neg):** Digunakan untuk mengungkapkan penyangkalan dari sebuah proposisi. Misalnya, jika p mewakili proposisi "Sedang hujan," maka $\neg p$ mewakili "Tidak sedang hujan."
 - **Konjungsi (\wedge):** Mewakili operasi AND logika. Operator ini benar hanya jika kedua proposisi yang digabungkan benar. Misalnya, jika p mewakili "Sedang hujan" dan q mewakili "Tanah basah," maka $p \wedge q$ mewakili "Sedang hujan dan tanah basah."
 - **Disjungsi (\vee):** Mewakili operasi OR logika. Operator ini benar jika setidaknya satu dari proposisi yang digabungkan adalah benar. Misalnya, jika p mewakili "Sedang hujan" dan q mewakili "Angin kencang," maka $p \vee q$ mewakili "Sedang hujan atau angin kencang."
- c. **Implikasi (\rightarrow):** Implikasi merupakan hubungan logika yang menyatakan bahwa jika suatu proposisi benar, maka proposisi lain juga benar. Misalnya, jika p mewakili "Jika hujan, maka tanah basah," dan q mewakili "Hujan," maka $p \rightarrow q$ menyatakan "Jika hujan, maka tanah basah."

- d. Ekuivalensi (\leftrightarrow): Ekuivalensi merupakan hubungan logika yang menyatakan bahwa dua proposisi memiliki nilai kebenaran yang sama. Misalnya, jika p mewakili "Hujan dan tanah basah" dan q mewakili "Tanah basah jika dan hanya jika hujan," maka $p \leftrightarrow q$ menyatakan "Hujan dan tanah basah jika dan hanya jika tanah basah jika dan hanya jika hujan."

Proposisi logika merupakan alat penting dalam penyelidikan, pemodelan, dan analisis masalah dalam kecerdasan buatan dan logika komputasional.

BAB VI & VII

REASONING : FIRST ORDER LOGIC

(PREDICATE CALCULUS)

Logika orde pertama adalah sistem formal yang memungkinkan kita untuk beralasan tentang pernyataan yang melibatkan objek, properti, dan relasi. Ini adalah alat yang kuat yang digunakan dalam matematika, ilmu komputer, filsafat, dan disiplin lainnya untuk mengungkapkan dan menganalisis pernyataan logika yang kompleks.

Dalam logika orde pertama, kita memiliki beberapa komponen kunci:

1. Simbol: Logika orde pertama menggunakan simbol untuk mewakili berbagai elemen. Simbol-simbol ini termasuk variabel (misalnya, x , y , z), konstanta (misalnya, a , b , c), predikat (misalnya, P , Q), dan fungsi (misalnya, f , g).
2. Istilah: Istilah adalah blok dasar dalam logika orde pertama. Mereka bisa berupa variabel, konstanta, atau fungsi yang diterapkan pada istilah. Misalnya, " x " dan " a " adalah istilah, dan " $f(x)$ " adalah istilah jika " f " adalah simbol fungsi.
3. Formula: Formula dalam logika orde pertama dibangun menggunakan istilah, predikat, dan konektor logika. Predikat mewakili properti atau hubungan antara objek, dan dapat mengambil istilah sebagai argumen. Konektor logika meliputi konjungsi (\wedge), disjungsi (\vee), implikasi (\rightarrow), negasi (\neg), dan kuantifikasi (\forall dan \exists).
4. Kuantifier: Logika orde pertama menggunakan kuantifier untuk mengungkapkan pernyataan tentang semua atau beberapa objek. Kuantifier universal (\forall) menyatakan bahwa suatu pernyataan berlaku untuk semua objek dalam suatu domain, sedangkan kuantifier eksistensial (\exists) menyatakan bahwa setidaknya ada satu objek untuk yang pernyataan tersebut benar.
6. Aksioma dan Aturan Penalaran: Aksioma adalah pernyataan dasar yang diasumsikan benar, dan aturan penalaran memungkinkan kita untuk menghasilkan pernyataan baru yang valid dari pernyataan yang sudah ada.

Semantik: Logika orde pertama memiliki semantik yang terdefinisi dengan baik yang memberikan makna pada simbol-simbol dan menentukan nilai kebenaran dari formula dalam berbagai interpretasi atau model.

Dengan menggunakan logika orde pertama, kita dapat mengungkapkan berbagai pernyataan logika, merumuskan hubungan antara objek, dan membuat deduksi berdasarkan aksioma dan aturan yang diberikan. Ini menyediakan dasar untuk penalaran formal dan menjadi dasar bagi banyak sistem penalaran otomatis, seperti pembuktian teorema dan bahasa pemrograman logika.

Perlu dicatat bahwa logika orde pertama lebih ekspresif daripada logika proposisional, karena memungkinkan kita untuk merumuskan pernyataan tentang objek individu dan properti mereka, sedangkan log

BAB IX

FUZZY LOGIC - FUZZINESS & PROBABILITAS

A. fuzziness dan probabilitas

Fuzziness adalah sebuah konsep dalam teori ketidakpastian yang mengacu pada keadaan di mana suatu objek atau konsep tidak memiliki batas atau definisi yang tegas. Dalam konteks ini, ketidakpastian atau ketidakjelasan dapat terjadi ketika objek atau konsep tersebut memiliki sifat-sifat atau atribut yang tidak dapat diukur dengan tepat atau dijelaskan dengan tegas.

Dalam matematika, fuzziness atau ketidakpastian sering kali dinyatakan dengan menggunakan logika fuzzy. Logika fuzzy memungkinkan representasi variabel yang memiliki nilai sebagian atau secara parsial di antara kategori atau nilai-nilai yang dapat diterima. Ini berbeda dengan logika klasik yang menggunakan kategorisasi biner, di mana suatu variabel hanya dapat memiliki nilai yang benar atau salah.

Sementara itu, probabilitas adalah sebuah ukuran numerik yang menggambarkan tingkat kepercayaan atau keyakinan terhadap terjadinya suatu peristiwa. Probabilitas dapat dianggap sebagai jumlah atau peluang peristiwa tersebut terjadi dalam konteks ruang sampel atau himpunan semua hasil yang mungkin.

Probabilitas dapat dinyatakan dalam skala dari 0 hingga 1, di mana 0 berarti peristiwa tersebut tidak mungkin terjadi, 1 berarti peristiwa tersebut pasti terjadi, dan angka di antara 0 dan 1 mencerminkan tingkat keyakinan terhadap terjadinya peristiwa tersebut. Semakin tinggi probabilitas, semakin besar kemungkinan peristiwa tersebut terjadi.

Dalam beberapa kasus, probabilitas juga dapat dinyatakan dalam bentuk logika fuzzy untuk mengakomodasi ketidakpastian atau ketidakjelasan yang mungkin terkait dengan suatu peristiwa. Dalam hal ini, probabilitas menjadi fungsi fuzzy yang menggambarkan tingkat keanggotaan peristiwa tersebut dalam himpunan nilai yang mungkin.

B. Fuzzy set

Teori himpunan fuzzy adalah kerangka matematika yang mengatasi ketidakpastian dan kekaburan dalam data dan informasi. Teori ini diperkenalkan oleh Lotfi Zadeh pada tahun 1965 sebagai perluasan dari teori himpunan klasik. Himpunan fuzzy digunakan untuk mewakili dan memanipulasi informasi yang samar atau tidak pasti dalam berbagai bidang, termasuk matematika, ilmu komputer, kecerdasan buatan, dan rekayasa.

Konsep utama dalam teori himpunan fuzzy adalah konsep fungsi keanggotaan, yang memberikan derajat keanggotaan untuk setiap elemen dalam himpunan. Berbeda dengan himpunan klasik di mana suatu elemen hanya bisa menjadi anggota atau bukan, himpunan fuzzy memungkinkan adanya keanggotaan parsial. Fungsi keanggotaan memetakan elemen-elemen himpunan ke nilai-nilai dalam interval $[0, 1]$, di mana 0 menunjukkan tidak ada keanggotaan dan 1 menunjukkan keanggotaan penuh.

Berikut adalah beberapa aspek dan konsep utama dalam teori himpunan fuzzy:

Fungsi Keanggotaan: Fungsi keanggotaan menggambarkan derajat keanggotaan setiap elemen dalam himpunan fuzzy. Fungsi ini bisa memiliki berbagai bentuk, seperti segitiga, trapesium, Gaussian, atau bentuk lain yang sesuai dengan tingkat kekaburan yang diinginkan.

Dukungan: Dukungan dari himpunan fuzzy adalah himpunan semua elemen yang memiliki derajat keanggotaan yang tidak nol. Ini mewakili sejauh mana himpunan fuzzy didefinisikan.

Alpha-Cut: Alpha-cut dari himpunan fuzzy adalah himpunan tegas yang diperoleh dengan memilih elemen-elemen dengan derajat keanggotaan yang lebih besar atau sama dengan nilai ambang tertentu (α). Ini mewakili tingkat kepercayaan atau kepastian dalam keanggotaan elemen-elemen tersebut.

Operasi pada Himpunan Fuzzy: Himpunan fuzzy dapat digabungkan dan dimanipulasi menggunakan operasi himpunan seperti gabungan, irisan, dan komplement. Operasi-operasi ini didefinisikan berdasarkan operasi-operasi yang sesuai dalam teori himpunan klasik, namun mempertimbangkan derajat keanggotaan.

Logika Fuzzy: Teori himpunan fuzzy membentuk dasar bagi logika fuzzy, yang merupakan logika dengan banyak nilai yang memungkinkan penalaran dan pengambilan keputusan dalam kondisi ketidakpastian. Logika fuzzy memperluas logika biner klasik dengan memperbolehkan adanya derajat kebenaran antara benar dan salah.

Sistem Inferensi Fuzzy: Himpunan fuzzy dan logika fuzzy umumnya digunakan dalam desain sistem inferensi fuzzy (FIS). FIS adalah sistem berbasis aturan yang menggunakan logika fuzzy untuk memodelkan dan mengendalikan sistem kompleks dengan data masukan yang tidak pasti atau tidak akurat.

Teori himpunan fuzzy menyediakan alat yang kuat untuk menangani ketidakpastian, ketidak

C. Fuzzy Logic

Fuzzy logic adalah kerangka matematika yang digunakan untuk mengatasi ketidakpastian dan ketidaktepatan dalam pengambilan keputusan dan pemecahan masalah. Berbeda dengan logika tradisional yang mengandalkan nilai benar/salah biner, fuzzy logic memungkinkan adanya tingkat kebenaran yang intermediat, yang direpresentasikan oleh variabel linguistik seperti "sangat benar" atau "agak salah".

Konsep sentral dalam fuzzy logic adalah fungsi keanggotaan, yang mendefinisikan tingkat keanggotaan elemen dalam himpunan tertentu. Fungsi keanggotaan ini dapat mengambil nilai antara 0 dan 1, yang menunjukkan tingkat keanggotaan. Fuzzy logic beroperasi pada himpunan fuzzy, yang didefinisikan oleh fungsi keanggotaan ini.

Fuzzy logic menyediakan cara untuk memodelkan dan merumuskan informasi yang samar dan tidak pasti. Ini sangat berguna dalam situasi di mana nilai numerik yang tepat sulit didapatkan atau di mana penalaran dan pertimbangan manusia terlibat. Beberapa aplikasi utama fuzzy logic meliputi:

1.Sistem Kontrol: Fuzzy logic umum digunakan dalam sistem kontrol untuk menangani lingkungan yang kompleks dan tidak pasti. Kontroler fuzzy dapat menangkap pengetahuan ahli dan memberikan kontrol yang kuat dalam situasi di mana metode kontrol tradisional mungkin tidak mencukupi.

2.Pengambilan Keputusan: Fuzzy logic dapat digunakan dalam proses pengambilan keputusan untuk mengakomodasi informasi yang tidak tepat atau tidak lengkap. Ini memungkinkan representasi faktor-faktor subjektif atau kualitatif yang sulit diukur secara tepat.

3.Pengenalan Pola: Teknik-teknik fuzzy logic diterapkan dalam tugas pengenalan pola, seperti pengenalan gambar dan ucapan. Sistem fuzzy dapat mengatasi variasi dan ketidakpastian dalam data, sehingga cocok untuk menangani pola yang berisik atau ambigu.

4.Sistem Pakar: Fuzzy logic digunakan dalam sistem pakar untuk meniru penalaran dan pengambilan keputusan manusia. Dengan menggabungkan aturan fuzzy dan mekanisme inferensi fuzzy, sistem-sistem ini dapat menangani pengetahuan yang tidak tepat dan tidak pasti.

5.Pemrosesan Bahasa Alami: Fuzzy logic digunakan dalam tugas pemrosesan bahasa alami, seperti pengambilan informasi dan klasifikasi teks. Ini membantu dalam menangani kesamaran dan ambiguitas inheren dari bahasa manusia.

Penting untuk dicatat bahwa fuzzy logic bukan pengganti logika klasik, tetapi alat tambahan yang melengkapi logika tradisional untuk mengatasi informasi yang tidak pasti dan samar dengan lebih efektif.

D. Fuzzy Systems

Sistem fuzzy, juga dikenal sebagai sistem logika fuzzy, adalah model komputasi yang dirancang untuk menangani ketidakpastian dan informasi yang tidak tepat. Mereka berdasarkan teori himpunan fuzzy, yang dikembangkan oleh Lotfi A. Zadeh pada tahun 1960-an sebagai cara untuk mewakili dan memanipulasi konsep yang samar atau ambigu.

Sistem fuzzy sangat berguna dalam situasi di mana sistem logika biner tradisional kesulitan menangani ketidakpastian atau ketika model matematika

yang tepat sulit untuk dirumuskan. Mereka telah diterapkan dalam berbagai bidang, termasuk sistem kontrol, pengambilan keputusan, pengenalan pola, dan kecerdasan buatan.

Komponen utama dari sistem fuzzy adalah sebagai berikut:

1. Himpunan Fuzzy: Himpunan fuzzy memgeneralisasikan konsep himpunan klasik dengan memungkinkan elemen memiliki tingkat keanggotaan yang bervariasi. Alih-alih nilai keanggotaan yang ketat 0 atau 1, elemen dapat memiliki tingkat keanggotaan antara 0 dan 1, yang menunjukkan tingkat kepemilikan mereka terhadap himpunan.

2. Fungsi Keanggotaan: Fungsi keanggotaan mendefinisikan tingkat keanggotaan untuk setiap elemen dalam himpunan fuzzy. Mereka memetakan elemen ke nilai keanggotaan mereka, yang mewakili tingkat kebenaran atau kepuasan dari kondisi tertentu. Fungsi keanggotaan dapat memiliki berbagai bentuk, seperti segitiga, trapesium, Gaussian, atau sigmoidal, tergantung pada bentuk himpunan fuzzy.

3. Operasi Logika Fuzzy: Operasi logika fuzzy digunakan untuk menggabungkan himpunan fuzzy dan memanipulasi nilai keanggotaan mereka. Operasi logika fuzzy dasar meliputi gabungan, irisan, dan komplemen. Operasi ini memungkinkan penalaran dan inferensi fuzzy.

4. Basis Aturan: Basis aturan berisi kumpulan aturan fuzzy jika-maka yang menentukan perilaku sistem fuzzy. Setiap aturan terdiri dari bagian hipotesis (bagian jika) dan bagian konklusi (bagian maka). Hipotesis menentukan kondisi berdasarkan himpunan fuzzy dan fungsi keanggotaan, sementara konklusi mendefinisikan tindakan atau keluaran berdasarkan kondisi tersebut.

5. Inferensi Fuzzy: Inferensi fuzzy adalah proses menentukan tingkat kebenaran atau ketidakbenaran konklusi berdasarkan tingkat kebenaran hipotesis. Ini melibatkan penerapan operasi logika fuzzy dan menggabungkan aturan untuk menghasilkan keluaran atau keputusan yang jelas.

6. Defuzzifikasi: Defuzzifikasi adalah langkah terakhir dalam sistem fuzzy, di mana keluaran fuzzy dikonversi menjadi nilai yang jelas atau keputusan yang jelas. Berbagai metode defuzzifikasi dapat digunakan, seperti centroid, rata-rata terbobot, atau keanggotaan maksimum.

Sistem fuzzy menawarkan kerangka kerja yang fleksibel dan intuitif untuk menangani ketidakpastian dan ketidaktepatan

BAB X

Planning (teknik dekomposisi masalah) - Goal Stack Planning (GSP) - Constraint Posting (CP)

A. Goal Stack Planning (GSP)

Goal Stack Planning (GSP) adalah salah satu metode perencanaan yang digunakan dalam kecerdasan buatan (artificial intelligence) untuk menghasilkan rencana tindakan yang terstruktur dalam mencapai tujuan tertentu. GSP melibatkan representasi tumpukan (stack) tujuan yang harus dicapai dan aturan-aturan yang menggambarkan bagaimana tujuan-tujuan tersebut dapat dicapai.

Dalam GSP, tujuan-tujuan direpresentasikan sebagai tumpukan, di mana tujuan utama ditempatkan di bagian atas tumpukan dan tujuan-tujuan yang lebih spesifik ditempatkan di bawahnya. Setiap tujuan dalam tumpukan dapat memiliki tindakan yang terkait dengannya untuk mencapai tujuan tersebut. Tindakan-tindakan ini juga dapat memicu tujuan-tujuan lain yang harus dicapai sebelum mencapai tujuan awal.

Proses perencanaan dalam GSP melibatkan pemecahan tumpukan tujuan menjadi tumpukan tindakan yang diperlukan untuk mencapai setiap tujuan. Perencanaan dimulai dengan tujuan utama di atas tumpukan dan kemudian melibatkan dekomposisi tujuan menjadi subtujuan yang lebih kecil. Pemecahan tumpukan tujuan terus dilakukan hingga semua tujuan telah terdekomposisi menjadi tindakan-tindakan konkret yang dapat dilakukan.

Salah satu keuntungan dari GSP adalah kemampuannya untuk menghasilkan rencana tindakan yang terstruktur secara hierarkis, di mana tujuan-tujuan yang lebih tinggi dalam tumpukan menyediakan panduan umum untuk mencapai tujuan-tujuan yang lebih spesifik. GSP juga memungkinkan fleksibilitas dalam mengubah atau menyesuaikan rencana jika ada perubahan lingkungan atau kondisi yang mempengaruhi tujuan-tujuan.

GSP telah digunakan dalam berbagai aplikasi kecerdasan buatan, termasuk perencanaan tindakan dalam robotika, sistem pemrograman otomatis, dan sistem kontrol yang kompleks.

B. Constraint Posting (CP)

Constraint Posting (CP) adalah pendekatan dalam masalah optimasi yang digunakan untuk memodelkan batasan dan memecahkan masalah dengan cara

mencari solusi yang memenuhi semua batasan tersebut. CP menggunakan konsep pemrograman kendala (constraint programming) untuk memodelkan masalah sebagai kumpulan variabel, domain variabel, dan batasan.

Dalam CP, variabel-variabel digunakan untuk mewakili nilai-nilai yang harus ditentukan dalam masalah. Setiap variabel memiliki domain, yaitu kumpulan nilai yang dapat diambil oleh variabel tersebut. Batasan-batasan diterapkan pada variabel-variabel ini untuk membatasi nilai-nilai yang mungkin diambil oleh variabel dan mengatur hubungan antara variabel-variabel tersebut.

Misalnya, jika kita memiliki variabel A dan B, dan kita ingin memastikan bahwa nilai A harus lebih kecil dari B, kita dapat menetapkan batasan $A < B$. Dalam CP, batasan ini akan digunakan untuk mengeliminasi kombinasi nilai-nilai yang tidak memenuhi batasan tersebut.

Salah satu keunggulan CP adalah kemampuannya untuk menangani masalah yang kompleks dengan batasan yang beragam dan rumit. Dalam CP, masalah dapat dipecahkan secara deklaratif, artinya kita hanya perlu menyatakan batasan-batasan yang harus dipenuhi, dan sistem CP akan mencari solusi yang memenuhi semua batasan tersebut.

CP telah digunakan dalam berbagai bidang, termasuk perencanaan jadwal, optimasi produksi, pengaturan tugas, dan banyak lagi. Pendekatan CP memungkinkan model yang fleksibel dan ekspresif untuk masalah optimasi dengan memanfaatkan konsep batasan dan pemrograman kendala.

BAB XI

Decision Tree Learning

A. ID3

Learning ID3 (Iterative Dichotomiser 3) adalah sebuah algoritma pembelajaran mesin yang digunakan untuk membangun pohon keputusan (decision tree) dari data yang diberikan. Algoritma ini merupakan salah satu metode pembelajaran yang populer dalam klasifikasi data.

Pohon keputusan adalah struktur hierarkis yang menggambarkan serangkaian keputusan yang harus diambil berdasarkan fitur-fitur data. Setiap simpul dalam pohon keputusan mewakili keputusan atau pemisahan data berdasarkan atribut-atribut tertentu. Setiap cabang dari simpul tersebut mewakili nilai-nilai yang mungkin dari atribut tersebut, sedangkan daun-daun di pohon keputusan mewakili kelas atau label yang akan diberikan kepada data.

Berikut adalah langkah-langkah utama dalam algoritma pembelajaran ID3:

1. Memilih atribut yang paling informatif: ID3 menggunakan metode pengukuran informasi seperti gain informasi atau rasio gain untuk memilih atribut yang paling baik dalam memisahkan data.
2. Membuat simpul dalam pohon keputusan: Setelah atribut terbaik dipilih, simpul baru dibuat dalam pohon keputusan berdasarkan atribut tersebut. Setiap cabang dari simpul mewakili nilai-nilai yang mungkin dari atribut.
3. Membagi data berdasarkan nilai atribut: Data pelatihan dibagi menjadi subset-subset yang berbeda berdasarkan nilai-nilai atribut yang mungkin. Setiap subset kemudian diproses secara rekursif menggunakan langkah-langkah ID3 untuk membangun cabang-cabang pohon keputusan yang lebih lanjut.
4. Mengulangi langkah-langkah di atas: Langkah-langkah 1 hingga 3 diulangi secara iteratif untuk setiap cabang atau simpul baru yang dibuat hingga semua data pelatihan terbagi dengan sempurna atau sampai kondisi penghentian tertentu terpenuhi.
5. Menentukan label atau kelas di daun: Setelah pohon keputusan dibangun, label atau kelas ditentukan untuk setiap daun dalam pohon berdasarkan mayoritas kelas data pelatihan yang terkait.

Algoritma ID3 memiliki kecenderungan untuk memilih atribut dengan banyak nilai dan rentang yang luas, serta rentan terhadap overfitting jika tidak diatur dengan baik. Terdapat juga variasi algoritma seperti C4.5 dan CART yang merupakan perluasan dari ID3 dengan perbaikan dan penyesuaian tambahan.

ID3 adalah salah satu algoritma pembelajaran mesin yang sederhana dan mudah dipahami, namun terbatas pada masalah klasifikasi dan tidak dapat menangani atribut kontinu secara langsung.

B. C4.5

C4.5 adalah algoritma pembelajaran mesin yang digunakan untuk membangun pohon keputusan (decision tree) dari data pelatihan. Tujuannya adalah untuk melakukan klasifikasi data atau memprediksi nilai target berdasarkan fitur-fitur yang ada dalam data.

Berikut adalah langkah-langkah utama dalam algoritma pembelajaran C4.5:

1. Memilih atribut terbaik: C4.5 menggunakan metode pengukuran gain rasio untuk memilih atribut yang paling baik dalam memisahkan data. Gain rasio menggabungkan pengukuran informasi (seperti yang digunakan dalam ID3) dengan penyeimbangan ukuran atribut.
2. Membuat simpul dalam pohon keputusan: Setelah atribut terbaik dipilih, simpul baru dibuat dalam pohon keputusan berdasarkan atribut tersebut. Setiap cabang dari simpul mewakili nilai-nilai yang mungkin dari atribut.
3. Membagi data berdasarkan nilai atribut: Data pelatihan dibagi menjadi subset-subset yang berbeda berdasarkan nilai-nilai atribut yang mungkin. Setiap subset kemudian diproses secara rekursif menggunakan langkah-langkah C4.5 untuk membangun cabang-cabang pohon keputusan yang lebih lanjut.
4. Mengulangi langkah-langkah di atas: Langkah-langkah 1 hingga 3 diulangi secara iteratif untuk setiap cabang atau simpul baru yang dibuat hingga semua data pelatihan terbagi dengan sempurna atau sampai kondisi penghentian tertentu terpenuhi.
5. Menentukan label atau kelas di daun: Setelah pohon keputusan dibangun, label atau kelas ditentukan untuk setiap daun dalam pohon berdasarkan mayoritas kelas data pelatihan yang terkait.

Selain membangun pohon keputusan, C4.5 juga dapat mengatasi atribut kontinu dengan melakukan diskritisasi atau pengelompokan nilai kontinu menjadi beberapa interval. Selain itu, C4.5 dapat menangani data yang hilang (missing data) dengan mengganti nilai yang hilang berdasarkan statistik atau menggunakan probabilitas distribusi atribut yang ada.

C4.5 adalah salah satu algoritma pembelajaran pohon keputusan yang lebih canggih dan fleksibel dibandingkan dengan ID3, dan telah digunakan secara luas dalam berbagai aplikasi klasifikasi dan prediksi.

BAB XII

JARINGAN SARAF TIRUAN

A. konsep dasar JST

Konsep dasar JST (Jaringan Syaraf Tiruan) merujuk pada prinsip-prinsip dasar yang mendasari struktur dan fungsi jaringan syaraf tiruan. Jaringan syaraf tiruan adalah model matematika yang terinspirasi oleh cara kerja otak manusia dan digunakan dalam bidang kecerdasan buatan untuk memodelkan dan menyelesaikan masalah yang kompleks.

Berikut ini adalah beberapa konsep dasar dalam JST:

1. **Neuron:** Neuron adalah unit dasar dalam jaringan syaraf tiruan. Ini meniru neuron biologis dengan menerima masukan (dalam bentuk sinyal atau bobot) dari neuron lain, memprosesnya, dan menghasilkan keluaran. Neuron juga memiliki fungsi aktivasi yang dapat mengubah output berdasarkan batasan tertentu.
2. **Bobot:** Bobot adalah nilai numerik yang diasosiasikan dengan setiap koneksi antara neuron dalam jaringan. Bobot menunjukkan kekuatan koneksi atau pengaruh yang dimiliki oleh neuron yang satu terhadap neuron yang lain. Bobot-bobot ini dapat disesuaikan atau diperbarui selama proses pembelajaran jaringan untuk meningkatkan kemampuan jaringan dalam mempelajari pola dan menghasilkan keluaran yang diinginkan.
3. **Fungsi Aktivasi:** Fungsi aktivasi adalah fungsi matematis yang diterapkan pada keluaran neuron untuk memperoleh keluaran yang diinginkan. Fungsi ini dapat beragam, seperti fungsi sigmoid, fungsi tangen hiperbolik, atau fungsi ReLU (Rectified Linear Unit). Fungsi aktivasi ini mengintroduksi sifat non-linearitas ke dalam jaringan, yang memungkinkan jaringan untuk mempelajari hubungan non-linear dalam data.
4. **Feedforward:** Feedforward adalah proses dimana sinyal masukan mengalir melalui jaringan dari lapisan masukan ke lapisan keluaran tanpa ada umpan balik atau koneksi siklus. Dalam feedforward, setiap neuron di lapisan tertentu hanya terhubung dengan neuron di lapisan berikutnya.
5. **Pembelajaran:** Pembelajaran dalam JST melibatkan kemampuan jaringan untuk memperbarui bobot-bobotnya berdasarkan kesalahan antara keluaran yang dihasilkan oleh jaringan dan keluaran yang diharapkan. Proses pembelajaran dapat menggunakan berbagai algoritma seperti backpropagation, pembelajaran terbimbing (supervised learning), atau pembelajaran tanpa pengawasan (unsupervised learning) untuk

mengoptimalkan bobot-bobot sehingga jaringan dapat menghasilkan keluaran yang akurat.

6. Lapisan: Jaringan syaraf tiruan sering terdiri dari beberapa lapisan neuron yang terhubung satu sama lain. Lapisan-lapisan ini biasanya terdiri dari lapisan masukan (input layer), lapisan tersembunyi (hidden layer), dan lapisan keluaran (output layer). Lapisan tersembunyi berperan dalam pemrosesan dan ekstraksi fitur dari masukan, sedangkan lapisan keluaran menghasilkan output akhir dari jaringan.

Konsep dasar ini membentuk dasar untuk memahami jaringan syaraf tiruan dan bagaimana jaringan tersebut dapat memodelkan dan memproses informasi secara paralel untuk memecahkan masalah yang kompleks. Dengan memahami konsep-konsep ini, kita dapat memanfaatkan JST untuk berbagai aplikasi seperti pengenalan pola, klasifikasi, prediksi, dan banyak lagi.

B. model syaraf tiruan (Neuron)

Model syaraf tiruan (neuron) adalah representasi matematika yang digunakan untuk meniru unit dasar pengolahan informasi dalam jaringan syaraf tiruan. Neuron dalam konteks ini terinspirasi oleh neuron biologis yang ada dalam sistem saraf manusia.

Model syaraf tiruan terdiri dari beberapa komponen penting, yang meliputi:

1. Masukan (Input): Masukan dalam model syaraf tiruan adalah sinyal atau informasi yang diterima oleh neuron. Sinyal ini dapat berasal dari neuron-neuron lain dalam jaringan atau dari lingkungan eksternal.
2. Bobot (Weight): Setiap masukan ke neuron diberi bobot numerik yang menggambarkan kekuatan koneksi atau pengaruh masukan tersebut terhadap keluaran neuron. Bobot ini dapat dianggap sebagai faktor penting yang mempengaruhi kontribusi masukan terhadap aktivitas neuron.
3. Fungsi Agregasi: Fungsi agregasi (summing function) digunakan untuk menggabungkan masukan dan bobot mereka. Ini bisa berupa penjumlahan sederhana atau operasi lain seperti perkalian atau perkalian titik dalam kasus jaringan syaraf tiruan dengan struktur yang lebih kompleks.
4. Fungsi Aktivasi: Fungsi aktivasi diterapkan pada hasil dari fungsi agregasi untuk menghasilkan output dari neuron. Fungsi ini mengenalkan non-linearitas pada respons neuron, yang memungkinkan neuron untuk memproses informasi yang lebih kompleks daripada hanya melakukan operasi linier.

5. Ambang (Threshold): Ambang adalah nilai ambang batas yang digunakan dalam fungsi aktivasi. Jika hasil dari fungsi agregasi memenuhi ambang batas, neuron akan menghasilkan output yang sesuai. Ambang ini mempengaruhi sejauh mana neuron akan merespons terhadap masukan.
6. Keluaran (Output): Keluaran dari neuron merupakan hasil akhir dari proses pengolahan informasi berdasarkan masukan, bobot, fungsi agregasi, dan fungsi aktivasi. Output dapat digunakan sebagai masukan untuk neuron lain dalam jaringan atau sebagai hasil akhir dari jaringan syaraf tiruan.

Dalam jaringan syaraf tiruan yang lebih kompleks, neuron-neuron ini dihubungkan satu sama lain melalui koneksi dan membentuk arsitektur jaringan yang lebih luas. Neuron-neuron ini bekerja secara paralel untuk memproses informasi dan mempelajari pola yang terkandung dalam data.

Model syaraf tiruan merupakan fondasi dari kecerdasan buatan yang berfokus pada pemodelan pengolahan informasi seperti yang terjadi dalam otak manusia. Dengan menggabungkan banyak neuron dan mengatur koneksi antara mereka, jaringan syaraf tiruan dapat menghasilkan model yang dapat mempelajari dan menggeneralisasi dari data serta melakukan tugas-tugas seperti klasifikasi, prediksi, dan pengenalan pola.

C. aktivasi dan arsitektur jaringan pada JST

Dalam konteks Jaringan Syaraf Tiruan (JST), istilah "aktivasi" merujuk pada respons atau keluaran yang dihasilkan oleh sebuah neuron atau lapisan neuron dalam jaringan. Aktivasi ditentukan oleh fungsi aktivasi yang diterapkan pada hasil dari fungsi agregasi atau kombinasi dari masukan dan bobot yang diterima oleh neuron.

Fungsi aktivasi pada JST memiliki peran penting dalam memperkenalkan sifat non-linearitas ke dalam jaringan. Dalam model syaraf tiruan yang sederhana, fungsi aktivasi bisa berupa fungsi step (misalnya, fungsi Heaviside), yang menghasilkan keluaran biner berdasarkan ambang batas yang ditentukan. Namun, dalam jaringan yang lebih kompleks, fungsi aktivasi yang umum digunakan adalah:

1. Sigmoid: Fungsi sigmoid (seperti sigmoid logistik atau tangen hiperbolik) menghasilkan keluaran yang berada dalam rentang tertentu, seperti antara 0 dan 1. Fungsi ini biasa digunakan dalam jaringan syaraf tiruan untuk tugas klasifikasi biner atau multi-kelas.

2. ReLU (Rectified Linear Unit): Fungsi ReLU menghasilkan keluaran nol untuk nilai masukan negatif dan keluaran sama dengan nilai masukan untuk nilai masukan positif. Fungsi ini memiliki keuntungan dalam mempercepat pelatihan jaringan dan mengurangi masalah gradien yang hilang.
3. Fungsi aktivasi Lainnya: Selain sigmoid dan ReLU, terdapat berbagai fungsi aktivasi lain yang digunakan dalam JST, termasuk fungsi softmax untuk klasifikasi multi-kelas, fungsi tanh, fungsi step, dan lain-lain. Pilihan fungsi aktivasi tergantung pada jenis masalah yang dihadapi dan karakteristik data yang diolah.

Arsitektur jaringan pada JST mengacu pada struktur dan organisasi neuron-neuron dalam jaringan. Arsitektur jaringan melibatkan pemilihan jumlah lapisan dan jumlah neuron dalam setiap lapisan. Berikut ini adalah beberapa arsitektur jaringan yang umum dalam JST:

1. Jaringan Feedforward: Ini adalah jenis jaringan paling dasar, di mana informasi mengalir dalam satu arah dari lapisan masukan ke lapisan tersembunyi dan akhirnya ke lapisan keluaran. Tidak ada umpan balik (feedback) dari lapisan keluaran ke lapisan masukan dalam jaringan ini.
2. Jaringan Rekurens: Jaringan rekurens memiliki koneksi siklus, yang memungkinkan umpan balik dari lapisan keluaran kembali ke lapisan masukan. Ini memungkinkan jaringan untuk mempertahankan keadaan internal atau memori yang dihasilkan oleh informasi sebelumnya.
3. Arsitektur Multilayer Perceptron (MLP): MLP adalah jaringan feedforward dengan setidaknya satu lapisan tersembunyi antara lapisan masukan dan keluaran. Lapisan tersembunyi berfungsi untuk memproses dan menggali fitur-fitur yang terkandung dalam data.
4. Convolutional Neural Network (CNN): CNN adalah jenis arsitektur jaringan yang efektif untuk memproses data berbentuk grid, seperti gambar. Jaringan ini menggunakan lapisan konvolusi untuk mengidentifikasi fitur-fitur visual dan lapisan tersembunyi untuk menggabungkan dan mengklasifikasikan fitur-fitur tersebut.
5. Recurrent Neural Network (RNN): RNN adalah jenis arsitektur jaringan rekurens yang mampu mengolah data berurutan, seperti urutan waktu atau urutan teks. Jaringan ini memiliki koneksi siklus yang memungkinkan informasi di masa lalu mempengaruhi komputasi di masa depan.

Pilihan arsitektur jaringan tergantung pada jenis masalah yang ingin diselesaikan dan karakteristik data yang diolah. Setiap arsitektur memiliki kelebihan dan kelemahan tertentu serta aplikasi yang lebih sesuai tergantung pada tugas yang dihadapi

D. supervised learning dan unsupervised learning

Supervised learning dan unsupervised learning adalah dua paradigma utama dalam pembelajaran mesin (machine learning). Berikut adalah penjelasan singkat untuk kedua paradigma tersebut:

1. **Supervised Learning (Pembelajaran Terbimbing):** Dalam supervised learning, algoritma mesin belajar menggunakan dataset yang telah diberi label (labelled dataset) yang menghubungkan setiap contoh data dengan label atau kelas yang sesuai. Tujuan utama dari supervised learning adalah untuk mempelajari pola atau hubungan antara fitur (input) dengan label (output) yang diberikan.

Proses pembelajaran terbimbing melibatkan dua tahap utama: tahap pelatihan dan tahap pengujian. Pada tahap pelatihan, algoritma belajar dari data yang telah diberi label untuk membangun model yang dapat memprediksi label untuk data baru. Pada tahap pengujian, model tersebut diuji dengan menggunakan data yang tidak diberi label untuk mengukur kinerja atau akurasi prediksi model.

Contoh dari supervised learning termasuk klasifikasi (classification), di mana tujuannya adalah untuk memprediksi kelas atau label yang diskret, dan regresi (regression), di mana tujuannya adalah untuk memprediksi nilai yang kontinu. Contoh algoritma supervised learning adalah Decision Trees, Naive Bayes, Support Vector Machines (SVM), dan Artificial Neural Networks (ANN).

2. **Unsupervised Learning (Pembelajaran Tanpa Pengawasan):** Dalam unsupervised learning, algoritma mesin belajar menggunakan dataset yang tidak memiliki label (unlabelled dataset). Tujuan dari unsupervised learning adalah untuk menemukan pola, struktur, atau informasi tersembunyi yang ada dalam dataset tanpa adanya informasi label yang terkait.

Proses unsupervised learning tidak memerlukan label atau kelas yang ditentukan sebelumnya. Algoritma ini menggali informasi dari data dengan mencari pola atau kesamaan di antara contoh-contoh data. Contoh tugas unsupervised learning termasuk clustering (pengelompokan) untuk mengelompokkan data berdasarkan kesamaan dan asosiasi untuk menemukan hubungan antara entitas dalam dataset.

Contoh algoritma unsupervised learning termasuk K-Means Clustering, Hierarchical Clustering, Principal Component Analysis (PCA), dan Generative Adversarial Networks (GAN).

Perbedaan utama antara supervised learning dan unsupervised learning terletak pada ketersediaan label atau kelas dalam dataset. Supervised learning memanfaatkan informasi label yang diberikan untuk mempelajari pola, sedangkan unsupervised learning mencari pola atau struktur tanpa label yang sudah ditentukan.

BAB XIII

ALGORITMA GENETIKA

A. Definisi algoritma genetika

Algoritma genetika adalah teknik optimisasi yang terinspirasi dari mekanisme evolusi dan seleksi alam dalam teori evolusi. Algoritma genetika digunakan untuk mencari solusi terbaik atau pendekatan yang optimal untuk masalah yang kompleks atau optimisasi.

Dalam algoritma genetika, solusi dari masalah yang ingin dipecahkan direpresentasikan dalam bentuk kromosom atau individu. Setiap kromosom terdiri dari gen-gen yang merepresentasikan atribut-atribut atau parameter-parameter solusi. Gen-gen ini dapat berupa bilangan biner, bilangan bulat, bilangan real, atau representasi lainnya tergantung pada jenis masalah yang sedang diselesaikan.

Proses algoritma genetika melibatkan evolusi dan seleksi berulang-ulang dalam populasi kromosom untuk mencapai solusi yang lebih baik. Berikut adalah langkah-langkah umum dalam algoritma genetika:

1. Inisialisasi Populasi: Populasi awal kromosom atau individu dibangkitkan secara acak atau berdasarkan heuristik tertentu.
2. Evaluasi Fitness: Setiap individu dalam populasi dinilai berdasarkan tingkat kecocokannya dengan kriteria solusi yang diinginkan. Fungsi fitness digunakan untuk mengukur kualitas solusi yang dihasilkan oleh setiap individu.
3. Seleksi: Individu-individu dengan tingkat kecocokan yang lebih tinggi memiliki peluang yang lebih besar untuk dipilih sebagai "orang tua" untuk generasi berikutnya. Prinsip seleksi ini didasarkan pada konsep seleksi alam dalam teori evolusi.
4. Rekombinasi (Crossover): Proses rekombinasi menggabungkan informasi genetik dari dua individu "orang tua" yang dipilih secara acak untuk menghasilkan individu "anak" baru. Rekombinasi ini dilakukan dengan menggunakan teknik crossover yang membagi dan menggabungkan bagian-bagian kromosom antara dua individu.

5. Mutasi: Mutasi secara acak memperkenalkan variasi genetik baru dalam populasi dengan mengubah beberapa gen individu dengan probabilitas rendah. Mutasi ini memungkinkan adanya penjelajahan ruang solusi yang lebih luas.
6. Penggantian Generasi: Individu-individu "anak" yang dihasilkan menggantikan individu-individu "orang tua" dalam populasi. Dengan demikian, populasi berevolusi dan menghasilkan generasi baru yang lebih baik dalam hal kualitas solusi.

Proses seleksi, rekombinasi, dan mutasi diulang secara berulang-ulang hingga kriteria penghentian tertentu tercapai, misalnya mencapai solusi yang optimal atau mencapai jumlah iterasi tertentu.

Algoritma genetika adalah metode yang sangat fleksibel dan dapat diterapkan dalam berbagai masalah optimisasi, penggalian data, pembelajaran mesin, dan pemodelan sistem kompleks. Keunggulan utama algoritma genetika adalah kemampuannya untuk mengeksplorasi ruang solusi secara efisien dan menemukan pendekatan yang optimal

B. Komponen komponen algoritma genetika

Algoritma genetika terdiri dari beberapa komponen yang bekerja bersama untuk mencapai optimisasi atau pencarian solusi terbaik. Berikut adalah komponen-komponen utama dalam algoritma genetika:

1. Representasi Kromosom: Kromosom merupakan representasi solusi dalam algoritma genetika. Kromosom terdiri dari gen-gen yang merepresentasikan atribut-atribut atau parameter-parameter solusi. Representasi kromosom dapat berupa string biner, bilangan bulat, bilangan real, atau representasi lainnya tergantung pada jenis masalah yang sedang diselesaikan.
2. Populasi: Populasi adalah kumpulan individu atau kromosom yang ada dalam satu generasi dalam algoritma genetika. Populasi awal dibangkitkan secara acak atau berdasarkan heuristik tertentu.
3. Fungsi Fitness: Fungsi fitness digunakan untuk mengevaluasi kualitas solusi dari setiap individu dalam populasi. Fungsi ini mengukur sejauh mana solusi tersebut memenuhi kriteria atau tujuan yang diinginkan dalam

masalah yang sedang diselesaikan. Individu dengan tingkat kecocokan (fitness) yang lebih tinggi memiliki peluang lebih besar untuk dipilih dalam proses seleksi.

4. Seleksi: Seleksi adalah proses memilih individu-individu yang memiliki tingkat kecocokan yang lebih tinggi untuk menjadi "orang tua" dalam menghasilkan generasi baru. Prinsip seleksi didasarkan pada konsep seleksi alam dalam teori evolusi. Ada beberapa teknik seleksi yang umum digunakan, seperti seleksi turnamen, seleksi proporsional, dan seleksi elitisme.

5. Rekombinasi (Crossover): Rekombinasi adalah proses menggabungkan informasi genetik dari dua individu "orang tua" yang dipilih secara acak untuk menghasilkan individu "anak" baru. Rekombinasi dilakukan dengan membagi dan menggabungkan bagian-bagian kromosom antara dua individu. Teknik crossover yang umum digunakan antara lain titik pemotongan (single-point crossover), dua titik pemotongan (two-point crossover), dan uniform crossover.

6. Mutasi: Mutasi adalah proses secara acak memperkenalkan variasi genetik baru dalam populasi dengan mengubah beberapa gen individu dengan probabilitas rendah. Mutasi memungkinkan adanya penjelajahan ruang solusi yang lebih luas dan membantu keluar dari kemungkinan solusi lokal yang tidak optimal.

7. Penggantian Generasi: Setelah rekombinasi dan mutasi dilakukan, individu "anak" menggantikan individu "orang tua" dalam populasi. Dengan demikian, populasi berevolusi dan menghasilkan generasi baru. Proses seleksi, rekombinasi, dan mutasi diulang secara berulang hingga kriteria penghentian tercapai.

Selain komponen-komponen utama tersebut, ada juga parameter-parameter yang perlu ditentukan dalam algoritma genetika, seperti ukuran populasi, probabilitas rekombinasi, probabilitas mutasi, dan kriteria penghentian. Parameter-parameter ini akan mempengaruhi kiner

BAB XIV

COGNITIF SCIENCE

A. Defenisi

Cognitive science dan kecerdasan buatan (artificial intelligence, AI) memiliki keterkaitan erat dalam upaya memahami dan mereplikasi kemampuan kognitif manusia. Berikut adalah beberapa cara di mana cognitive science berkontribusi pada pengembangan kecerdasan buatan:

1. **Inspirasi dari Model Kognitif:** Cognitive science menyediakan wawasan tentang bagaimana manusia memproses informasi, berpikir, dan mengambil keputusan. Konsep-konsep ini digunakan sebagai dasar untuk mengembangkan model kognitif dalam kecerdasan buatan. Misalnya, pemodelan pikiran manusia dalam bidang psikologi kognitif dapat menginspirasi pengembangan algoritma dan sistem kecerdasan buatan yang meniru kemampuan kognitif manusia.
2. **Pemahaman Bahasa:** Linguistik kognitif adalah cabang cognitive science yang mempelajari hubungan antara bahasa dan pemikiran. Pemahaman tentang bagaimana manusia memahami, menghasilkan, dan menggunakan bahasa telah berkontribusi pada pengembangan sistem kecerdasan buatan dalam pemrosesan bahasa alami, terjemahan mesin, dan analisis teks.
3. **Pemodelan Sistem Kognitif:** Cognitive science menyediakan landasan untuk memodelkan sistem kognitif manusia, termasuk memori, perhatian, dan pembelajaran. Prinsip-prinsip ini kemudian diterapkan dalam pengembangan kecerdasan buatan untuk meningkatkan pemrosesan informasi, pengambilan keputusan, dan kemampuan adaptasi.
4. **Neurosains Kognitif:** Keterkaitan antara neurosains kognitif dan kecerdasan buatan melibatkan pemahaman tentang aktivitas otak yang terkait dengan fungsi kognitif. Studi tentang otak manusia dapat memberikan wawasan tentang struktur dan proses kognitif yang kemudian digunakan dalam pengembangan model dan algoritma kecerdasan buatan.
5. **Penggunaan Data dan Analisis:** Cognitive science juga berperan dalam penggunaan data dan analisis untuk memahami perilaku manusia, pola pikir, dan preferensi. Pengetahuan ini dapat dimanfaatkan dalam pengembangan kecerdasan buatan yang mampu mempelajari dari data, mengenali pola, dan

menghasilkan rekomendasi atau keputusan yang sesuai dengan preferensi pengguna.

Melalui integrasi pengetahuan dan metode dari cognitive science, kecerdasan buatan dapat menjadi lebih efektif dalam meniru, mereplikasi, atau meningkatkan kemampuan kognitif manusia. Pendekatan multidisiplin antara cognitive science dan kecerdasan buatan memungkinkan pengembangan sistem yang lebih cerdas, adaptif, dan responsif terhadap tugas-tugas kompleks.