

DIKTAT

**KECERDASAN BUATAN
(ARTIFICIAL INTELLIGENCE)
Edisi Revisi I**



Oleh:

RAKHMAT KURNIAWAN. R, S.T., M.KOM

NIDN: 2016038501

**FAKULTAS SAINS DAN TEKNOLOGI
UIN SUMATERA UTARA MEDAN**

2020

KATA PENGANTAR

Puji syukur kehadirat Allah SWT berkat rahmat dan karunia-Nya penulis dapat menyelesaikan Diktat Matakuliah Arsitektur dan Organisasi Komputer.

Matakuliah Kecerdasan Buatan adalah salah satu matakuliah wajib yang ada pada kurikulum Program Studi Ilmu Komputer di Fakultas Sains dan Teknologi UIN Sumatera Utara Medan.

Penulis mengucapkan banyak terima kasih atas dukungan dan bantuan para pimpinan, rekan-rekan dosen, teman sejawat di lingkungan Fakultas Sains dan Teknologi UIN Sumatera Utara Medan atas terselesaikannya Diktat ini. Semoga Diktat ini dapat membantu dan mendukung kegiatan pembelajar yang ada di Program Studi Ilmu Komputer.

Penulis menyadari Diktat Kecerdasan Buatan ini masih banyak kekurangan dan keterbatasan, maka dalam hal ini penulis sangat mengharapkan kritik dan saran yang membangun dari berbagai pihak guna penyempurnaan Diktat ini dimasa yang akan datang.

Akhir kata semoga segala usaha yang penulis lakukan ini bermanfaat bagi kita semua, dan Semoga Allah SWT berkenan memberikan berkahnya sehingga semua harapan dan cita-cita penulis dapat terkabulkan. Amin

Medan, September 2020

Rakhmat Kurniawan. R, S.T., M.Kom

DAFTAR ISI

| | |
|---|-----|
| KATA PENGANTAR..... | ii |
| DAFTAR ISI | iii |
| BAB I | 1 |
| PENGANTAR KECERDASAN BUATAN..... | 1 |
| 1.1. Definisi Kecerdasan Buatan..... | 1 |
| 1.2. Beda Kecerdasan Buatan & Kecerdasan Alami | 1 |
| 1.3. Sejarah Kecerdasan Buatan | 3 |
| 1.4. Fondasi Ilmu Kecerdasan Buatan | 4 |
| 1.5. Kecerdasan Buatan Pada Aplikasi Komersial | 5 |
| BAB II..... | 9 |
| BIDANG ILMU AI..... | 9 |
| 2.1. Sistem Pakar | 9 |
| 2.2. Algoritma genetika | 11 |
| 2.3. Logika Fuzzy..... | 12 |
| 2.4. Jaringan Syaraf Tiruan | 12 |
| 2.5. Robotika | 14 |
| BAB III | 16 |
| RUANG KEADAAN DAN TEKNIK PENCARIAN | 16 |
| 3.1. Mendefinisikan Masalah Sebagai Suatu Ruang Keadaan | 16 |
| 3.2. Metode Pelacakan/Pencarian | 23 |
| BAB IV..... | 34 |
| REPRESENTASI PENGETAHUAN | 34 |
| 4.1. Logika | 34 |

| | |
|---|----|
| 4.1.1. Logika Proposisi | 36 |
| 4.1.2. Logika Predikat | 39 |
| 4.2. List Dan Pohon/Tree | 42 |
| 4.2.1. List | 42 |
| 4.2.2. Pohon/Tree | 43 |
| 4.3. Jaringan Semantik | 43 |
| 4.4. Frame | 44 |
| 4.5. Tabel Keputusan (Decission Table) | 45 |
| 4.6. Pohon Keputusan (Decission Tree) | 46 |
| 4.7. Naskah (Script) | 46 |
| 4.8. Sistem Produksi (Aturan Produksi/Production Rules) | 48 |
| BAB V | 51 |
| PENANGANAN KETIDAKPASTIAN | 51 |
| 5.1. Theorema Bayes | 51 |
| 5.2. Certainty Factor (CF) | 54 |
| 5.3. Damster-Shafer | 59 |
| BAB VI | 65 |
| SISTEM PAKAR | 65 |
| 6.1. Manfaat Sistem Pakar | 68 |
| 6.2. Kelemahan Sistem Pakar | 69 |
| 6.3. Konsep Dasar Sistem Pakar | 69 |
| 6.4. Perbedaan Sistem Konvensional Dengan Sistem Pakar | 70 |
| 6.5. Elemen Yang Terkait Pengembangan Sistem Pakar | 70 |
| 6.6. Area Permasalahan Aplikasi Sistem Pakar | 71 |
| 6.7. Bentuk / Tipe Sistem Pakar | 72 |
| 6.8. Struktur Sistem Pakar | 73 |
| 6.9. Arsitektur Sistem Pakar | 73 |

| | |
|--|-----|
| 6.10. Basis Pengetahuan (Knowledge Base) | 76 |
| 6.11. Mesin Inferensi (Inference Engine) | 77 |
| BAB VII..... | 88 |
| MACHINE LEARNING | 88 |
| 7.1. Rote Learning | 88 |
| 7.2. Learning by Taking Advice | 88 |
| 7.3. Learning from example | 88 |
| 7.4. Learning in Problem Solving..... | 89 |
| 7.5. Discovery..... | 89 |
| BAB VIII..... | 90 |
| COMPUTER VISION DAN APLIKASINYA..... | 90 |
| BAB IX..... | 93 |
| NATURAL LANGUAGE PROCESSING..... | 93 |
| 9.1. NLP Area..... | 93 |
| 9.2. Terminologi NLP | 94 |
| 9.3. Information Retrieval..... | 95 |
| 9.4. Morphological Analysis | 96 |
| 9.5. Stemming & Lemmatization..... | 99 |
| 9.6. Contoh Aplikasi NLP..... | 100 |
| BAB X..... | 102 |
| JARINGAN SYARAF TIRUAN | 102 |
| 10.1. Sejarah | 102 |
| 10.2. Model..... | 103 |
| DAFTAR PUSTAKA | 104 |

BAB I

PENGANTAR KECERDASAN BUATAN

1.1. Definisi Kecerdasan Buatan

Kecerdasan buatan (Artificial Intelligence) adalah Bagian dari ilmu komputer yang mempelajari bagaimana membuat mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia bahkan bisa lebih baik daripada yang dilakukan manusia. ¹

Menurut John McCarthy, 1956, AI : Untuk mengetahui dan memodelkan proses – proses berpikir manusia dan mendesain mesin agar dapat menirukan perilaku manusia.

Cerdas = memiliki pengetahuan + pengalaman, penalaran (bagaimana membuat keputusan & mengambil tindakan), moral yang baik. Agar mesin bisa cerdas (bertindak seperti & sebaik manusia) maka harus diberi bekal pengetahuan & mempunyai kemampuan untuk menalar.

2 bagian utama yg dibutuhkan untuk aplikasi kecerdasan buatan :

- 1) basis pengetahuan (knowledge base): berisi fakta-fakta, teori, pemikiran & hubungan antara satu dengan lainnya.
- 2) motor inferensi (inference engine) : kemampuan menarik kesimpulan berdasarkan pengetahuan

1.2. Beda Kecerdasan Buatan & Kecerdasan Alami

Kelebihan kecerdasan buatan :

- 1) Lebih bersifat permanen. Kecerdasan alami bisa berubah karena sifat manusia pelupa. Kecerdasan buatan tidak berubah selama sistem komputer & program tidak mengubahnya.
- 2) Lebih mudah diduplikasi & disebar. Mentransfer pengetahuan manusia dari 1 orang ke orang lain membutuhkan proses yang sangat lama & keahlian tidak akan pernah dapat diduplikasi dengan lengkap. Jadi jika pengetahuan terletak pada

¹ Andri Kristanto, *Kecerdasan Buatan* (Yogyakarta: Graha Ilmu, 2004).

suatu sistem komputer, pengetahuan tersebut dapat disalin dari komputer tersebut & dapat dipindahkan dengan mudah ke komputer yang lain.

- 3) Lebih murah. Menyediakan layanan komputer akan lebih mudah & murah dibandingkan mendatangkan seseorang untuk mengerjakan sejumlah pekerjaan dalam jangka waktu yang sangat lama.
- 4) Bersifat konsisten dan teliti karena kecerdasan buatan adalah bagian dari teknologi komputer sedangkan kecerdasan alami senantiasa berubah-ubah
- 5) Dapat didokumentasi. Keputusan yang dibuat komputer dapat didokumentasi dengan mudah dengan cara melacak setiap aktivitas dari sistem tersebut. Kecerdasan alami sangat sulit untuk direproduksi.
- 6) Dapat mengerjakan beberapa task lebih cepat dan lebih baik dibanding manusia

Kelebihan kecerdasan alami:

- 1) Kreatif: manusia memiliki kemampuan untuk menambah pengetahuan, sedangkan pada kecerdasan buatan untuk menambah pengetahuan harus dilakukan melalui sistem yang dibangun.
- 2) Memungkinkan orang untuk menggunakan pengalaman atau pembelajaran secara langsung. Sedangkan pada kecerdasan buatan harus mendapat masukan berupa input-input simbolik.
- 3) Pemikiran manusia dapat digunakan secara luas, sedangkan kecerdasan buatan sangat terbatas.

Tabel 1.1. Beda Kecerdasan Buatan & Program Konvensional

| | Kecerdasan buatan | Program konvensional |
|-------------------|---|--|
| Fokus pemrosesan | Konsep simbolik / numerik (pengetahuan) | Data & informasi |
| Pencarian | Heuristik | Algoritma |
| Sifat input | Bisa tidak lengkap | Harus lengkap |
| Keterangan | Disediakan | Biasanya tidak disediakan |
| Struktur | Kontrol dipisahkan dari pengetahuan | Kontrol terintegrasi dengan informasi (data) |
| Sifat output | Kuantitatif | Kualitatif |
| Kemampuan menalar | Ya | Tidak |

Program kecerdasan buatan dapat ditulis dalam semua bahasa komputer, baik dalam bahasa C, Pascal, Basic, dan bahasa pemrograman lainnya. Tetapi dalam

perkembangan selanjutnya, dikembangkan bahasa pemrograman yang khusus untuk aplikasi kecerdasan buatan yaitu LISP dan PROLOG.²

1.3. Sejarah Kecerdasan Buatan

Tahun 1950 – an Alan Turing, seorang pionir AI dan ahli matematika Inggris melakukan percobaan Turing (Turing Test) yaitu sebuah komputer melalui terminalnya ditempatkan pada jarak jauh. Di ujung yang satu ada terminal dengan software AI dan diujung lain ada sebuah terminal dengan seorang operator. Operator itu tidak mengetahui kalau di ujung terminal lain dipasang software AI. Mereka berkomunikasi dimana terminal di ujung memberikan respon terhadap serangkaian pertanyaan yang diajukan oleh operator. Dan sang operator itu mengira bahwa ia sedang berkomunikasi dengan operator lainnya yang berada pada terminal lain.

Turing beranggapan bahwa jika mesin dapat membuat seseorang percaya bahwa dirinya mampu berkomunikasi dengan orang lain, maka dapat dikatakan bahwa mesin tersebut cerdas (seperti layaknya manusia).

Turing memprediksi bahwa pada tahun 2000, komputer mungkin memiliki kesempatan 30% untuk membodohi orang awam selama 5 menit. Prediksi Turing tersebut terbukti. Saat ini komputer sudah dapat melakukan serangkaian tes Turing yang dikenal sebagai imitation game. Untuk dapat melakukan hal tersebut komputer perlu memiliki beberapa kemampuan yaitu:

- ✓ Pemrosesan bahasa alami (natural language processing) agar komputer dapat berkomunikasi dengan bahasa alami manusia.
- ✓ Representasi pengetahuan (knowledge representation) untuk menyimpan apa yang diketahuinya.
- ✓ Penalaran otomatis (Automated reasoning) yang menggunakan informasi yang tersimpan untuk menjawab pertanyaan maupun menarik kesimpulan baru.
- ✓ Pembelajaran mesin (Machine learning) untuk beradaptasi pada lingkungan baru dan mendeteksi serta mengenali pola
- ✓ Computer vision untuk menangkap dan mempersepsikan obyek.
- ✓ Robotika untuk memanipulasi obyek dan bergerak

² Sri Kusumadewi, *Artificial Intelligence (Teknik Dan Aplikasinya)* (Yogyakarta: Graha Ilmu, 2003).

Keenam disiplin ini membentuk ilmu AI, dan Allan Turing adalah orang yang berjasa mendesain serangkaian tes yang tetap relevan 50 tahun kedepan.³

1.4. Fondasi Ilmu Kecerdasan Buatan

Ilmu kecerdasan buatan merupakan kontribusi dari berbagai disiplin ilmu, tidak eksklusif dari bidang komputer saja. Fondasi ilmu yang membentuk kecerdasan buatan yaitu:

1) Filsafat:

- Pikiran adalah seperti mesin
- Menggunakan pengetahuan untuk mengoperasikan sesuatu
- Pemikiran digunakan untuk mengambil tindakan

2) Matematika

- Alat untuk memanipulasi pernyataan logis
- Memahami perhitungan
- Penalaran algoritma

3) Ekonomi

Pengambilan keputusan untuk memaksimalkan hasil yang diharapkan

4) Neuroscience (Ilmu syaraf)

- Studi tentang sistem syaraf
- Bagaimana otak memproses informasi

5) Psikologi

- Ilmu kognitif
- Manusia dan hewan adalah mesin pengolah informasi

6) Teknik komputer

Menyediakan alat/artefak untuk aplikasi AI

7) Teori kontrol

Merancang perangkat yang bertindak optimal berdasarkan umpan balik dari lingkungan.

8) Bahasa

³ Ibid.

Bagaimana bahasa berhubungan dengan pikiran. Knowledge representation language dan natural language processing.

1.5. Kecerdasan Buatan Pada Aplikasi Komersial

Lingkup utama kecerdasan buatan :

- 1) Sistem pakar (expert system) : komputer sebagai sarana untuk menyimpan pengetahuan para pakar sehingga komputer memiliki keahlian menyelesaikan permasalahan dengan meniru keahlian yang dimiliki pakar.

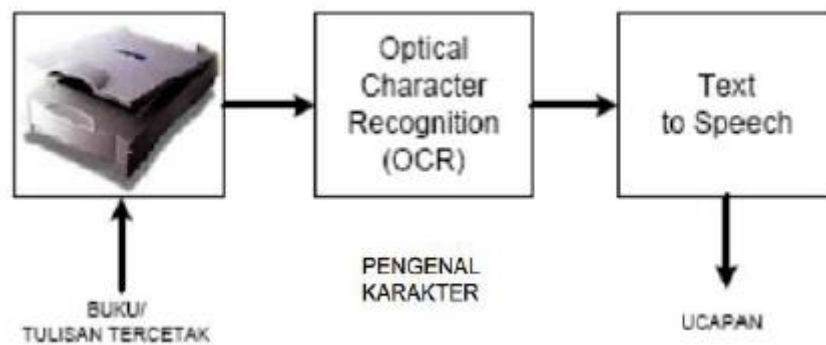
Diagnosa Penyakit THT

Apakah Anda demam (Y/T) ? y
 Apakah Anda sakit kepala (Y/T) ? y
 Apakah Anda merasa nyeri pada saat berbicara atau menelan (Y/T) ? y
 Apakah Anda batuk (Y/T) ? y
 Apakah Anda mengalami nyeri tenggorokan (Y/T) ? y
 Apakah selaput lendir Anda berwarna merah dan bengkak (Y/T) ? y

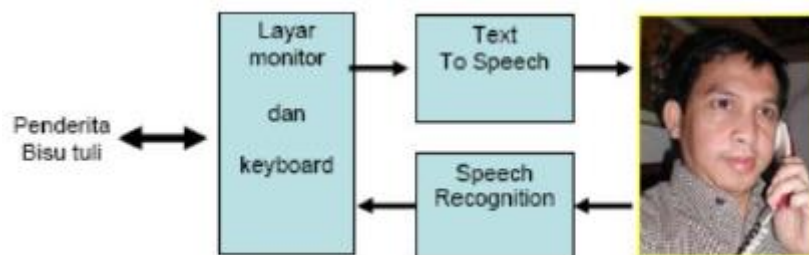
Penyakit Anda adalah TONSILITIS

Ingin mengulang lagi (Y/T) ?

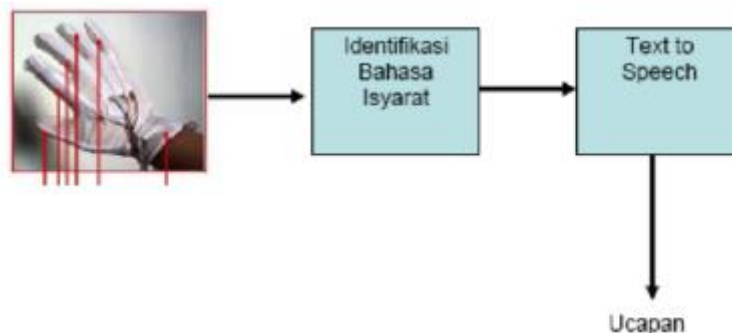
- 2) Pengolahan bahasa alami (natural language processing): user dapat berkomunikasi dengan komputer menggunakan bahasa sehari-hari, misal bahasa inggris, bahasa indonesia, bahasa jawa, dll, contoh :
 - Pengguna sistem dapat memberikan perintah dengan bahasa sehari-hari, misalnya, untuk menghapus semua file, pengguna cukup memberikan perintah "komputer, tolong hapus semua file !" maka sistem akan mentranslasikan perintah bahasa alami tersebut menjadi perintah bahasa formal yang dipahami oleh komputer, yaitu "delete *.* <ENTER>".
 - Translator bahasa inggris ke bahasa indonesia begitu juga sebaliknya,dll, tetapi sistem ini tidak hanya sekedar kamus yang menerjemahkan kata per kata, tetapi juga mentranslasikan sintaks dari bahasa asal ke bahasa tujuan
 - Text summarization : suatu sistem yang dapat membuat ringkasan hal-hal penting dari suatu wacana yang diberikan.
- 3) Pengenalan ucapan (speech recognition) : manusia dapat berkomunikasi dengan komputer menggunakan suara. Contoh :
 - memberikan instruksi ke komputer dengan suara
 - alat bantu membaca untuk tunanetra, mempunyai masukan berupa teks tercetak (misalnya buku) dan mempunyai keluaran berupa ucapan dari teks tercetak yang diberikan.



- Telpon untuk penderita bisu-tuli



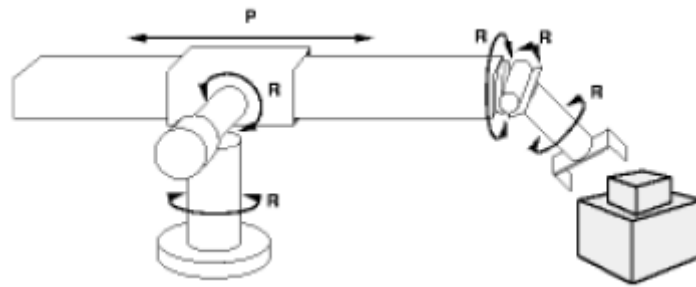
- Alat untuk tuna wicara




konversi dari SMS (Short Message System) ke ucapan sehingga pesan SMS dapat didengar. Dengan demikian memungkinkan untuk mendengar pesan SMS sambil melakukan aktivitas yang menyulitkan untuk membacanya, seperti mengendarai mobil.

4) Robotika & sistem sensor

- Sistem sensor pada mesin cuci yaitu menggunakan sensor optik, mengeluarkan cahaya ke air dan mengukur bagaimana cahaya tersebut sampai ke ujung lainnya. Makin kotor, maka sinar yang sampai makin redup. Sistem juga mampu menentukan jenis kotoran tersebut daki/minyak. Sistem juga bisa menentukan putaran yang tepat secara otomatis berdasarkan jenis dan banyaknya kotoran serta jumlah yang akan dicuci.
- Robotika



- 5) Computer vision : menginterpretasikan gambar atau objek-objek tampak melalui komputer.

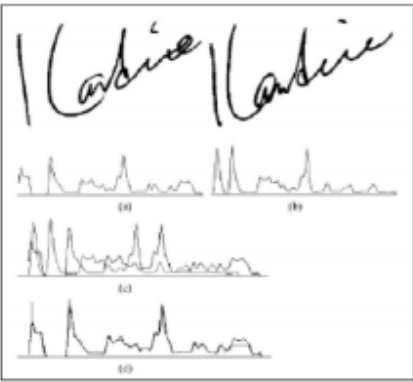


Pengenalan pola sidik jari seseorang

| Designated Writing Area | | Identify |
|-------------------------|----------------|----------------------------|
| Character: | English: _____ | Number: _____ |
| | _____ | 9 |
| | | No.1 d=74.02 w=1.624 |
| Data from William Go | | |
| Total: 20 | Done: 20 | First: 20 |
| Second: 0 | Third: 0 | Other: 0 |
| Reject: 0 | | |

0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9

Pengenalan angka



Pengenalan/deteksi tanda tangan asli/palun (tanda tangan yang dibuat oleh orang yang sama/berbeda)

- 6) Intelligent computer-aided instruction : komputer dapat digunakan sebagai tutor yang dapat melatih & mengajar. Contoh : Learn to speak English



7) Game playing

8) SOFT Computing

Soft computing merupakan inovasi baru dalam membangun sistem cerdas yaitu sistem yang memiliki keahlian seperti manusia pada domain tertentu, mampu beradaptasi dan belajar agar dapat bekerja lebih baik jika terjadi perubahan lingkungan. Soft computing mengeksplorasi adanya toleransi terhadap ketidaktepatan, ketidakpastian, dan kebenaran parsial untuk dapat diselesaikan dan dikendalikan dengan mudah agar sesuai dengan realita (Prof. Lotfi A Zadeh, 1992).

Metodologi-metodologi yang digunakan dalam Soft computing adalah :

- ✓ Sistem Fuzzy (mengakomodasi ketidaktepatan) Logika Fuzzy (fuzzy logic)
- ✓ Jaringan Syaraf (menggunakan pembelajaran) Jaringan Syaraf Tiruan (neural network)
- ✓ Probabilistic Reasoning (mengakomodasi ketidakpastian)
- ✓ Evolutionary Computing (optimasi) Algoritma Genetika⁴

⁴ Ibid.

BAB II

BIDANG ILMU AI

2.1. Sistem Pakar

Sistem pakar adalah suatu program komputer yang dirancang untuk mengambil keputusan seperti keputusan yang diambil oleh seorang atau beberapa orang pakar. Menurut Marimin (1992), sistem pakar adalah sistem perangkat lunak komputer yang menggunakan ilmu, fakta, dan teknik berpikir dalam pengambilan keputusan untuk menyelesaikan masalah-masalah yang biasanya hanya dapat diselesaikan oleh tenaga ahli dalam bidang yang bersangkutan.

Dalam penyusunannya, sistem pakar mengkombinasikan kaidah-kaidah penarikan kesimpulan (inference rules) dengan basis pengetahuan tertentu yang diberikan oleh satu atau lebih pakar dalam bidang tertentu. Kombinasi dari kedua hal tersebut disimpan dalam komputer, yang selanjutnya digunakan dalam proses pengambilan keputusan untuk penyelesaian masalah tertentu.

Suatu sistem pakar disusun oleh tiga modul utama (Staugaard, 1987), yaitu :

1) Modul Penerimaan Pengetahuan Knowledge Acquisition Mode)

Sistem berada pada modul ini, pada saat ia menerima pengetahuan dari pakar. Proses mengumpulkan pengetahuan-pengetahuan yang akan digunakan untuk pengembangan sistem, dilakukan dengan bantuan knowledge engineer. Peran knowledge engineer adalah sebagai penghubung antara suatu sistem pakar dengan pakarnya

2) ModulKonsultasi(ConsultationMode)

Pada saat sistem berada pada posisi memberikan jawaban atas permasalahan yang diajukan oleh user, sistem pakar berada dalam modul konsultasi. Pada modul ini, user berinteraksi dengan sistem dengan menjawab pertanyaan-pertanyaan yang diajukan oleh sistem

3) Modul Penjelasan(Explanation Mode)

Modul ini menjelaskan proses pengambilan keputusan oleh sistem (bagaimana suatu keputusan dapat diperoleh).⁵

Komponen utama pada struktur sistem pakar (Hu et al, 1987) meliputi:

1) Basis Pengetahuan (Knowledge Base)

Basis pengetahuan merupakan inti dari suatu sistem pakar, yaitu berupa representasi pengetahuan dari pakar. Basis pengetahuan tersusun atas fakta dan kaidah. Fakta adalah informasi tentang objek, peristiwa, atau situasi. Kaidah adalah cara untuk membangkitkan suatu fakta baru dari fakta yang sudah diketahui. Menurut Gondran (1986) dalam Utami (2002), basis pengetahuan merupakan representasi dari seorang pakar, yang kemudian dapat dimasukkan kedalam bahasa pemrograman khusus untuk kecerdasan buatan (misalnya PROLOG atau LISP) atau shell system pakar (misalnya EXSYS, PC-PLUS, CRYSTAL, dsb).

2) Mesin Inferensi (Inference Engine)

Mesin inferensi berperan sebagai otak dari sistem pakar. Mesin inferensi berfungsi untuk memandu proses penalaran terhadap suatu kondisi, berdasarkan pada basis pengetahuan yang tersedia. Di dalam mesin inferensi terjadi proses untuk memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan dalam rangka mencapai solusi atau kesimpulan. Dalam prosesnya, mesin inferensi menggunakan strategi penalaran dan strategi pengendalian.

Strategi penalaran terdiri dari strategi penalaran pasti (Exact Reasoning) dan strategi penalaran tak pasti (Inexact Reasoning). Exact reasoning akan dilakukan jika semua data yang dibutuhkan untuk menarik suatu kesimpulan tersedia, sedangkan inexact reasoning dilakukan pada keadaan sebaliknya.

Strategi pengendalian berfungsi sebagai panduan arah dalam melakukan proses penalaran. Terdapat tiga teknik pengendalian yang sering digunakan, yaitu

⁵ E. Turban, *Decision Support and Expert Systems; Management Support Systems* (New Jersey: Prentice Hall, 1995).

forward chaining, backward chaining, dan gabungan dari kedua teknik pengendalian tersebut.

3) Basis Data (Database)

Basis data terdiri atas semua fakta yang diperlukan, dimana fakta-fakta tersebut digunakan untuk memenuhi kondisi dari kaidah-kaidah dalam sistem. Basis data menyimpan semua fakta, baik fakta awal pada saat sistem mulai beroperasi, maupun fakta-fakta yang diperoleh pada saat proses penarikan kesimpulan sedang dilaksanakan. Basis data digunakan untuk menyimpan data hasil observasi dan data lain yang dibutuhkan selama pemrosesan.

4) Antarmuka Pemakai (User Interface)

Fasilitas ini digunakan sebagai perantara komunikasi antara pemakai dengan sistem. Hubungan antar komponen penyusun struktur sistem pakar dapat dilihat pada Gambar di bawah ini:

2.2. Algoritma genetika

Algoritma Genetika adalah suatu algoritma pencarian yang meniru mekanisme dari genetika alam. Algoritma Genetika ini banyak dipakai pada aplikasi bisnis, teknik maupun pada bidang keilmuan. Algoritma ini dapat dipakai untuk mendapatkan solusi yang tepat untuk masalah optimal dari satu variabel atau multi variabel. Sebelum Algoritma ini dijalankan, masalah apa yang ingin dioptimalkan itu harus dinyatakan dalam fungsitujuan, yang dikenal dengan fungsi fitness. Jika nilai fitness semakin besar, maka sistem yang dihasilkan semakin baik. Operasi yang dilakukan adalah reproduksi, crossover, dan mutasi untuk mendapatkan sebuah solusi menurut nilai fitnessnya.⁶

Selanjutnya konstruksi dasar dari Algoritma Genetika adalah sebagai berikut:

- Pendefinisian Chromosome
- Pendefinisian Fungsi Fitness
- Membangkitkan Sebuah Populasi Awal
- Reproduksi
- Crossover

⁶ Lawrence Davis, *Handbook Of Genetic Algorithms* (New York: Van Nostrand Reinhold, 1991).

- Mutasi

Contoh: Aplikasi Algoritma Genetika Untuk Merancang Fungsi Keanggotaan Pada Kendali Logika Fuzzy.

2.3. Logika Fuzzy

Logika Fuzzy (logika samar) merupakan logika yang berhadapan langsung dengan konsep kebenaran sebagian, dimana logika klasik menyatakan bahwa segala hal dapat di ekspresikan dalam binary 0 atau 1. logika fuzzy memungkinkan nilai keanggotaan antara 0 dan 1. Karena alasan diatas maka pada penelitian ini akan dibuat perancangan perangkat lunak dan perangkat keras robot avoider dengan menggunakan aplikasi Fuzzy Logic sebagai kendali system. Perlu diketahui bahwa Teori Himpunan Samar dan Logika Samar sangat berkembang pesat pada saat ini. Banyak sekali masalah-masalah nyata yang lebih tepat diselesaikan menggunakan Teori Himpunan Samar dan Logika Samar. Banyak sekali muncul teori-teori baru pada saat ini misalnya: Topologi Fuzzy, Analisa Fuzzy, Aljabar Fuzzy (Fuzzy Semi Group, Fuzzy Ring, Fuzzy Group, dan sebagainya).

Logika fuzzy telah lama dikenal dan digunakan dalam berbagai bidang oleh para ahli dan insinyur. Penggunaan logika fuzzy pada awalnya digunakan untuk beberapa bidang, seperti sistem diagnosa penyakit (dalam bidang kedokteran); pemodelan system pemasaran, riset operasi (dalam bidang ekonomi); kendali kualitas air, prediksi adanya gempa bumi, klasifikasi dan pencocokan pola (dalam bidang teknik). Penggunaan logika fuzzy dalam bidang sistem daya (power system) juga sudah dilakukan, antara lain dalam analisis kemungkinan, prediksi dan pengaturan beban, identifikasi gangguan pada generator dan penjadwalan pemeliharaan generator.⁷

2.4. Jaringan Syaraf Tiruan

Jaringan saraf tiruan (JST) (Bahasa Inggris: artificial neural network (ANN), atau juga disebut simulated neural network (SNN), atau umumnya hanya disebut neural network (NN)), adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan jaringan saraf manusia. JST merupakan sistem adaptif yang dapat

⁷ Kusuma Dewi Sri and Hari Purnomo, *Aplikasi Logika Fuzzy Untuk Mendukung Keputusan* (Yogyakarta: Graha Ilmu, 2004).

mengubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan tersebut.

Secara sederhana, JST adalah sebuah alat pemodelan data statistik non-linier. JST dapat digunakan untuk memodelkan hubungan yang kompleks antara input dan output untuk menemukan pola-pola pada data.

Saat ini bidang kecerdasan buatan dalam usahanya menirukan intelegensi manusia, belum mengadakan pendekatan dalam bentuk fisiknya melainkan dari sisi yang lain. Pertama-tama diadakan studi mengenai teori dasar mekanisme proses terjadinya intelegensi. Bidang ini disebut „Cognitive Science“. Dari teori dasar ini dibuatlah suatu model untuk disimulasikan pada komputer, dan dalam perkembangannya yang lebih lanjut dikenal berbagai sistem kecerdasan buatan yang salah satunya adalah jaringan saraf tiruan. Dibandingkan dengan bidang ilmu yang lain, jaringan saraf tiruan relatif masih baru. Sejumlah literatur menganggap bahwa konsep jaringan saraf tiruan bermula pada makalah Waffen McCulloch dan Walter Pitts pada tahun 1943. Dalam makalah tersebut mereka mencoba untuk memformulasikan model matematis sel sel otak. Metode yang dikembangkan berdasarkan sistem saraf biologi ini, merupakan suatu langkah maju dalam industri computer.

Suatu jaringan saraf tiruan memproses sejumlah besar informasi secara paralel dan terdistribusi, hal ini terinspirasi oleh model kerja otak biologis. Beberapa definisi tentang jaringan saraf tiruan adalah sebagai berikut di bawah ini. Hecht-Nielsen (1988) mendefinisikan sistem saraf buatan sebagai berikut: "Suatu neural network (NN), adalah suatu struktur pemroses informasi yang terdistribusi dan bekerja secara paralel, yang terdiri atas elemen pemroses (yang memiliki memori lokal dan beroperasi dengan informasi lokal) yang diinterkoneksi bersama dengan alur sinyal searah yang disebut koneksi. Setiap elemen pemroses memiliki koneksi keluaran tunggal yang bercabang (fan out) ke sejumlah koneksi kolateral yang diinginkan (setiap koneksi membawa sinyal yang sama dari keluaran elemen pemroses tersebut). Keluaran dari elemen pemroses tersebut dapat merupakan sebarang jenis persamaan matematis yang diinginkan. Seluruh proses yang berlangsung pada setiap elemen pemroses harus benar-benar dilakukan secara lokal, yaitu keluaran hanya bergantung pada nilai masukan pada saat itu yang diperoleh melalui koneksi dan nilai yang tersimpan dalam memori lokal". Menurut Haykin, S. (1994), *Neural Networks: A*

Comprehensive Foundation, NY, Macmillan, mendefinisikan jaringan saraf sebagai berikut: “Sebuah jaringan saraf adalah sebuah prosesor yang terdistribusi paralel dan mempunyai kecenderungan untuk menyimpan pengetahuan yang didapatkannya dari pengalaman dan membuatnya tetap tersedia untuk digunakan. Hal ini menyerupai kerja otak dalam dua hal yaitu: 1. Pengetahuan diperoleh oleh jaringan melalui suatu proses belajar. 2. Kekuatan hubungan antar sel saraf yang dikenal dengan bobot sinapsis digunakan untuk menyimpan pengetahuan.

Dan menurut Zurada, J.M. (1992), *Introduction To Artificial Neural Systems*, Boston: PWS Publishing Company, mendefinisikan sebagai berikut: “Sistem saraf tiruan atau jaringan saraf tiruan adalah sistem selular fisik yang dapat memperoleh, menyimpan dan menggunakan pengetahuan yang didapatkan dari pengalaman”. DARPA Neural Network Study (1988, AFCEA International Press, p. 60) mendefinisikan jaringan syaraf buatan sebagai berikut : Sebuah jaringan syaraf adalah sebuah sistem yang dibentuk dari sejumlah elemen pemroses sederhana yang bekerja secara paralel dimana fungsinya ditentukan oleh stuktur jaringan, kekuatan hubungan, dan pengolahan dilakukan pada komputasi elemen atau nodes.⁸

2.5. Robotika

Robotika adalah salah satu wacana teknologi untuk menuju peradaban yang lebih maju. Kebanyakan orang selalu beranggapan bahwa robot adalah kemajuan teknologi yang mampu menggeser tingkah laku seseorang untuk melakukan suatu tindakan. Dengan kemajuan yang pesat, maka kebutuhan akan SDM akan merosot tajam. Layaknya revolusi pada bangsa Eropa.

Sangat disayangkan selali bila titik ikon kemajuan teknologi tersebut tidak seiring dengan cepat nya pemahaman masyarakat pada umumnya yang selalu menganalogikan robot adalah biang kerok hilangnya tenaga buruh untuk memacu pertumbuhan perekonomian.

Hal ini layaknya dua sisi perbedaan yang tidak akan bisa menyatu sama lain. Tapi bisa dicermati kembali, bila orang pelukis ternama akan tergusur karena kemampuan sebuah robot pelukis yang bisa membuat lukisan yang sama. Sebuah robot yang mampu untuk memahat patung yang hampir mirip pula. Seluruh ilustrasi tersebut

⁸ Arief Hermawan, *Jaringan Syaraf Tiruan (Teori Dan Aplikasi)* (Penerbit Andi, 2006).

memang sepintas robot bisa menguasai semua, tapi sangat disayangkan hasil kerja robot adalah tak lebih dari sebuah alat cetak dan seonggok besi aluminium dan komponen elektronika yang dirakit pada papan PCB. Sebuah lukisan dari Afandi tentunya akan bernilai ratusan juta beda ukuran dengan lukisan robot yang paling-paing laku di jual 10 ribuan di pinggir jalan.

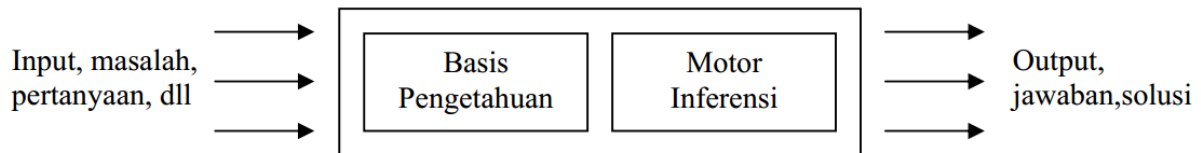
Istilah robot yang dahulu kala berjulukan Robota, tak lain adalah kata lain dari seorang buruh. Lain halnya dengan seorang manusia yang diciptakan se-sempurna mungkin oleh sang Pencipta. Sampai kapanpun robot adalah pembantu manusia. Bila sang teknokrat menciptakan robot untuk menjadi penguasa dunia, semoga saja dia tidak berumur panjang. Namun robot adalah sarana untuk membangun peradaban yang lebih maju dan memberikan kemudahan bagi manusia sebagai penciptanya. Dengan hasil demikian maka seluruh kajian tentang robotika menjadi lebih memasyarakat diseluruh elemen masyarakat. Dan buakan menjadi momok yang harus ditakuti.

Robot adalah simbol dari kemajuan dari sebuah teknologi, karena didalam nya mencakup seluruh elemen keilmuan. Elektronika, Mekanika, Mekatronika, Kinematika, Dinamika, dan lain sebagainya. Hal ini menjadi suatu alasan yang sangat tepat untuk mengash ilmu didalam nya. Ikon pendidikan akan menjadi semakin termasyur bila selalu mengutamakan teknologi didalam nya. Sebuah ikon ini sangat penting untuk membangun semangat kemajuan, karena hal ini akan menjadi sebuah patokan awal dari sebuah perjuangan untuk selalu dilanjutkan kepada generasi penerus.

BAB III

RUANG KEADAAN DAN TEKNIK Pencarian

Sistem yang menggunakan kecerdasan buatan akan memberikan output berupa solusi dari suatu masalah berdasarkan kumpulan pengetahuan yang ada.



Pada gambar, input yg diberikan pada sistem yg menggunakan kecerdasan buatan adalah berupa masalah. Sistem harus dilengkapi dengan sekumpulan pengetahuan yang ada pada basis pengetahuan. Sistem harus memiliki motor inferensi agar mampu mengambil kesimpulan berdasarkan fakta atau pengetahuan. Output yang diberikan berupa solusi masalah sebagai hasil dari inferensi.

Secara umum, untuk membangun suatu sistem yang mampu menyelesaikan masalah, perlu dipertimbangkan 4 hal :

- 1) Mendefinisikan masalah dengan tepat. Pendefinisian ini mencakup spesifikasi yang tepat mengenai keadaan awal dan solusi yang diharapkan.
- 2) Menganalisis masalah tersebut serta mencari beberapa teknik penyelesaian masalah yang sesuai.
- 3) Merepresentasikan pengetahuan yang perlu untuk menyelesaikan masalah tersebut.
- 4) Memilih teknik penyelesaian masalah yang terbaik⁹

3.1. Mendefinisikan Masalah Sebagai Suatu Ruang Keadaan

Misalkan permasalahan yang dihadapi adalah permainan catur, maka harus ditentukan :

- 1) posisi awal pada papan catur

posisi awal setiap permainan catur selalu sama, yaitu semua bidak diletakkan di atas papan catur dalam 2 sisi, yaitu kubu putih dan kubu hitam.

⁹ Akhmad Hidayanto, *Knowledge Management* (Departemen Teknik Industri Universitas Indonesia, 2006).

2) aturan – aturan untuk melakukan gerakan

aturan – aturan ini sangat berguna untuk menentukan gerakan suatu bidak, yaitu melangkah dari satu keadaan ke keadaan lain. Misalkan untuk mempermudah menunjukkan posisi bidak, setiap kotak ditunjukkan dalam huruf (a,b,c,d,e,f,g,h) pada arah horisontal dan angka (1,2,3,4,5,6,7,8) pada arah vertikal. Suatu aturan

```
if bidak putih pada kotak(e,2),
    and kotak(e,3) kosong,
    and kotak(e,4) kosong
then gerakkan bidak dari (e,2) ke (e,4)
```

untuk menggerakkan bidak dari posisi (e,2) ke (e,4) dapat ditunjukkan dengan aturan:

3) tujuan (goal)

tujuan yang ingin dicapai adalah posisi pada papan catur yang menunjukkan kemenangan seseorang terhadap lawannya. Kemenangan ini ditandai dengan posisi raja yang sudah tidak dapat bergerak lagi.

Contoh tersebut menunjukkan representasi masalah dalam Ruang Keadaan (State Space), yaitu suatu ruang yang berisi semua keadaan yang mungkin. Kita dapat memulai bermain catur dengan menempatkan diri pada keadaan awal, kemudian bergerak dari satu keadaan ke keadaan yang lain sesuai dengan aturan yang ada, dan mengakhiri permainan jika salah satu telah mencapai tujuan.

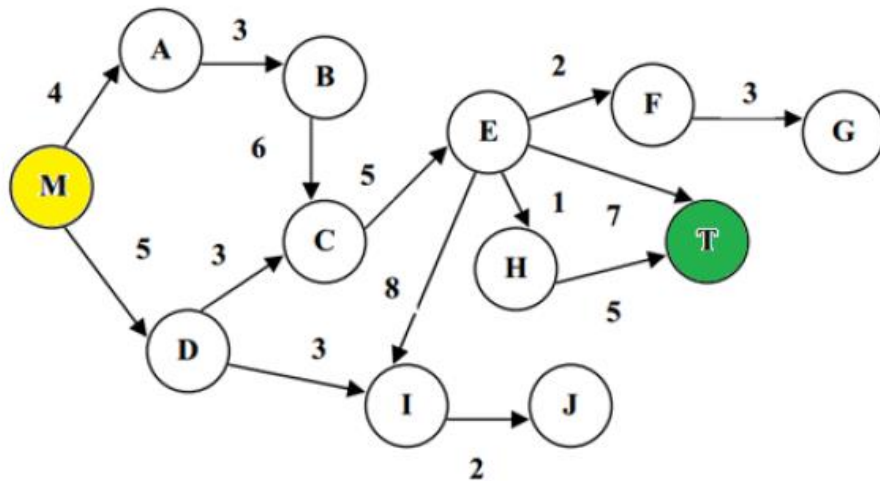
Jadi untuk mendeskripsikan masalah dengan baik harus :

- 1) Mendefinisikan suatu ruang keadaan (state space)
- 2) Menetapkan satu atau lebih keadaan awal (initial state)
- 3) Menetapkan satu atau lebih tujuan (goal state)
- 4) Menetapkan kumpulan aturan

Ada beberapa cara untuk merepresentasikan Ruang Keadaan, antara lain :

1) Graph Keadaan

Graph terdiri dari node-node yang menunjukkan keadaan yaitu keadaan awal dan keadaan baru yang akan dicapai dengan menggunakan operator. Node-node dalam graph keadaan saling dihubungkan dengan menggunakan arc (busur) yang diberi panah untuk menunjukkan arah dari suatu keadaan ke keadaan berikutnya.



Graph keadaan dengan node M menunjukkan keadaan awal, node T adalah tujuan. Ada 4 lintasan dari M ke T :

M-A-B-C-E-T

M-A-B-C-E-H-T

M-D-C-E-T

M-D-C-E-H-T

Lintasan buntu atau lintasan yang tidak sampai ke tujuan :

M-A-B-C-E-F-G

M-A-B-C-E-I-J

M-D-C-E-F-G

M-D-C-E-I-J

M-D-I-J

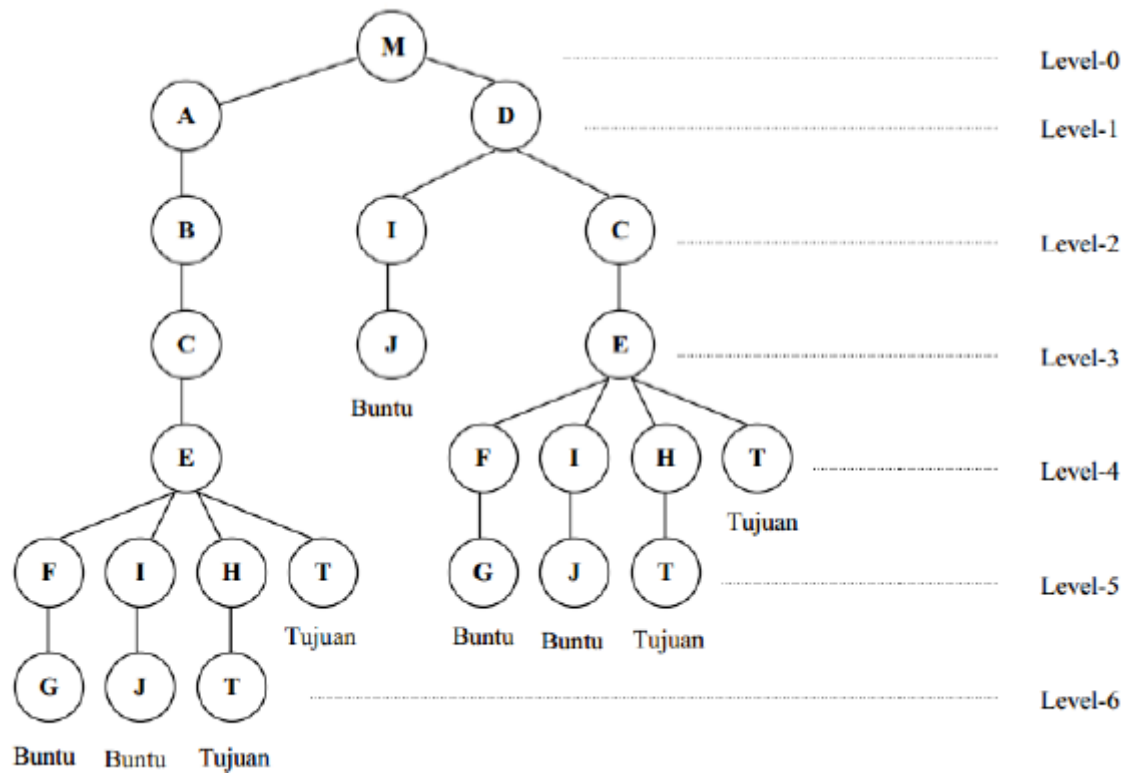
2) Pohon Pelacakan / Pencarian

Struktur pohon digunakan untuk menggambarkan keadaan secara hirarkis. Node yg terletak pada level- 0 disebut 'akar'.

Node akar : menunjukkan keadaan awal & memiliki beberapa percabangan yang terdiri atas beberapa node yg disebut 'anak' .

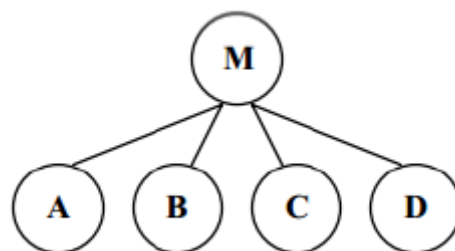
Node-node yg tidak memiliki anak disebut 'daun' menunjukkan akhir dari suatu pencarian, dapat berupa tujuan yang diharapkan (goal) atau jalan buntu (dead end).

Gambar berikut menunjukkan pohon pencarian untuk graph keadaan dengan 6 level.

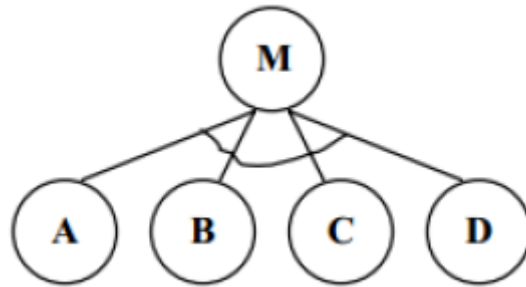


3) Pohon And/Or

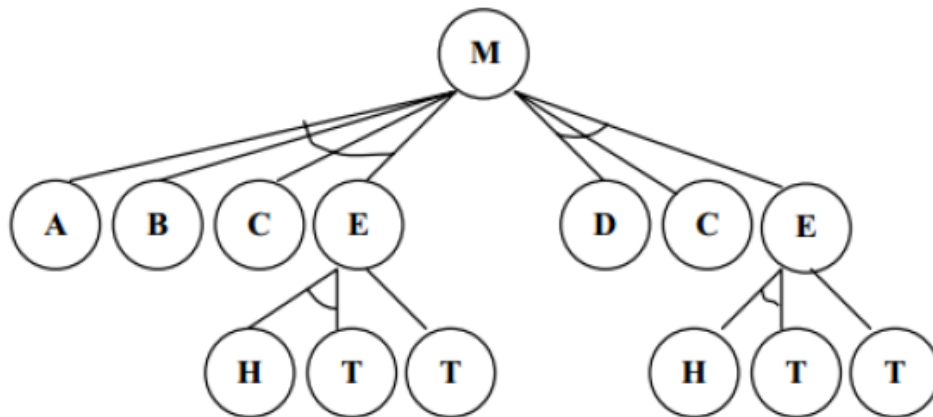
Masalah M dicari solusinya dengan 4 kemungkinan yaitu A OR B OR C OR D.



Masalah M hanya dapat diselesaikan dengan A AND B AND C AND D



Contoh : Dengan menggunakan pohon AND/OR tujuan yang dicapai pada pohon di Gambar sebelumnya bisa dipersingkat hanya sampai level-2 saja.



Contoh 1 : Masalah EMBER

Ada 2 ember masing-masing berkapasitas 4 galon (ember A) dan 3 galon (ember B). Ada pompa air yg akan digunakan untuk mengisi air pada ember tersebut. Bagaimana dapat mengisi tepat 2 galon air ke dalam ember berkapasitas 4 galon?

Penyelesaian :

1. Identifikasi ruang keadaan (state space)

Permasalahan ini dapat digambarkan sebagai himpunan pasangan bilangan bulat :

- ✓ x = jumlah air yg diisikan ke ember 4 galon (ember A)
- ✓ y = jumlah air yg diisikan ke ember 3 galon (ember B)

Ruang keadaan = (x,y) sedemikian hingga $x \in \{0,1,2,3,4\}$ dan $y \in \{0,1,2,3\}$

2. Keadaan awal & tujuan

Keadaan awal : kedua ember kosong = $(0,0)$

Tujuan : ember 4 galon berisi 2 galon air = $(2,n)$ dengan sembarang n

3. Keadaan ember

Keadaan ember bisa digambarkan sebagai berikut :

| Keadaan awal | Tujuan ∇ | | | | |
|--------------|-----------------|-------|-------|-------|--|
| (0,0) | (1,0) | (2,0) | (3,0) | (4,0) | |
| (0,1) | (1,1) | (2,1) | (3,1) | (4,1) | |
| (0,2) | (1,2) | (2,2) | (3,2) | (4,2) | |
| (0,3) | (1,3) | (2,3) | (3,3) | (4,3) | |

4. Aturan-aturan

Diasumsikan kita dapat mengisi ember air itu daripompa air, membuang air dari ember ke luar, menuangkan air dari ember yang satu ke ember yang lain. Kita buat beberapa aturan-aturan yang dapat digambarkan sebagai berikut :

| Aturan ke- | Jika | Maka |
|------------|-------------------------------------|---|
| 1 | (x,y) $x < 4$ | $(4,y)$ Isi ember A |
| 2 | (x,y) $y < 3$ | $(x,3)$ Isi ember B |
| 3 | (x,y) $x > 0$ | $(x - d,y)$ Tuang sebagian air keluar dari ember A |
| 4 | (x,y) $y > 0$ | $(x,y - d)$ Tuang sebagian air keluar dari ember B |
| 5 | (x,y) $x > 0$ | $(0,y)$ Kosongkan ember A dengan membuang airnya |
| 6 | (x,y) $y > 0$ | $(x,0)$ Kosongkan ember B dengan membuang airnya |
| 7 | (x,y) $x+y \geq 4$ dan $y > 0$ | $(4,y - (4 - x))$ Tuang air dari ember B ke ember A sampai ember A penuh |
| 8 | (x,y) $x+y \geq 3$ dan $x > 0$ | $(x - (3 - y),3)$ Tuang air dari ember A ke ember B sampai ember B penuh |
| 9 | (x,y) $x+y \leq 4$ dan $y > 0$ | $(x+y,0)$ Tuang seluruh air dari ember B ke ember A |
| 10 | (x,y) $x+y \leq 3$ dan $x > 0$ | $(0,x+y)$ Tuang seluruh air dari ember A ke ember B |
| 11 | $(0,2)$ | $(2,0)$ Tuang 2 galon air dari ember B ke ember A |

5. Representasi ruang keadaan dengan pohon pelacakan

Pencarian suatu solusi dapat dilukiskan dengan menggunakan pohon. Tiap-tiap node menunjukkan satu keadaan. Jalur dari parent ke child, menunjukkan

1 operasi. Tiap node memiliki node child yg menunjukkan keadaan yg dapat dicapai oleh parent.

Solusi yg ditemukan :

Solusi 1

| Isi ember A | Isi ember B | Aturan yg dipakai |
|-------------|-------------|-------------------|
| 0 | 0 | 1 |
| 4 | 0 | 8 |
| 1 | 3 | 6 |
| 1 | 0 | 10 |
| 0 | 1 | 1 |
| 4 | 1 | 8 |
| 2 | 3 | Solusi |

Solusi 2

| Isi ember A | Isi ember B | Aturan yg dipakai |
|-------------|-------------|-------------------|
| 0 | 0 | 2 |
| 0 | 3 | 9 |
| 3 | 0 | 2 |
| 3 | 3 | 7 |
| 4 | 2 | 5 |
| 0 | 2 | 9 |
| 2 | 0 | Solusi |

Representasi ruang keadaan untuk kasus EMBER



Contoh 2: Masalah PETANI,KAMBING,SERIGALA,SAYURAN,PERAHU

Seorang petani akan menyeberangkan seekor kambing,seekor serigala,sayuran dengan sebuah perahu yg melalui sungai. Perahu hanya bisa memuat petani & satu penumpang yg lain (kambing, serigala, atau sayuran). Jika ditinggalkanpetani tersebut, maka sayuran dimakan kambing dan kambing akan dimakan serigala.

Penyelesaian :

1. Identifikasi ruang keadaan

Permasalahan ini dapat dilambangkan dengan (jumlah kambing,jumlah serigala,jumlah sayuran,jumlah perahu).

Contoh : daerah asal (0,1,1,1) = daerah asal tidak ada kambing,ada serigala, ada sayuran,ada perahu

2. Keadaan awal & tujuan

Keadaan awal, pada kedua daerah :

daerah asal = (1,1,1,1)

daerah seberang = (0,0,0,0)

Keadaan tujuan, pada kedua daerah :

daerah asal = (0,0,0,0)

daerah seberang = (1,1,1,1)

3. Aturan-aturan

| Aturan ke- | Aturan |
|------------|----------------------|
| 1 | Kambing menyeberang |
| 2 | Sayuran menyeberang |
| 3 | Serigala menyeberang |
| 4 | Kambing kembali |
| 5 | Sayuran kembali |
| 6 | Serigala kembali |
| 7 | Perahu kembali |

4. Solusi yg ditemukan

| Daerah asal | Daerah seberang | Aturan yg dipakai |
|-------------|-----------------|-------------------|
| (1,1,1,1) | (0,0,0,0) | 1 |
| (0,1,1,0) | (1,0,0,1) | 7 |
| (0,1,1,1) | (1,0,0,0) | 3 |
| (0,0,1,0) | (1,1,0,1) | 4 |
| (1,0,1,1) | (0,1,0,0) | 2 |
| (1,0,0,0) | (0,1,1,1) | 7 |
| (1,0,0,1) | (0,1,1,0) | 1 |
| (0,0,0,0) | (1,1,1,1) | Solusi |

3.2. Metode Pelacakan/Pencarian

Hal penting dalam menentukan keberhasilan sistem cerdas adalah kesuksesan dalam pencarian. Pencarian = suatu proses mencari solusi dari suatu permasalahan

melalui sekumpulan kemungkinan ruang keadaan (state space). Ruang keadaan = merupakan suatu ruang yang berisi semua keadaan yang mungkin.

Untuk mengukur performansi metode pencarian, terdapat empat kriteria yang dapat digunakan :

- ✓ Completeness : apakah metode tersebut menjamin penemuan solusi jika solusinya memang ada?
- ✓ Time complexity : berapa lama waktu yang diperlukan?
- ✓ Space complexity : berapa banyak memori yang diperlukan
- ✓ Optimality : apakah metode tersebut menjamin menemukan solusi yang terbaik jika terdapat beberapa solusi berbeda?

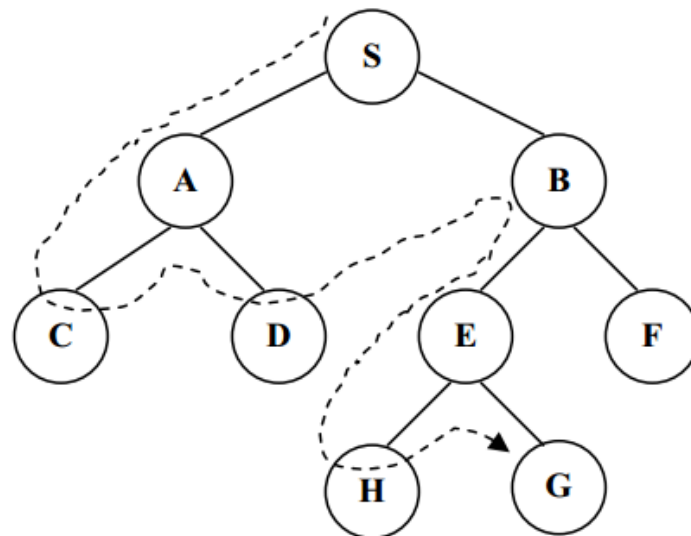
Teknik pencarian :

- Pencarian buta (blind search) : tidak ada informasi awal yang digunakan dalam proses pencarian
 - 1) Pencarian melebar pertama (Breadth – First Search)
 - 2) Pencarian mendalam pertama (Depth – First Search)
- Pencarian terbimbing (heuristic search) : adanya informasi awal yang digunakan dalam proses pencarian
 - 1) Pendakian Bukit (Hill Climbing)
 - 2) Pencarian Terbaik Pertama (Best First Search)

A. Pencarian Buta (blind search)

- 1) Breadth – First Search

Semua node pada level n akan dikunjungi terlebih dahulu sebelum mengunjungi node-node pada level $n+1$. Pencarian dimulai dari node akar terus ke level 1 dari kiri ke kanan, kemudian berpindah ke level berikutnya dari kiri ke kanan hingga solusi ditemukan.



Keuntungan :

- ✓ tidak akan menemui jalan buntu, menjamin ditemukannya solusi (jika solusinya memang ada) dan solusi yang ditemukan pasti paling baik
- ✓ jika ada 1 solusi, maka breadth – first search akan menemukannya,
- ✓ jika ada lebih dari 1 solusi, maka solusi minimum akan ditemukan.
- ✓ Kesimpulan : completeness dan optimal

Kelemahan :

- ✓ membutuhkan memori yang banyak, karena harus menyimpan semua simpul yang pernah dibangkitkan. Hal ini harus dilakukan agar BFS dapat melakukan penelusuran simpul simpul sampai di level bawah
- ✓ membutuhkan waktu yang cukup lama

2) Depth – First Search

Pencarian dilakukan pada suatu simpul dalam setiap level dari yang paling kiri. Jika pada level yang paling dalam tidak ditemukan solusi, maka pencarian dilanjutkan pada simpul sebelah kanan dan simpul yang kiri dapat dihapus dari memori. Jika pada level yang paling dalam tidak ditemukan solusi, maka pencarian dilanjutkan pada level sebelumnya. Demikian seterusnya sampai ditemukan solusi.

Keuntungan :

- ✓ membutuhkan memori relatif kecil, karena hanya node-node padalintasan yang aktif saja yang disimpan
- ✓ Secara kebetulan, akan menemukan solusi tanpaharus menguji lebih banyak lagi dalam ruang keadaan, jadi jika solusi yang dicari berada pada level yang dalam dan paling kiri, maka DFS akan menemukannya dengan cepat □ waktu cepat

Kelemahan :

- ✓ Memungkinkan tidak ditemukannya tujuanyang diharapkan, karena jika pohon yang dibangkitkan mempunyai level yang sangat dalam (tak terhingga) □ tidak complete karena tidak ada jaminan menemukan solusi
- ✓ Hanya mendapat 1 solusi pada setiap pencarian, karena jika terdapat lebih dari satu solusi yang samatetapi berada pada level yang berbeda, maka DFS tidak menjamin untuk menemukan solusi yang paling baik □ tidak optimal.

B. Heuristic Search

Pencarian buta tidak selalu dapat diterapkan dengan baik, hal ini disebabkan waktu aksesnya yang cukup lama & besarnya memori yang diperlukan. Untuk masalah dengan ruang masalah yang besar, teknik pencarian buta bukan metode yang baik karena keterbatasan kecepatan komputer dan memori.

Metode heuristic search diharapkan bisa menyelesaikan permasalahan yang lebih besar.

Metode heuristic search menggunakan suatu fungsi yang menghitung biaya perkiraan (estimasi) dari suatu simpul tertentu menuju ke simpul tujuan □ disebut fungsi heuristic. Aplikasi yang menggunakan fungsi heuristic : Google, Deep Blue Chess Machine

Misal kasus 8-puzzle. Ada 4 operator yang dapat digunakan untuk menggerakkan dari satu keadaan ke keadaan yang baru.

1. Ubin kosong digeser ke kiri
2. Ubin kosong digeser ke kanan
3. Ubin kosong digeser ke bawah
4. Ubin kosong digeser ke atas

Langkah awal

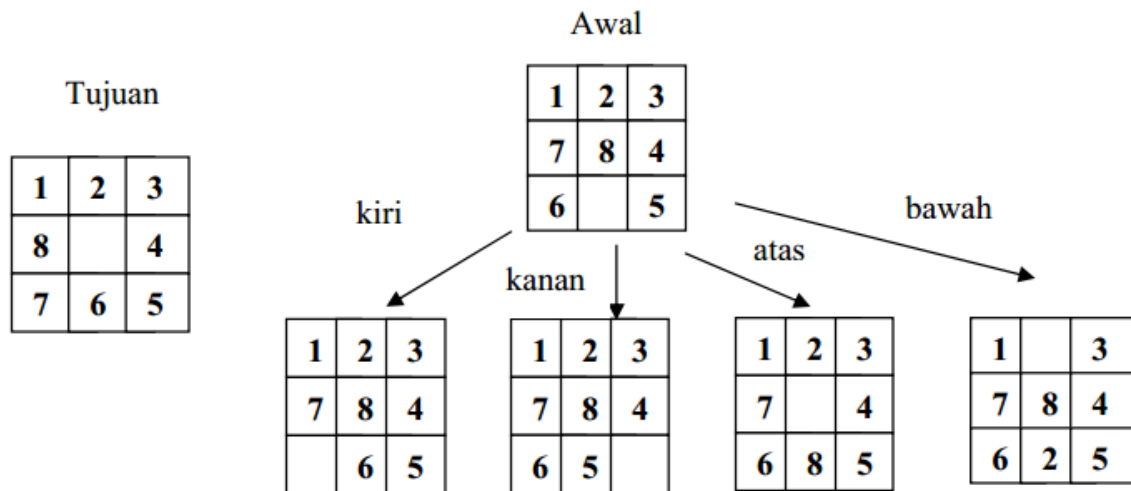
Keadaan awal

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 7 | 8 | 4 |
| 6 | | 5 |



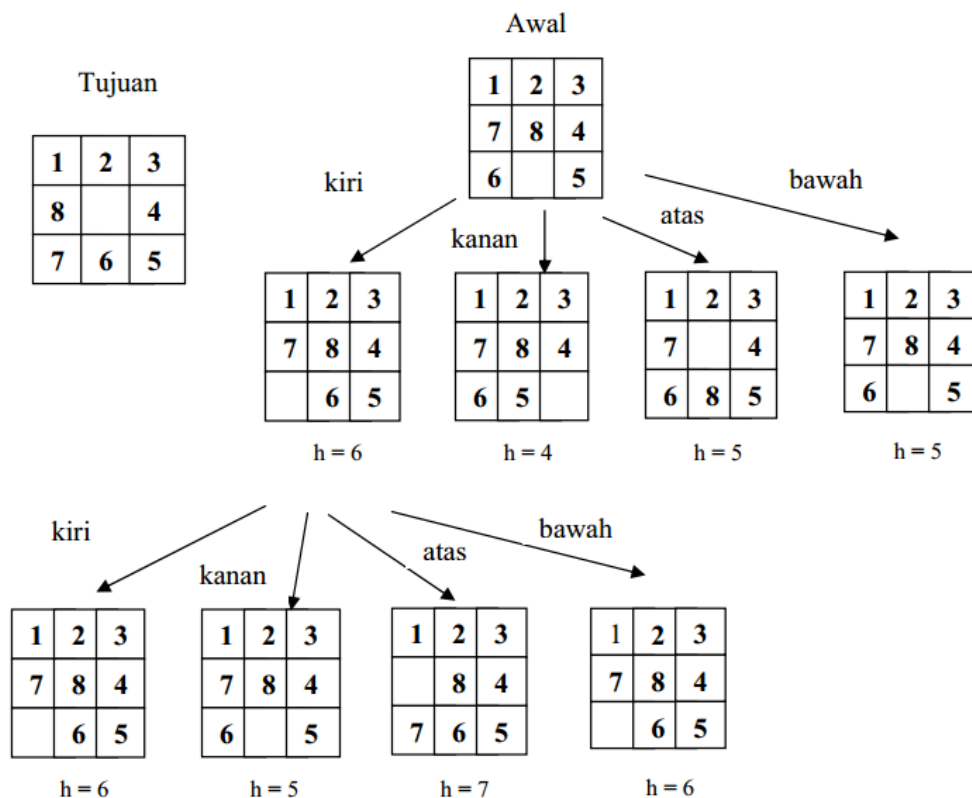
Tujuan

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 8 | | 4 |
| 7 | 6 | 5 |

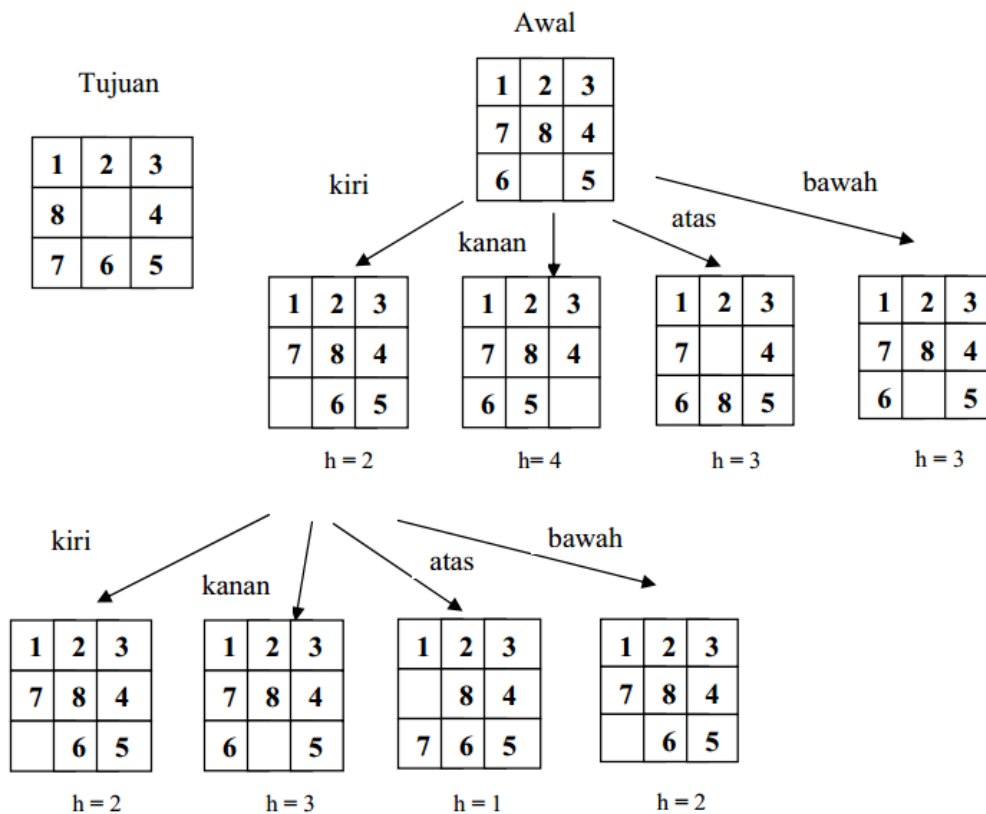


Pada pencarian heuristik perlu diberikan informasi khusus, yaitu :

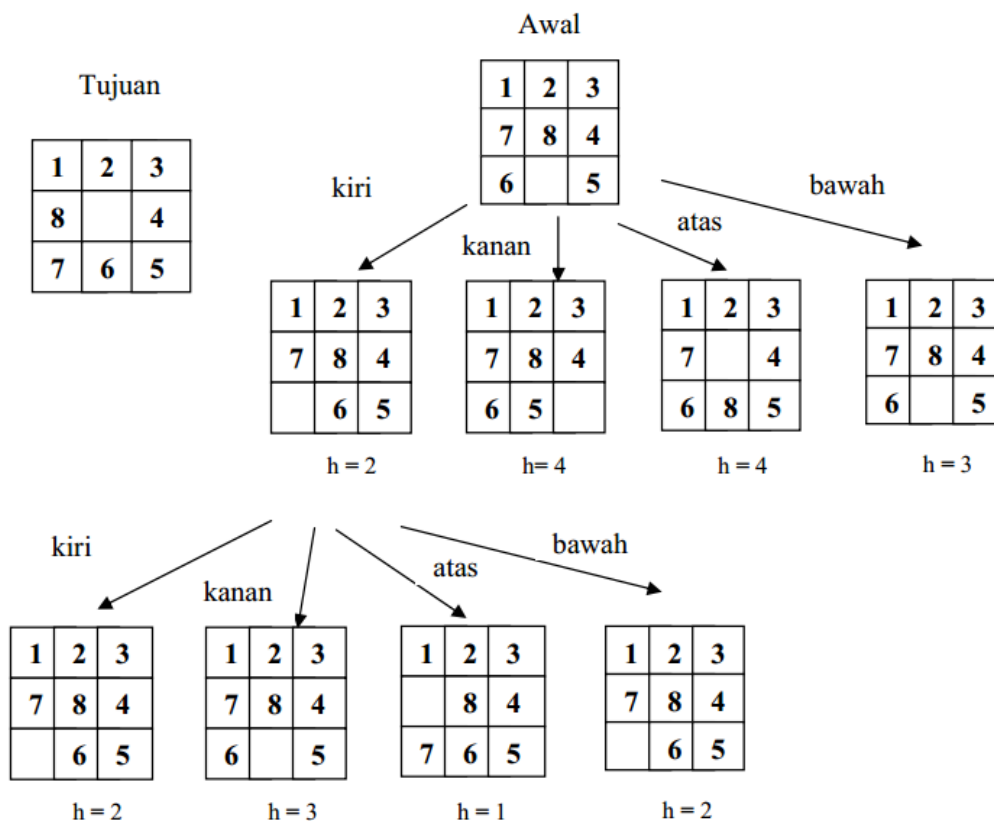
- Untuk jumlah ubin yang menempati posisi yang benar Jumlah yang lebih tinggi adalah yang lebih diharapkan (lebih baik)



- Untuk jumlah ubin yang menempati posisi yang salah Jumlah yang lebih kecil adalah yang diharapkan (lebih baik)



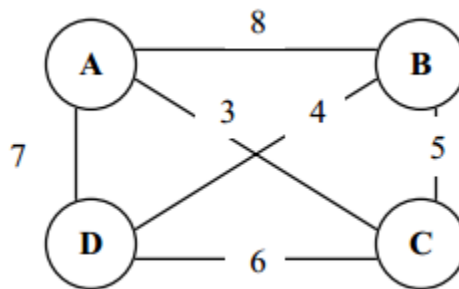
- Menghitung total gerakan yang diperlukan untuk mencapai tujuan Jumlah yang lebih kecil adalah yang diharapkan (lebih baik)



1) Hill Climbing

Contoh : Traveling Salesman Problem (TSP)

Seorang salesman ingin mengunjungi n kota. Jarak antara tiap-tiap kota sudah diketahui. Kita ingin mengetahui rute terpendek dimana setiap kota hanya boleh dikunjungi tepat 1 kali. Misal ada 4 kota dengan jarak antara tiap-tiap kota seperti berikut ini :



Solusi-solusi yang mungkin dengan menyusun kota-kota dalam urutan abjad, misal :

A – B – C – D : dengan panjang lintasan (=19)

A – B – D – C : (=18)

A – C – B – D : (=12)

A – C – D – B : (=13)

dst

a. Metode simple hill climbing

Ruang keadaan berisi semua kemungkinan lintasan yang mungkin. Operator digunakan untuk menukar posisi kota-kota yang bersebelahan. Fungsi heuristik yang digunakan adalah panjang lintasan yang terjadi. Operator yang akan digunakan adalah menukar urutan posisi 2 kota dalam 1 lintasan. Bila ada n kota, dan ingin mencari kombinasi lintasan dengan menukar posisi urutan 2 kota, maka akan didapat sebanyak :

$$\frac{n!}{2!(n-2)!} = \frac{4!}{2!(4-2)!} = 6 \text{ kombinasi}$$

Keenam kombinasi ini akan dipakai sebagai operator, yaitu :

Tukar 1,2 = menukar urutan posisi kota ke – 1 dengan kota ke – 2

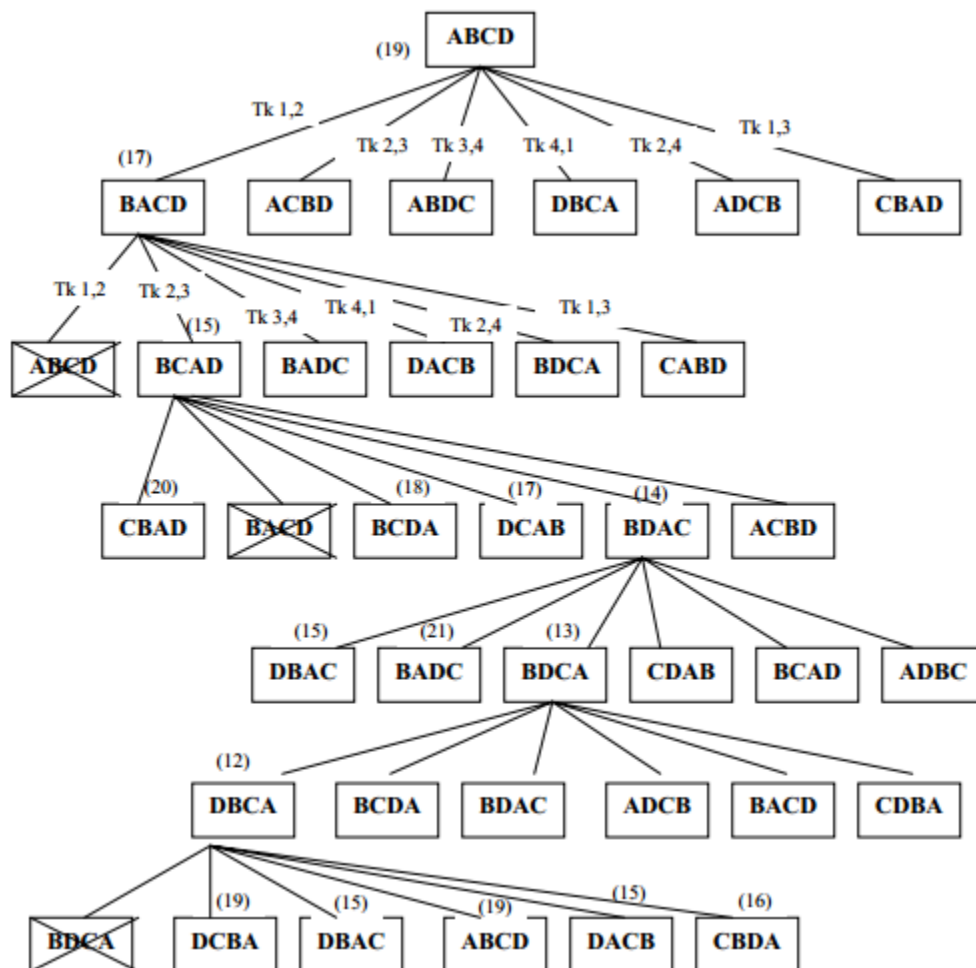
Tukar 2,3 = menukar urutan posisi kota ke – 2 dengan kota ke – 3

Tukar 3,4 = menukar urutan posisi kota ke – 3 dengan kota ke – 4

Tukar 4,1 = menukar urutan posisi kota ke – 4 dengan kota ke – 1

Tukar 2,4 = menukar urutan posisi kota ke – 2 dengan kota ke – 4

Tukar 1,3 = menukar urutan posisi kota ke – 1 dengan kota ke – 3



Keadaan awal, lintasan ABCD (=19).

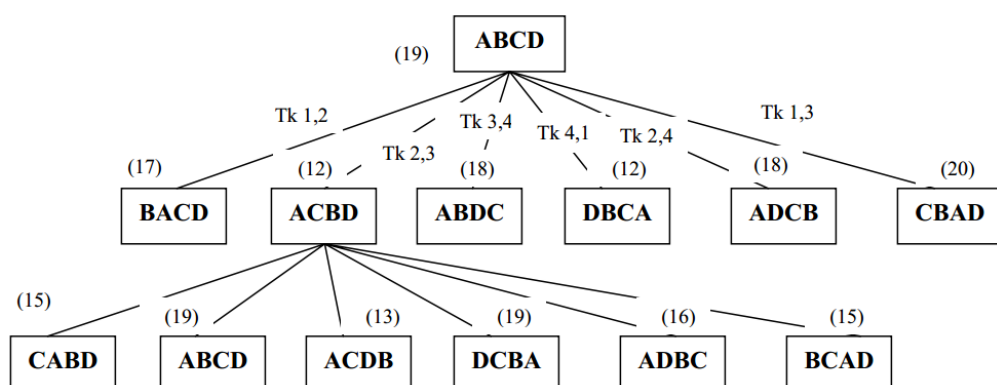
- ❖ Level pertama, hill climbing mengunjungi BACD (=17), BACD (=17) < ABCD (=19), sehingga BACD menjadi pilihan selanjutnya dengan operator Tukar 1,2

- ❖ Level kedua, mengunjungi ABCD, karena operator Tukar 1,2 sudah dipakai BACD, maka pilih node lain yaitu BCAD (=15), BCAD (=15) < BACD (=17)
- ❖ Level ketiga, mengunjungi CBAD (=20), CBAD (=20) > BCAD (=15), maka pilih node lain yaitu BCDA (=18), pilih node lain yaitu DCAB (=17), pilih node lain yaitu BDAC (=14), BDAC (=14) < BCAD (=15)
- ❖ Level keempat, mengunjungi DBAC (=15), DBAC(=15)> BDAC (=14), maka pilih node lain yaitu BADC (=21), pilih node lain yaitu BDCA (=13), BDCA (=13) < BDAC (=14)
- ❖ Level kelima, mengunjungi DBCA (=12), DBCA (=12) < BDCA (=13)
- ❖ Level keenam, mengunjungi BDCA, karena operator Tukar 1,2 sudah dipakai DBCA, maka pilih node lain yaitu DCBA, pilih DBAC, pilih ABCD, pilih DACB, pilih CBDA

Karena sudah tidak ada node yang memiliki nilai heuristik yang lebih kecil dibanding nilai heuristik DBCA, maka node DBCA(=12) adalah lintasan terpendek (SOLUSI)

b. Metode steepest – ascent hill climbing

Steepest – ascent hill climbing hampir samadengan simple – ascent hill climbing, hanya saja gerakan pencarian tidak dimulai dari kiri, tetapi berdasarkan nilai heuristik terbaik.



Keadaan awal, lintasan ABCD (=19).

- ❖ Level pertama, hill climbing memilih nilai heuristik terbaik yaitu ACBD (=12) sehingga ACBD menjadi pilihan selanjutnya.

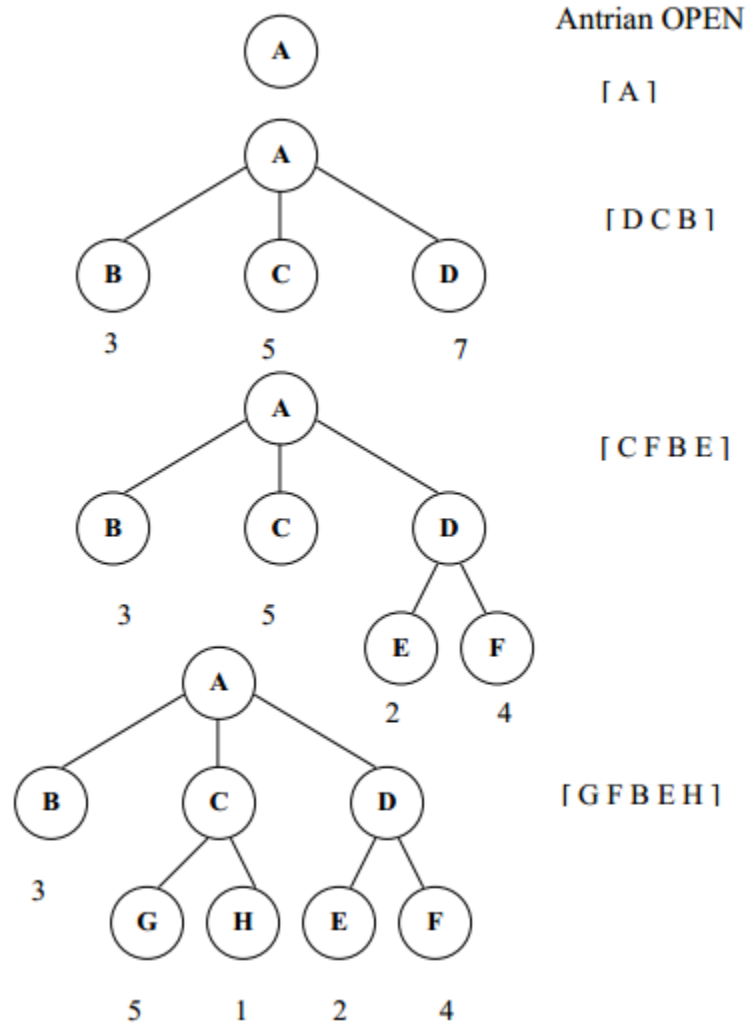
- ❖ Level kedua, hill climbing memilih nilai heuristik terbaik, karena nilai heuristik lebih besar dibanding ACBD, maka hasil yang diperoleh lintasannya tetap ACBD (=12)

2) Best First Search

Metode best first search merupakan kombinasi dari metode depth first search dan breadth first search dengan mengambil kelebihan dari kedua metode tersebut. Hill climbing tidak diperbolehkan untuk kembali ke node pada lebih rendah meskipun node tersebut memiliki nilai heuristik lebih baik. Pada best first search, pencarian diperbolehkan mengunjungi node di lebih rendah, jika ternyata node di level lebih tinggi memiliki nilai heuristik lebih buruk. Untuk mengimplementasikan metode ini, dibutuhkan 2 antrian yang berisi node-node, yaitu :

OPEN: berisi node-node yang sudah dibangkitkan, sudah memiliki fungsi heuristik namun belum diuji. Umumnya berupa antrian berprioritas yang berisi elemen-elemen dengan nilai heuristik tertinggi.

CLOSED: berisi node-node yang sudah diuji.



KETERANGAN

Diasumsikan node dengan nilai yang lebih besar memiliki nilai evaluasi yang lebih baik. Pada keadaan awal, antrian berisi A. Pengujian dilakukan di level pertama, node D memiliki nilai terbaik, sehingga menempati antrian pertama, disusul dengan C dan B. Node D memiliki cabang E dan F yang masing-masing bernilai 2 & 4. Dengan demikian C merupakan pilihan terbaik dengan menempati antrian pertama. Demikian seterusnya.¹⁰

¹⁰ Kusumadewi, *Artificial Intelligence (Teknik Dan Aplikasinya)*.

BAB IV

REPRESENTASI PENGETAHUAN

Dua bagian dasar sistem kecerdasan buatan (menurut Turban) :

1) Basis pengetahuan :

Berisi fakta tentang objek-objek dalam domain yang dipilih dan hubungan diantara domain-domain tersebut

2) Inference Engine :

Merupakan sekumpulan prosedur yang digunakan untuk menguji basis pengetahuan dalam menjawab suatu pertanyaan, menyelesaikan masalah, atau membuat keputusan.

Basis pengetahuan berisi struktur data yang dapat dimanipulasi oleh suatu sistem inferensi yang menggunakan pencarian dan teknik pencocokan pola pada basis pengetahuan yang bermanfaat untuk menjawab pertanyaan, menggambarkan kesimpulan atau bentuk lainnya sebagai suatu fungsi kecerdasan

Karakteristik representasi pengetahuan

- ✓ Dapat deprogram dengan bahasa komputer dan disimpan dalam memori
- ✓ Fakta dan pengetahuan lain yang terkandung didalamnya dapat digunakan untuk melakukan penalaran

Dalam menyelesaikan masalah harus dibutuhkan pengetahuan yang cukup dan sistem juga harus memiliki kemampuan untuk menalar. Basis pengetahuan dan kemampuan untuk melakukan penalaran merupakan bagian terpenting dari sistem yang menggunakan kecerdasan buatan.¹¹

4.1. Logika

Logika adalah bentuk representasi pengetahuan yang paling tua. Proses logika adalah proses membentuk kesimpulan atau menarik suatu inferensi berdasarkan fakta yang telah ada. Input dari proses logika berupa premis atau fakta-fakta yang diakui

¹¹ Turban, *Decision Support and Expert Systems; Management Support Systems*.

kebenarannya sehingga dengan melakukan penalaran pada proses logika dapat dibentuk suatu inferensi atau kesimpulan yang benar juga.



Ada 2 penalaran yang dapat dilakukan untuk mendapat konklusi :

- 1) Penalaran deduktif : dimulai dari prinsip umum untuk mendapatkan konklusi yang lebih khusus.

Contoh :

Premis mayor : Jika hujan turun saya tidak akan berangkat kuliah

Premis minor : Hari ini hujan turun

Konklusi : Hari ini saya tidak akan berangkat kuliah

- 2) Penalaran induktif : dimulai dari fakta-fakta khusus untuk mendapatkan kesimpulan umum.

Contoh :

Premis -1 : Aljabar adalah pelajaran yang sulit

Premis -2 : Geometri adalah pelajaran yang sulit

Premis -3 : Kalkulus adalah pelajaran yang sulit

Konklusi : Matematika adalah pelajaran yang sulit

Munculnya premis baru bisa mengakibatkan gugurnya konklusi yang sudah diperoleh, misal : Premis -4 : Kinematika adalah pelajaran yang sulit. Premis tersebut menyebabkan konklusi: "Matematika adalah pelajaran yang sulit", menjadi salah, karena Kinematika bukan merupakan bagian dari Matematika, sehingga bila menggunakan penalaran induktif sangat dimungkinkan adanya ketidakpastian.

4.1.1. Logika Proposisi

Proposisi adalah suatu pernyataan yang dapat bernilai Benar atau Salah. Simbol-simbol seperti P dan Q menunjukkan proposisi. Dua atau lebih proposisi dapat digabungkan dengan menggunakan operator logika :

- Konjungsi : \wedge (and)
- Disjungsi : \vee (or)
- Negasi : \neg (not)
- Implikasi : \rightarrow (if then)
- Ekuivalensi : \leftrightarrow (if and only if)

Not

| P | not P |
|---|-------|
| B | S |
| S | B |

And, Or, If – Then, If – and – only – if

| P | Q | P and Q | P or Q | if P then Q | P if and only if Q |
|---|---|---------|--------|-------------|--------------------|
| B | B | B | B | B | B |
| B | S | S | B | S | S |
| S | B | S | B | B | S |
| S | S | S | S | B | B |

Untuk melakukan inferensi pada logika proposisi dapat dilakukan dengan menggunakan resolusi. Resolusi adalah suatu aturan untuk melakukan inferensi yang dapat berjalan secara efisien dalam suatu bentuk khusus yaitu conjunctive normal form(CNF), ciri – cirinya :

- setiap kalimat merupakan disjungsi literal
- semua kalimat terkonjungsi secara implisit

Langkah-langkah untuk mengubah suatu kalimat (konversi) ke bentuk CNF :

- Hilangkan implikasi dan ekuivalensi
 - ✓ $x \rightarrow y$ menjadi $\neg x \vee y$
 - ✓ $x \leftrightarrow y$ menjadi $(\neg x \vee y) \wedge (\neg y \vee x)$
- Kurangi lingkup semua negasi menjadi satu negasi saja
 - ✓ $\neg(\neg x)$ menjadi x
 - ✓ $\neg(x \vee y)$ menjadi $(\neg x \wedge \neg y)$
 - ✓ $\neg(x \wedge y)$ menjadi $(\neg x \vee \neg y)$
- Gunakan aturan asosiatif dan distributif untuk mengkonversi menjadi conjunction of disjunction
 - ✓ Asosiatif : $(A \vee B) \vee C$ menjadi $A \vee (B \vee C)$

✓ Distributif : $(A \wedge B) \vee C$ menjadi $(A \vee C) \wedge (B \vee C)$

➤ Buat satu kalimat terpisah untuk tiap-tiap konjungsi

Contoh :

Diketahui basis pengetahuan (fakta-fakta yang bernilai benar) sebagai berikut :

- 1) P
- 2) $(P \wedge Q) \rightarrow R$
- 3) $(S \vee T) \rightarrow Q$
- 4) T

Tentukan kebenaran R.

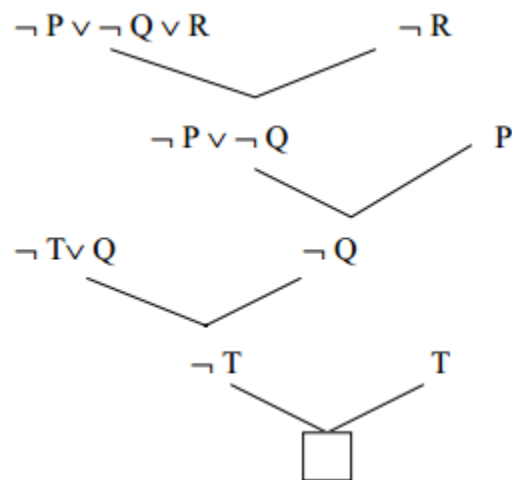
Untuk membuktikan kebenaran R dengan menggunakan resolusi, maka ubah dulu menjadi bentuk CNF.

| Kalimat | Langkah-langkah | CNF |
|---------------------------------|---|--|
| 1. P | Sudah merupakan bentuk CNF | P |
| 2. $(P \wedge Q) \rightarrow R$ | <ul style="list-style-type: none"> ■ Menghilangkan implikasi : $\neg(P \wedge Q) \vee R$ ■ Mengurangi lingkup negasi : $(\neg P \vee \neg Q) \vee R$ ■ Gunakan asosiatif : $\neg P \vee \neg Q \vee R$ | $\neg P \vee \neg Q \vee R$ |
| 3. $(S \vee T) \rightarrow Q$ | <ul style="list-style-type: none"> ■ Menghilangkan implikasi : $\neg(S \vee T) \vee Q$ ■ Mengurangi lingkup negasi : $(\neg S \wedge \neg T) \vee Q$ ■ Gunakan distributif : $(\neg S \vee Q) \wedge (\neg T \vee Q)$ | $(\neg S \vee Q)$ $(\neg T \vee Q)$ |
| 4. T | Sudah merupakan bentuk CNF | T |

Kemudian kita tambahkan kontradiksi pada tujuannya, R menjadi $\neg R$ sehingga fakta-fakta (dalam bentuk CNF) dapat disusun menjadi :

- 1) P
- 2) $\neg P \vee \neg Q \vee R$
- 3) $\neg S \vee Q$
- 4) $\neg T \vee Q$
- 5) T
- 6) $\neg R$

Sehingga resolusi dapat dilakukan untuk membuktikan kebenaran R, sebagai berikut :



Contoh bila diterapkan dalam kalimat:

- P : Ani anak yang cerdas
- Q : Ani rajin belajar
- R : Ani akan menjadi juara kelas
- S : Ani makannya banyak
- T : Ani istirahatnya cukup

Kalimat yang terbentuk :

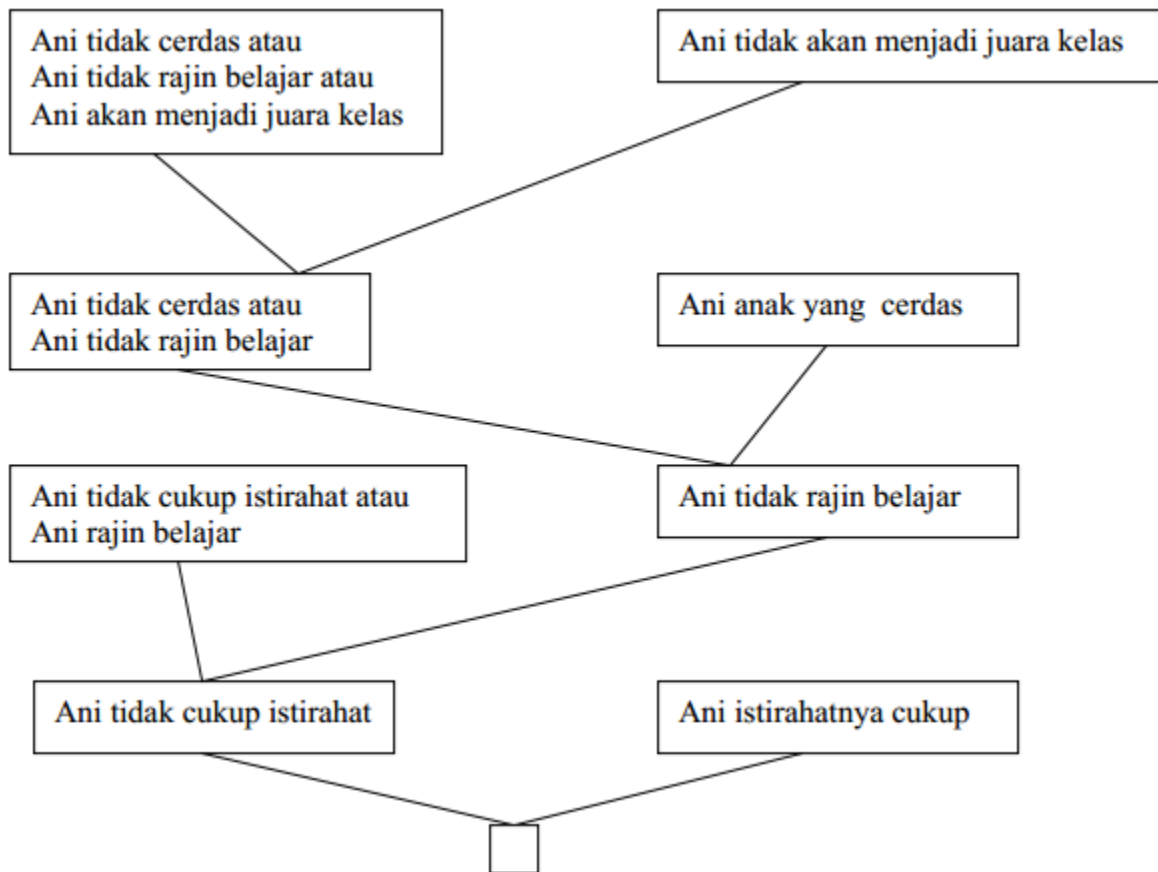
- Ani anak yang cerdas
- Jika ani anak yang cerdas dan ani rajin belajar, maka ani akan menjadi juara kelas
- Jika ani makannya banyak atau ani istirahatnya cukup, maka ani rajin belajar
- Ani istirahatnya cukup

Setelah dilakukan konversi ke bentuk CNF, didapat :

Fakta ke-2 : Ani tidak cerdas atau ani tidak rajin belajar atau ani akan menjadi juara kelas

Fakta ke-3 : Ani tidak makan banyak atau ani rajin belajar

Fakta ke-4 : Ani tidak cukup istirahat atau ani rajin belajar



4.1.2. Logika Predikat

Representasi Fakta Sederhana

Misal diketahui fakta-fakta sebagai berikut :

- Andi adalah seorang laki-laki : A
- Ali adalah seorang laki-laki : B
- Amir adalah seorang laki-laki : C
- Anto adalah seorang laki-laki : D
- Agus adalah seorang laki-laki : E

Jika kelima fakta tersebut dinyatakan dengan menggunakan proposisi, maka akan terjadi pemborosan, dimana beberapa pernyataan dengan predikat yang sama akan dibuat dalam proposisi yang berbeda. Logika predikat digunakan untuk merepresentasikan hal-hal yang tidak dapat direpresentasikan dengan menggunakan logika proposisi.

Pada logika predikat kita dapat merepresentasikan fakta-fakta sebagai suatu pernyataan yang disebut dengan wff (well-formed formula). Logika predikat merupakan dasar bagi bahasa AI seperti bahasa pemrograman PROLOG

Pada contoh diatas, dapat dituliskan :

`laki-laki(x)`

dimana x adalah variabel yang disubstitusikan dengan Andi, Ali, Amir, Anto, Agus, dan laki-laki yang lain.

Dalam logika predikat, suatu proposisi atau premis dibagi menjadi 2 bagian, yaitu argumen (objek) dan predikat (keterangan). Argumen adalah individu atau objek yang membuat keterangan. Predikat adalah keterangan yang membuat argumen dan predikat.

Contoh :

1. Jika besok tidak hujan, Tommy pergi ke gunung

`¬Hujan(besok) □ pergi(tommy, gunung)`

2. Diana adalah nenek dari ibu Amir

`nenek(Diana, ibu(Amir))`

3. Mahasiswa berada di dalam kelas

`didalam(mahasiswa, kelas)`

Dari contoh diatas dapat dijabarkan sebagai berikut :

di dalam= predikat (keterangan)

mahasiswa = argumen (objek)

kelas = argumen (objek)

4. Johan suka Maria

`suka(johan, maria)`

5. Pintu terbuka

`Buka(pintu)`

6. Johan suka Maria

$\text{suka}(\text{ramon}, \text{maria})$

Misal: Johan = x, Maria = y, Ramon = z

Maka: $\text{suka}(x, y) \wedge \text{suka}(z, y) \rightarrow \neg \text{suka}(x, z)$

Dibaca: Jika Johan suka Maria dan Ramon suka Maria, maka Johan tidak suka Ramon

Misal terdapat pernyataan sebagai berikut :

1. Andi adalah seorang mahasiswa
2. Andi masuk jurusan Elektro
3. Setiap mahasiswa elektro pasti mahasiswa teknik
4. Kalkulus adalah matakuliah yang sulit
5. Setiap mahasiswa teknik pasti akan suka kalkulus atau akan membencinya
6. Setiap mahasiswa pasti akan suka terhadap suatu matakuliah
7. Mahasiswa yang tidak pernah hadir pada kuliah matakuliah sulit, maka mereka pasti tidak suka terhadap matakuliah tersebut.
8. Andi tidak pernah hadir kuliah matakuliah kalkulus

Kedelapan pernyataan diatas dapat dibawa ke bentuk logika predikat dengan menggunakan operator : \rightarrow , \neg , \wedge , \vee , \forall (untuk setiap), \exists (terdapat), sebagai berikut :

1. mahasiswa(Andi)
2. elektro(Andi)
3. $\forall x : \text{elektro}(x) \rightarrow \text{teknik}(x)$
4. sulit(kalkulus)
5. $\forall x : \text{teknik}(x) \rightarrow \text{suka}(x, \text{kalkulus}) \vee \text{benci}(x, \text{kalkulus})$
6. $\forall x : \exists y : \text{suka}(x, y)$
7. $\forall x : \forall y : \text{mahasiswa}(x) \wedge \text{sulit}(y) \wedge \neg \text{hadir}(x, y) \rightarrow \neg \text{suka}(x, y)$
8. $\neg \text{hadir}(\text{Andi}, \text{kalkulus})$

Andaikan kita akan menjawab pertanyaan :

“Apakah Andi suka matakuliah kalkulus?”

Maka dari pernyataan ke-7 kita akan membuktikan bahwa “Andi tidak suka dengan matakuliah kalkulus”. Dengan menggunakan penalaran backward, bisa dibuktikan bahwa $\neg \text{suka}(\text{Andi}, \text{kalkulus})$ sebagai berikut :

$$\begin{array}{c}
 \neg \text{suka}(\text{Andi}, \text{kalkulus}) \\
 \uparrow \quad (7, \text{substitusi}) \\
 \text{mahasiswa}(\text{Andi}) \wedge \text{sulit}(\text{kalkulus}) \wedge \neg \text{hadir}(\text{Andi}, \text{kalkulus}) \\
 \uparrow \quad (1) \\
 \text{sulit}(\text{kalkulus}) \wedge \neg \text{hadir}(\text{Andi}, \text{kalkulus}) \\
 \uparrow \quad (4) \\
 \neg \text{hadir}(\text{Andi}, \text{kalkulus}) \\
 \uparrow \quad (8) \\
 \square
 \end{array}$$

4.2. List Dan Pohon/Tree

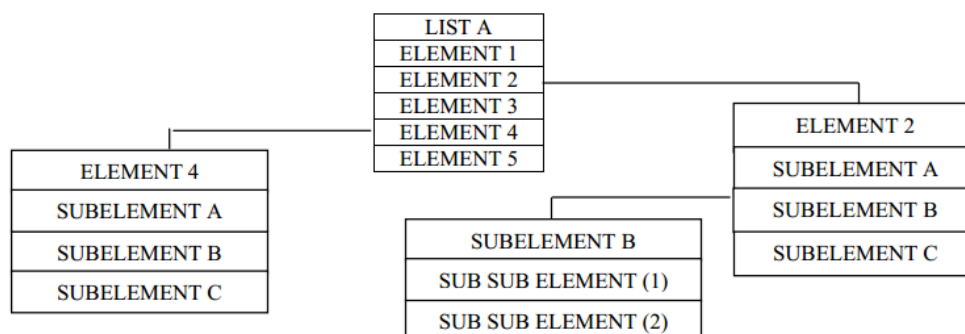
List dan Pohon/Tree merupakan struktur sederhana yang digunakan dalam representasi hirarki pengetahuan.

4.2.1. List

Adalah daftar dari rangkaian materi yang terkait. Hal ini bisa merupakan suatu daftar (list) nama orang yang anda kenal, barang-barang yang akan dibeli dari toko Serba Ada, hal-hal yang akan dikerjakan minggu ini, atau produk-produk berbagai jenis barang dalam katalog, dll.

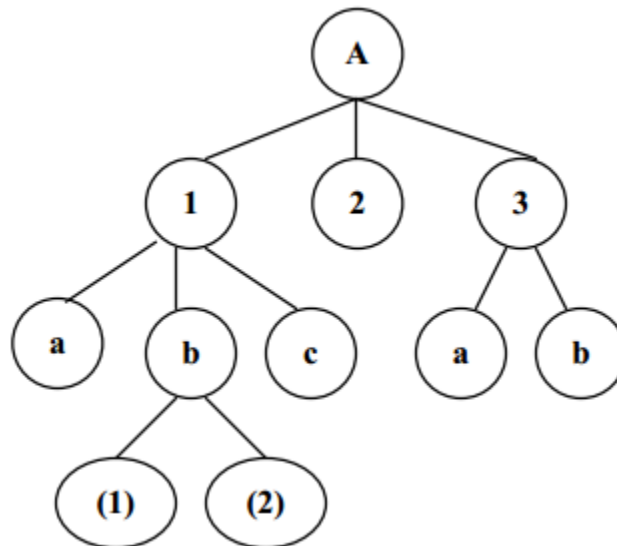
List biasanya digunakan untuk merepresentasikan hirarki pengetahuan dimana objek dikelompokkan, dikategorikan atau digabungkan sesuai dengan urutan atau hubungannya. Objek dibagi dalam kelompok atau jenis yang sama. Kemudian hubungan ditampilkan dengan menghubungkan satu sama lain.

Contoh:



4.2.2. Pohon/Tree

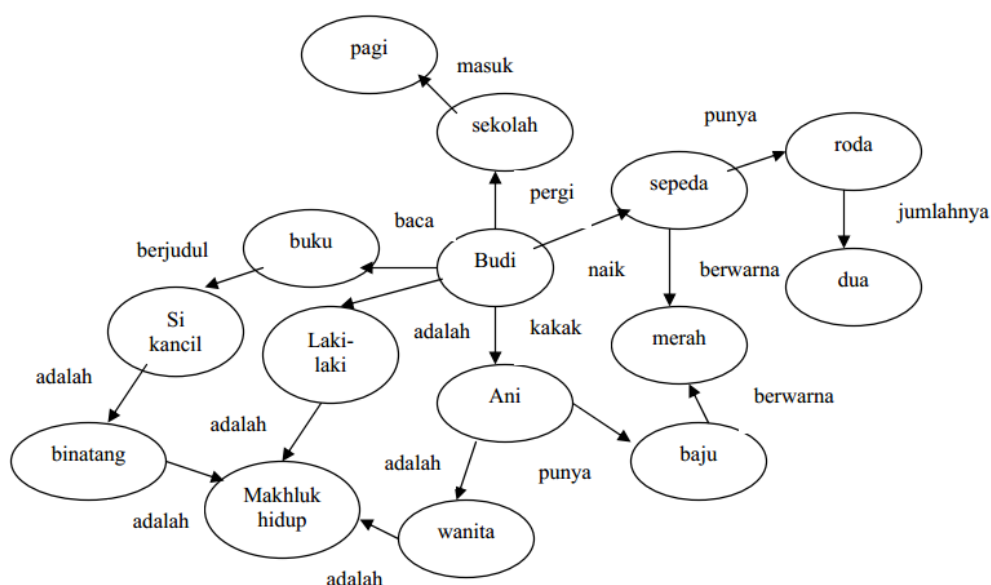
Struktur pohon adalah struktur grafik hirarki. Struktur ini merupakan cara yang sederhana untuk menggambarkan list dan hirarki pengetahuan lainnya. Contoh:



4.3. Jaringan Semantik

Jaringan semantik merupakan gambaran pengetahuan grafis yang menunjukkan hubungan antar berbagai objek. Jaringan semantik terdiri dari lingkaran-lingkaran yang menunjukkan objek dan informasi tentang objek-objek tersebut. Objek disini bisa berupa benda atau peristiwa. Antara 2 objek dihubungkan oleh arc yang menunjukkan hubungan antar objek.

Gambar berikut menunjukkan representasi pengetahuan menggunakan jaringan semantik.



4.4. Frame

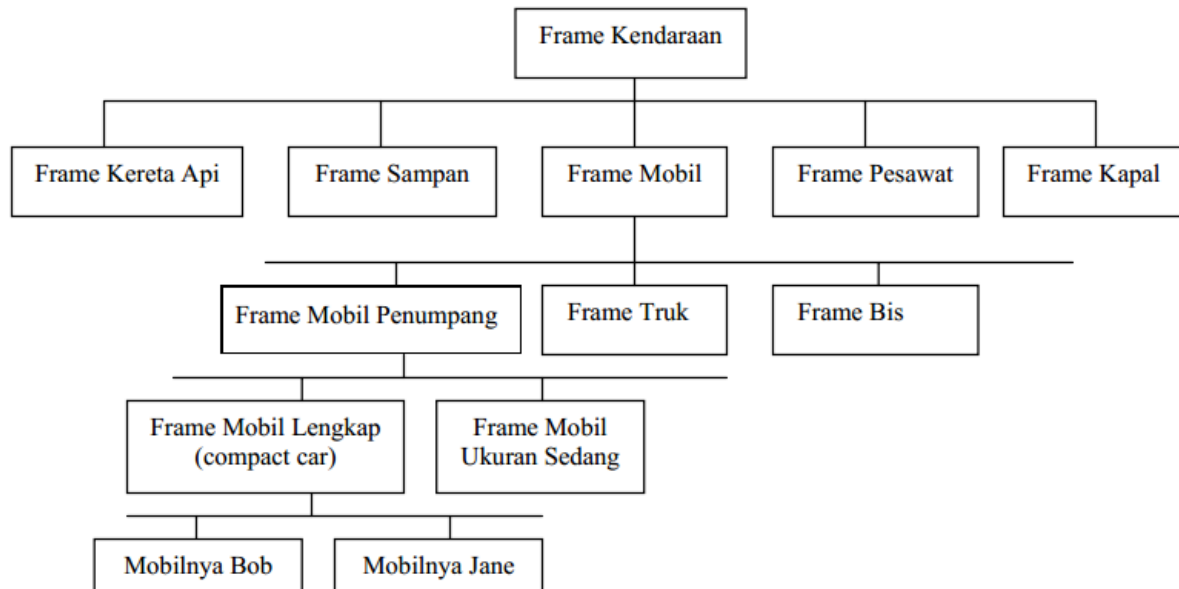
Frame merupakan kumpulan pengetahuan tentang suatu objek tertentu, peristiwa, lokasi, situasi, dll. Frame memiliki slot yang menggambarkan rincian (atribut) dan karakteristik objek. Frame biasanya digunakan untuk merepresentasikan pengetahuan yang didasarkan pada karakteristik yang sudah dikenal, yang merupakan pengalaman-pengalaman.

Dengan menggunakan frame, sangat mudah untuk membuat inferensi tentang objek, peristiwa, atau situasi baru, karena frame menyediakan basis pengetahuan yang ditarik dari pengalaman. Contoh:

| |
|--|
| Frame Mobil |
| Class : Transportasi |
| Nama pabrik : Audi |
| Negara : Jerman |
| Model : 5000 Turbo |
| Tipe : Sedan |
| Bobot : 3300 lb |
| Ukuran dasar roda : 105,8 inchi |
| Jumlah pintu : 4 (default) |
| Transmisi : 3-speed otomatis |
| Jumlah roda : 4 (default) |
| Mesin : (referensi kerangka mesin) <ul style="list-style-type: none"> • Tipe : in-line, overhead cam • Jumlah silinder : 5 |
| Akselerasi <ul style="list-style-type: none"> 0-60 : 40,4 detik ¼ mil : 17,1 detik, 85 mph Jarak gas : rata-rata 22 mpg |
| Frame Mesin |
| Kaliber silinder : 3,19 inci |
| Tak silinder : 3,4 inci |
| Rasio kompresi : 7,8 : 1 |
| Sistem bahan bakar : injeksi dengan pertukaran turbo |
| Tenaga : 140 HP |
| Torsi : 160/ft/LB |

Kebanyakan sistem AI menggunakan kumpulan frame yang saling terkait satu dengan lainnya bersama-sama yang dikenal dengan Hirarki frame. Gambar di bawah ini menunjukkan hirarki frame kendaraan, terdiri dari 5 frame yaitu frame kereta api, frame sampan, frame mobil, frame pesawat, frame kapal. Masing-masing frame masih

dapat dipecah lagi menjadi beberapa frame yang rinci, misal frame mobil terdiri dari frame penumpang mobil, frame truk, frame bis.



Susunan hirarki dari frame memungkinkan pewarisan frame. Akar dari tree terletak dipuncak, dimana level tertinggi dari abstraksi disajikan. Frame pada bagian dasar (bawah) disebut daun dari tree.

Hirarki mengijinkan pewarisan sifat-sifat. Setiap frame biasanya mewarisi sifat-sifat dari frame dengan level yang lebih tinggi. Pewarisan merupakan mekanisme untuk membentuk pengetahuan, yang menyediakan nilai slot, dari frame ke frame.

Didalam hirarki diatas, masing-masing frame dirinci hubungannya seperti hubungan antara frame orangtua (parent frame) dan anak (child frame).

| Parent Frame Nama : Compact Car | |
|------------------------------------|------------------------|
| Slot | Facets |
| Pemilik | Cek daftar registrasi |
| Warna | Daftar per manufaktur |
| No silinder | |
| Range | 4 atau 6 |
| Jika dibutuhkan | Tanya pemilik |
| Buatan | |
| Daftar range | Semua manufaktur |
| Jika dibutuhkan | Tanya pemilik |
| Model | Gunakan hubungan frame |
| Model (tahun) | |
| Range | 1950 – 2001 |
| Jika dibutuhkan | Tanya pemilik |

| Child Frame Nama : Mobilnya Jane | |
|-------------------------------------|--------|
| Slot | Facets |
| Pemilik | Jane |
| Warna | Biru |
| No silinder | 6 |
| Buatan | Honda |
| Model | Accord |
| Model (tahun) | 1992 |

4.5. Tabel Keputusan (Decision Table)

Dalam sebuah tabel keputusan/decision table, umumnya:

- ✓ Pengetahuan diorganisasikan dalam format spreadsheet, menggunakan baris dan kolom.
- ✓ Tabel dibagi 2 bagian, pertama sebuah list dari atribut dibuat dan untuk setiap atribut semua nilai yang mungkin ditampilkan. Kemudian sebuah list kesimpulan dirumuskan
- ✓ Pengetahuan dalam tabel diperoleh dari proses akuisisi pengetahuan.

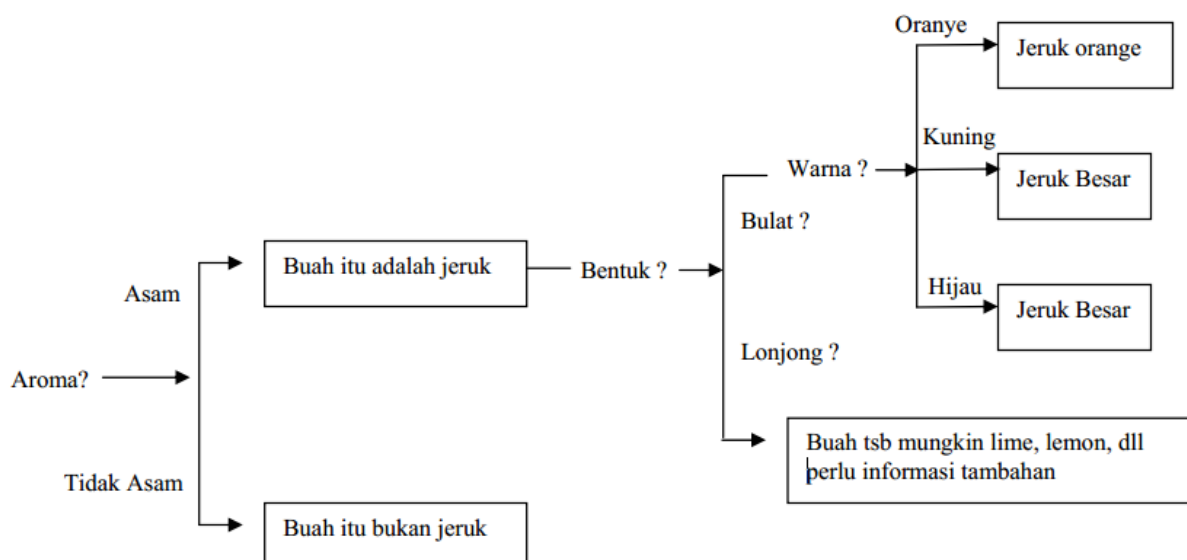
Contoh:

| Atribut | | | | | | | | |
|------------|--------|--------|--------|-------|---------|---------|---------|---------|
| Bentuk | Bulat | Bulat | Bulat | Bulat | Lonjong | Lonjong | Lonjong | Lonjong |
| Aroma | Asam | Asam | Manis | Manis | Manis | Manis | Asam | Manis |
| Warna | Kuning | Oranye | Kuning | Merah | Kuning | Kuning | Oranye | Hijau |
| Rasa | Asam | Manis | Manis | Manis | Manis | Manis | Asam | Manis |
| Kulit | Kasar | Kasar | Halus | Halus | Halus | Halus | Halus | Halus |
| Kesimpulan | | | | | | | | |
| Anggur | X | | | | | | | |
| Jeruk | | X | | | | | | |
| Apel | | | X | X | | | | |
| Pisang | | | | | X | | | |
| Pir | | | | | | X | | X |

4.6. Pohon Keputusan (Decision Tree)

Keuntungan utama representasi pengetahuan dengan pohon keputusan adalah dapat menyederhanakan proses akuisisi pengetahuan dan dapat dengan mudah dikonversikan ke bentuk aturan (rule) .

Contoh:



4.7. Naskah (Script)

Script adalah skema representasi pengetahuan yang sama dengan frame, yaitu merepresentasikan pengetahuan berdasarkan karakteristik yang sudah dikenal

sebagai pengalaman-pengalaman. Perbedaannya, frame menggambarkan objek, sedangkan script menggambarkan urutan peristiwa.

Dalam menggambarkan urutan peristiwa, script menggunakan slot yang berisi informasi tentang orang, objek, dan tindakan-tindakan yang terjadi dalam suatu peristiwa.

Elemen script meliputi :

- ✓ Kondisi input, yaitu kondisi yang harus dipenuhi sebelum terjadi atau berlaku suatu peristiwa dalam script
- ✓ Track, yaitu variasi yang mungkin terjadi dalam suatu script
- ✓ Prop, berisi objek-objek pendukung yang digunakan selama peristiwa terjadi
- ✓ Role, yaitu peran yang dimainkan oleh seseorang dalam peristiwa
- ✓ Scene, yaitu adegan yang dimainkan yang menjadi bagian dari suatu peristiwa
- ✓ Hasil, yaitu kondisi yang ada setelah urutan peristiwa dalam script terjadi.

Berikut ini adalah contoh script kejadian yang ada di “Ujian Akhir”

Jalur (track) : ujian mata kuliah Kecerdasan Buatan

Role (peran) : mahasiswa, pengawas

Prop pendukung) : lembar soal, lembar jawab, presensi, pena, dll

Kondisi input : mahasiswa terdaftar mengikuti ujian

Adegan (scene) -1 : Persiapan pengawas

- ✓ Pengawas menyiapkan lembar soal
- ✓ Pengawas menyiapkan lembar jawab
- ✓ Pengawas menyiapkan lembar presensi

Adegan-2 : Mahasiswa masuk ruangan

- ✓ Pengawas mempersilahkan mahasiswa masuk
- ✓ Pengawas membagikan lembar soal
- ✓ Pengawas membagikan lembar jawab
- ✓ Pengawas memimpin doa

Adegan – 3 : Mahasiswa mengerjakan soal ujian

- ✓ Mahasiswa menuliskan identitas di lembar jawab
- ✓ Mahasiswa menandatangani lembar jawab
- ✓ Mahasiswa mengerjakan soal
- ✓ Mahasiswa mengecek jawaban

Adegan – 4: Mahasiswa telah selesai ujian

- ✓ Pengawas mempersilahkan mahasiswa keluar ruangan
- ✓ Mahasiswa mengumpulkan kembali lembar jawab
- ✓ Mahasiswa keluar ruangan

Adegan – 5: Mahasiswa mengemasi lembar jawab

- ✓ Pengawas mengurutkan lembar jawab
- ✓ Pengawas mengecek lembar jawab dan presensi
- ✓ Pengawas meninggalkan ruangan

Hasil :

- ✓ Mahasiswa merasa senang dan lega
- ✓ Mahasiswa merasa kecewa
- ✓ Mahasiswa pusing
- ✓ Mahasiswa memaki - maki
- ✓ Mahasiswa sangat bersyukur

4.8. Sistem Produksi (Aturan Produksi/Production Rules)

Representasi pengetahuan dengan sistem produksi berupa aplikasi aturan (rule) yang berupa :

- 1) Antecedent, yaitu bagian yang mengekspresikan situasi atau premis (pernyataan berawalan IF)
- 2) Konsekuen, yaitu bagian yang menyatakan suatu tindakan tertentu atau konklusi yang diterapkan jika suatu situasi atau premis bernilai benar (pernyataan berawalan THEN)

Konsekuensi atau konklusi yang dinyatakan pada bagian THEN baru dinyatakan benar, jika bagian IF pada sistem tersebut juga benar atau sesuai dengan aturan tertentu.

Contoh :

"IF lalulintas padat THEN saya naik sepeda motor"

Aturan dapat ditulis dalam beberapa bentuk :

- IF premis THEN kesimpulan
Jika pendapatan tinggi MAKA pajak yang harus dibayar juga tinggi
- Kesimpulan IF premis
Pajak yang harus dibayar tinggi JIKA pendapatan tinggi
- Inclusion of ELSE
IF pendapatan tinggi OR pengeluaran tinggi, THEN pajak yang harus dibayar tinggi
ELSE pajak yang harus dibayar rendah
- Aturan yang lebih kompleks
IF rating kredit tinggi AND gaji lebih besar dari Rp 3 Juta OR aset lebih dari Rp 75
Juta AND sejarah pembayaran tidak miskin THEN pinjaman diatas Rp 10 Juta
disetujui dan daftar pinjaman masuk kategori "B"

Apabila pengetahuan direpresentasikan dengan aturan, maka ada 2 metode penalaran yang dapat digunakan :

- ❖ Forward Reasoning (penalaran maju)
Pelacakan dimulai dari keadaan awal (informasi atau fakta yang ada) dan kemudian dicoba untuk mencocokkan dengan tujuan yang diharapkan
- ❖ Backward Reasoning (penalaran mundur)
Penalaran dimulai dari tujuan atau hipotesa, baru dicocokkan dengan keadaan awal atau faktafakta yang ada.

Ada beberapa faktor yang mempengaruhi pemilihan backward atau forward dalam memilih metode penalaran :

- ✓ banyaknya keadaan awal dan tujuan. Jika jumlah keadaan awal lebih kecil daripada tujuan, maka digunakan penalaran forward. Sebaliknya jika jumlah tujuan lebih banyak daripada keadaan awal, maka dipilih penalaran backward
- ✓ rata-rata jumlah node yang dapat diraih langsung dari suatu node. Lebih baik dipilih yang jumlah node tiap cabangnya lebih sedikit

- ✓ apakah program butuh menanyai user untuk melakukan justifikasi terhadap proses penalaran? Jika ya, maka langkah baiknya jika dipilih arah yang lebih memudahkan user
- ✓ bentuk kejadian yang akan memicu penyelesaian masalah. Jika kejadian itu berupa fakta baru, maka lebih baik dipilih penalaran forward. Namun jika kejadian itu berupa query, maka lebih baik digunakan penalaran backward.¹²

¹² Ibid.

BAB V

PENANGANAN KETIDAKPASTIAN

Dalam kenyataan sehari-hari banyak masalah di dunia ini tidak dapat dimodelkan secara lengkap dan konsisten. Suatu penalaran dimana adanya penambahan fakta baru mengakibatkan ketidak konsistenan, dengan ciri-ciri penalaran sebagai berikut :

- ✓ adanya ketidakpastian
- ✓ adanya perubahan pada pengetahuan
- ✓ adanya penambahan fakta baru dapat mengubah konklusi yang sudah terbentuk

Contoh :

Premis-1 : Aljabar adalah pelajaran yang sulit

Premis-2 : Geometri adalah pelajaran yang sulit

Premis-3 : Kalkulus adalah pelajaran yang sulit

Konklusi : Matematika adalah pelajaran yang sulit

Munculnya premis baru bisa mengakibatkan gugurnya konklusi yang sudah diperoleh, misal :

“Premis-4 : Kinematika adalah pelajaran yang sulit”

Premis tersebut menyebabkan konklusi : “Matematika adalah pelajaran yang sulit”, menjadi salah, karena Kinematika bukan merupakan bagian dari Matematika, sehingga bila menggunakan penalaran induktif sangat dimungkinkan adanya ketidakpastian.

Untuk mengatasi ketidakpastian maka dapat digunakan penalaran/perhitungan ketidakpastian dengan menggunakan beberapa metode, misalnya theorem bayes, certainty factor dan damster-shafer.¹³

5.1. Theorema Bayes

Dasar dari theorema bayes adalah probabilitas dimana probabilitas menunjukkan kemungkinan sesuatu akan terjadi atau tidak. Rumus umum probabilitas adalah

$$p(x) = \frac{\text{jumlah kejadian berhasil}}{\text{jumlah semua kejadian}}$$

¹³ Kusumadewi, *Artificial Intelligence (Teknik Dan Aplikasinya)*.

Contoh:

Misal dari 10 orang sarjana , 3 orang menguasai cisco, sehingga peluang untuk memilih sarjana yang menguasai cisco adalah,

$$p(\text{cisco}) = 3/10 = 0.3$$

Rumus umum theorema bayes adalah:

$$p(H_i | E) = \frac{p(E | H_i) * (p(H_i))}{\sum_{k=1}^n p(E | H_k) * (p(H_k))}$$

dengan :

$p(H_i | E)$ = probabilitas hipotesis H_i benar jika diberikan evidence (fakta) E

$p(E | H_i)$ = probabilitas munculnya evidence(fakta) E jika diketahui hipotesis H_i benar

$p(H_i)$ = probabilitas hipotesis H_i (menurut hasil sebelumnya) tanpa memandang evidence(fakta) apapun

n = jumlah hipotesis yang mungkin

Contoh :

Mawar mengalami gejala ada bintik-bintik di wajahnya. Dokter menduga bahwa Mawar terkena cacar dengan :

- Probabilitas munculnya bintik-bintik di wajah, jika Mawar terkena cacar → $p(\text{bintik} | \text{cacar}) = 0.8$
- Probabilitas Mawar terkena cacar tanpa memandang gejala apapun → $p(\text{cacar}) = 0.4$
- Probabilitas munculnya bintik-bintik di wajah, jika Mawar terkena alergi → $p(\text{bintik} | \text{alergi}) = 0.3$
- Probabilitas Mawar terkena alergi tanpa memandang gejala apapun → $p(\text{alergi}) = 0.7$
- Probabilitas munculnya bintik-bintik di wajah, jika Mawar jerawat → $p(\text{bintik} | \text{jerawatan}) = 0.9$
- Probabilitas Mawar jerawat tanpa memandang gejala apapun → $p(\text{jerawatan}) = 0.5$

Maka :

- Probabilitas Mawar terkena cacar karena ada bintik-bintik di wajahnya :

$$p(\text{cacar} | \text{bintik}) = \frac{p(\text{bintik} | \text{cacar}) * p(\text{cacar})}{p(\text{bintik} | \text{cacar}) * p(\text{cacar}) + p(\text{bintik} | \text{alergi}) * p(\text{alergi}) + p(\text{bintik} | \text{jerawat}) * p(\text{jerawat})}$$

$$p(\text{cacar} | \text{bintik}) = \frac{(0.8) * (0.4)}{(0.8) * (0.4) + (0.3) * (0.7) + (0.9) * (0.5)} = \frac{0.32}{0.98} = 0.327$$

➤ Probabilitas Mawar terkena Alergi karena ada bintik-bintik di wajahnya :

$$p(\text{alergi} | \text{bintik}) = \frac{p(\text{bintik} | \text{alergi}) * p(\text{alergi})}{p(\text{bintik} | \text{cacar}) * p(\text{cacar}) + p(\text{bintik} | \text{alergi}) * p(\text{alergi}) + p(\text{bintik} | \text{jerawat}) * p(\text{jerawat})}$$

$$p(\text{alergi} | \text{bintik}) = \frac{(0.3) * (0.7)}{(0.8) * (0.4) + (0.3) * (0.7) + (0.9) * (0.5)} = \frac{0.21}{0.98} = 0.214$$

➤ Probabilitas Mawar terkena Jerawatan karena ada bintik-bintik di wajahnya :

$$p(\text{jerawat} | \text{bintik}) = \frac{p(\text{bintik} | \text{jerawat}) * p(\text{jerawat})}{p(\text{bintik} | \text{cacar}) * p(\text{cacar}) + p(\text{bintik} | \text{alergi}) * p(\text{alergi}) + p(\text{bintik} | \text{jerawat}) * p(\text{jerawat})}$$

$$p(\text{jerawat} | \text{bintik}) = \frac{(0.9) * (0.5)}{(0.8) * (0.4) + (0.3) * (0.7) + (0.9) * (0.5)} = \frac{0.45}{0.98} = 0.459$$

Berdasarkan perhitungan tersebut, maka Mawar lebih dekat dengan Hipotesa terkena Jerawatan.

Jika setelah dilakukan pengujian terhadap hipotesis muncul satu atau lebih evidence (fakta) atau observasi baru maka dapat dihitung dengan persamaan:

$$p(H | E, e) = p(H | E) * \frac{p(e | E, H)}{p(e | E)}$$

dengan:

e = evidencelama

E = evidenceatau observasibaru

$p(H | E, e)$ = probabilitas hipotesis H benar jika munculevidencebaru E dari evidencelama e

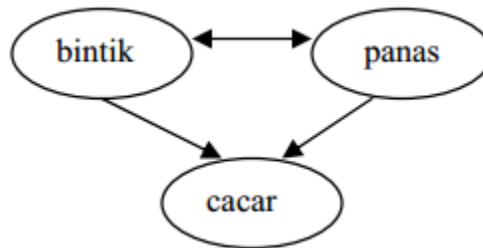
$p(H | E)$ = probabilitas hipotesis H benar jika diberikanevidence E

$p(e | E, H)$ = kaitan antara e dan E jika hipotesis H benar

$p(e | E)$ = kaitan antara e dan E tanpa memandanghipotesisapapun

Misal :

Adanya bintik-bintik di wajah merupakan gejala seseorang terkena cacar. Observasi baru menunjukkan bahwa selain bintik-bintik di wajah, panas badan juga merupakan gejala orang kena cacar. Jadi antara munculnya bintik-bintik di wajah dan panas badan juga memiliki keterkaitan satu sama lain.



Mawar ada bintik-bintik di wajahnya. Dokter menduga bahwa Asih terkena cacar dengan probabilitas terkena cacar bila ada bintik-bintik di wajah $\rightarrow p(\text{cacar} \mid \text{bintik}) = 0.8$ Ada observasi bahwa orang terkena cacar pasti mengalami panas badan. Jika diketahui probabilitas orang terkena cacar bila panas badan $\rightarrow p(\text{cacar} \mid \text{panas}) = 0.5$ Keterkaitan antara adanya bintik-bintik di wajah dan panas badan bila seseorang terkena cacar $\rightarrow p(\text{bintik} \mid \text{panas, cacar}) = 0.4$ Keterkaitan antara adanya bintik-bintik di wajah dan panas badan $\rightarrow p(\text{bintik} \mid \text{panas}) = 0.6$

Maka :

$$p(H \mid E, e) = p(H \mid E) * \frac{p(e \mid E, H)}{p(e \mid E)}$$

$$p(\text{cacar} \mid \text{panas, bintik}) = p(\text{cacar} \mid \text{panas}) * \frac{p(\text{bintik} \mid \text{panas, cacar})}{p(\text{bintik} \mid \text{panas})}$$

$$p(\text{cacar} \mid \text{panas, bintik}) = (0.5) * \frac{(0.4)}{(0.6)} = 0.33$$

5.2. Certainty Factor (CF)

Certainty Factor (CF) menunjukkan ukuran kepastian terhadap suatu fakta atau aturan.

$$CF[h, e] = MB[h, e] - MD[h, e]$$

Dimana

CF[h,e] = faktor kepastian

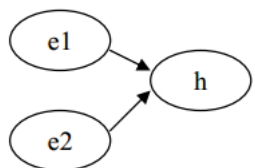
MB[h,e] = ukuran kepercayaan/tingkat keyakinan terhadap hipotesish, jika diberikan/dipengaruhi evidence e (antara 0 dan 1)

MD[h,e] = ukuran ketidakpercayaan / tingkat ketidak yakinan terhadap hipotesis h, jika diberikan/dipengaruhi evidence e (antara 0 dan 1)

3 hal yang mungkin terjadi :

1) Beberapa evidence dikombinasikan untuk menentukan CF dari suatu hipotesis.

Jika e1 dan e2 adalah observasi, maka :



$$MB[h, e1 \wedge e2] = \begin{cases} 0 & \text{jika } MD[h, e1 \wedge e2] = 1 \\ MB[h, e1] + MB[h, e2] * (1 - MB[h, e1]) & \text{lainnya} \end{cases}$$

$$MD[h, e1 \wedge e2] = \begin{cases} 0 & \text{jika } MB[h, e1 \wedge e2] = 1 \\ MD[h, e1] + MD[h, e2] * (1 - MD[h, e1]) & \text{lainnya} \end{cases}$$

Contoh:

Misal suatu observasi memberikan kepercayaan terhadap h dengan $MB[h, e1]=0,3$ dan $MD[h, e1]=0$ maka :

$$CF[h, e1] = 0,3 - 0 = 0,3$$

Jika ada observasi baru dengan $MB[h, e2]=0,2$ dan $MD[h, e2]=0$, maka :

$$MB[h, e1 \wedge e2] = 0,3 + 0,2 * (1 - 0,3) = 0,44$$

$$MD[h, e1 \wedge e2] = 0$$

$$CF[h, e1 \wedge e2] = 0,44 - 0 = 0,44$$

Mawar menderita bintik-bintik di wajahnya. Dokter memperkirakan Mawar terkena cacar dengan kepercayaan $MB[\text{cacar}, \text{bintik}]=0,80$ dan $MD[\text{cacar}, \text{bintik}]=0,01$ maka :

$$CF[\text{cacar}, \text{bintik}] = 0,80 - 0,01 = 0,79$$

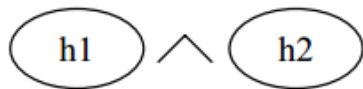
Jika ada observasi baru bahwa Mawar juga panas badan dengan kepercayaan, $MB[\text{cacar}, \text{panas}] = 0,7$ dan $MD[\text{cacar}, \text{panas}] = 0,08$ maka :

$$MB[\text{cacar}, \text{bintik} \wedge \text{panas}] = 0,8 + 0,7 * (1 - 0,8) = 0,94$$

$$MD[\text{cacar}, \text{bintik} \wedge \text{panas}] = 0,01 + 0,08 * (1 - 0,01) = 0,0892$$

$$CF[\text{cacar}, \text{bintik} \wedge \text{panas}] = 0,94 - 0,0892 = 0,8508$$

2) CF dihitung dari kombinasi beberapa hipotesis. Jika h1 dan h2 adalah hipotesis, maka :



$$MB[h1 \wedge h2, e] = \min (MB[h1, e], MB[h2, e])$$

$$MB[h1 \vee h2, e] = \max (MB[h1, e], MB[h2, e])$$

$$MD[h1 \wedge h2, e] = \min (MD[h1, e], MD[h2, e])$$

$$MD[h1 \vee h2, e] = \max (MD[h1, e], MD[h2, e])$$

Contoh 1:

Misal suatu observasi memberikan kepercayaan terhadap h1 dengan $MB[h1, e]=0,5$ dan $MD[h1, e]=0,2$ maka :

$$CF[h1, e] = 0,5 - 0,2 = 0,3$$

Jika observasi tersebut juga memberikan kepercayaan terhadap h2 dengan $MB[h2, e]=0,8$ dan $MD[h2, e]=0,1$ maka :

$$CF[h2, e] = 0,8 - 0,1 = 0,7$$

Untuk mencari $CF[h1 \wedge h2, e]$ diperoleh dari

$$MB[h1 \wedge h2, e] = \min (0,5 ; 0,8) = 0,5$$

$$MD[h1 \wedge h2, e] = \min (0,2 ; 0,1) = 0,1$$

$$CF[h1 \wedge h2, e] = 0,5 - 0,1 = 0,4$$

Untuk mencari $CF[h1 \vee h2, e]$ diperoleh dari

$$MB[h1 \vee h2, e] = \max (0,5 ; 0,8) = 0,8$$

$$MD[h1 \vee h2, e] = \max (0,2 ; 0,1) = 0,2$$

$$CF[h1 \vee h2, e] = 0,8 - 0,2 = 0,6$$

Mawar menderita bintik-bintik di wajahnya. Dokter memperkirakan Mawar terkena cacar dengan kepercayaan

$$MB[cacar, bintik] = 0,80 \text{ dan } MD[cacar, bintik]=0,01$$

$$\text{maka } CF[cacar, bintik] = 0,80 - 0,01 = 0,79$$

Jika observasi tersebut juga memberikan kepercayaan bahwa Mawar mungkin juga terkena alergi dengan kepercayaan $MB[alergi, bintik] = 0,4$ dan $MD[alergi, bintik]=0,3$

$$\text{maka } CF[alergi, bintik] = 0,4 - 0,3 = 0,1$$

Untuk mencari $CF[cacar \wedge alergi, bintik]$ diperoleh dari

$$MB[cacar \wedge alergi, bintik] = \min (0,8 ; 0,4) = 0,4$$

$$MD[cacar \wedge alergi, bintik] = \min (0,01 ; 0,3) = 0,01$$

$$CF[cacar \wedge alergi, bintik] = 0,4 - 0,01 = 0,39$$

Untuk mencari CF[cacar valergi, bintik] diperoleh dari

$$MB[\text{cacar Valergi, bintik}] = \max (0,8 ; 0,4) = 0,8$$

$$MD[\text{cacar Valergi, bintik}] = \max (0,01 ; 0,3) = 0,3$$

$$CF[\text{cacar Valergi, bintik}] = 0,8 - 0,3 = 0,5$$

KESIMPULAN : semula faktor kepercayaan bahwa Mawar terkena cacar dari gejala munculnya bintik-bintik di wajahnya adalah 0,79. Demikian pula faktor kepercayaan bahwa Mawar terkena alergi dari gejala munculnya bintik-bintik di wajah adalah 0,1.

Dengan adanya gejala yang sama mempengaruhi 2 hipotesis yang berbeda ini memberikan faktor kepercayaan :

$$\text{Mawar menderita cacar dan alergi} = 0,39$$

$$\text{Mawar menderita cacar atau alergi} = 0,5$$

Contoh 2:

Pertengahan tahun 2002, ada indikasi bahwa turunnya devisa Indonesia disebabkan oleh permasalahan TKI di Malaysia. Apabila diketahui

$$MB[\text{devisaturun, TKI}] = 0,8 \text{ dan } MD[\text{devisaturun, TKI}] = 0,3$$

maka $CF[\text{devisaturun, TKI}]$:

$$\begin{aligned} CF[\text{devisaturun, TKI}] &= MB[\text{devisaturun, TKI}] - MD[\text{devisaturun, TKI}] \\ &= 0,8 - 0,3 \\ &= 0,5 \end{aligned}$$

Akhir September 2002 kemarau berkepanjangan mengakibatkan gagal panen yang cukup serius, berdampak pada turunnya ekspor Indonesia. Bila diketahui

$$MB[\text{devisaturun, eksporturun}] = 0,75 \text{ dan}$$

$$MD[\text{devisaturun, eksporturun}] = 0,1$$

maka

$CF[\text{devisaturun, eksporturun}]$ dan

$CF[\text{devisaturun, TKI} \wedge \text{eksporturun}]$ adalah:

$$CF[\text{devisaturun, eksporturun}] =$$

$$MB[\text{devisaturun, eksporturun}] - MD[\text{devisaturun, eksporturun}]$$

$$= 0,75 - 0,1 = 0,65$$

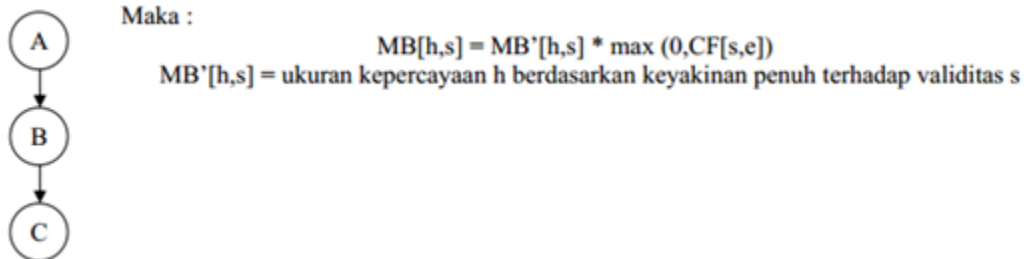
$$\begin{aligned}
& MB[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] = \\
& MB[\text{devisaturun}, \text{TKI}] + MB[\text{devisaturun}, \text{eksporturun}] \\
& * (1 - MB[\text{devisaturun}, \text{TKI}]) \\
& = 0,8 + 0,75 * (1 - 0,8) = 0,95 \\
& MD[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] = \\
& MD[\text{devisaturun}, \text{TKI}] + MD[\text{devisaturun}, \text{eksporturun}] * (1 - \\
& MD[\text{devisaturun}, \text{TKI}]) \\
& = 0,3 + 0,1 * (1 - 0,3) = 0,37 \\
& CF[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] = \\
& MB[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] - MD[\text{devisaturun}, \text{TKI} \wedge \\
& \text{eksporturun}] \\
& = 0,95 - 0,37 = 0,58
\end{aligned}$$

Isu terorisme di Indonesia pasca bom bali tgl 12 Oktober 2002 ternyata juga ikut mempengaruhi turunnya devisa Indonesia sebagai akibat berkurangnya wisatawan asing. Bila diketahui $MB[\text{devisaturun}, \text{bombali}] = 0,5$ dan

$$\begin{aligned}
& MD[\text{devisaturun}, \text{bombali}] = 0,3, \text{ maka} \\
& CF[\text{devisaturun}, \text{bombali}] \text{ dan} \\
& CF[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun} \wedge \text{bombali}] : \\
& CF[\text{devisaturun}, \text{bombali}] = MB[\text{devisaturun}, \text{bombali}] - \\
& MD[\text{devisaturun}, \text{bombali}] \\
& = 0,5 - 0,3 = 0,2 \\
& MB[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun} \wedge \text{bombali}] = \\
& MB[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] + \\
& MB[\text{devisaturun}, \text{bombali}] * (1 - MB[\text{devisaturun}, \\
& \text{TKI} \wedge \text{eksporturun}]) \\
& = 0,95 + 0,5 * (1 - 0,95) = 0,975 \\
& MD[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun} \wedge \text{bombali}] = \\
& MD[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] + \\
& MD[\text{devisaturun}, \text{bombali}] * \\
& 1 - MD[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}]) \\
& = 0,37 + 0,3 * (1 - 0,37) = 0,559 \\
& CF[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun} \wedge \text{bombali}] = \\
& MB[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun} \wedge \text{bombali}] -
\end{aligned}$$

$$\begin{aligned} & MD[\text{devisaturun, TKI } \wedge \text{eksporturun } \wedge \text{bombali}] \\ & = 0,975 - 0,559 = 0,416 \end{aligned}$$

- 3) Beberapa aturan saling bergandengan, ketidakpastian dari suatu aturan menjadi input untuk aturan yang lainnya, Maka:



Contoh :

PHK = terjadi PHK

Pengangguran = muncul banyak pengangguran

Gelandangan = muncul banyak gelandangan

Aturan 1 :

IF terjadi PHK THEN muncul banyak pengangguran

$CF[\text{pengangguran, PHK}] = 0,9$

Aturan 2 :

IF muncul banyak pengangguran THEN muncul banyak gelandangan

$MB[\text{gelandangan, pengangguran}] = 0,7$

Maka =

$MB[\text{gelandangan, pengangguran}] = [0,7] * [0,9] = 0,63$

5.3. Dempster-Shafer.

Metode Dempster-Shafer pertama kali diperkenalkan oleh Dempster, yang melakukan percobaan model ketidakpastian dengan range probabilitas sebagai probabilitas tunggal. Kemudian pada tahun 1976 Shafer mempublikasikan teori Dempster tersebut pada sebuah buku yang berjudul *Mathematical Theory of Evident*. Secara umum teori Dempster-Shafer ditulis dalam suatu interval [Belief, Plausibility].¹⁴

Belief (Bel) adalah ukuran kekuatan evidence dalam mendukung suatu himpunan proposisi. Jika bernilai 0 (nol) maka mengindikasikan bahwa tidak ada evidence, dan

¹⁴ J. Giarratano and G Riley, *Expert Systems ; Principles and Programming* (Boston: PWS Publishing Company, 2005).

jika bernilai 1 menunjukkan adanya kepastian. Menurut Giarratano dan Riley fungsi belief dapat diformulasikan dengan Persamaan

$$Bel(X) = \sum_{Y \subseteq X} m(Y)$$

sedangkan Plausibility (Pls) dinotasikan sebagai Persamaan

$$Pls(X) = 1 - Bel(X') = 1 - \sum_{Y \subseteq X'} m(Y)$$

dimana:

$Bel(X)$ = Belief (X)

$Pls(X)$ = Plausibility (X)

$m(X)$ = mass function dari (X)

$m(Y)$ = mass function dari (Y)

Plausibility juga bernilai 0 sampai 1, jika kita yakin akan X' maka dapat dikatakan Belief (X') = 1 sehingga dari rumus di atas nilai $Pls(X) = 0$. Beberapa kemungkinan range antara Belief dan Plausibility disajikan pada Tabel (Giarratano dan Riley, 1994):

| Kemungkinan | Keterangan |
|---|---------------------------------|
| [1, 1] | Semua Benar |
| [0, 0] | Semua Salah |
| [0, 1] | Ketidakpastian |
| [Bel, 1] where $0 < Bel < 1$ | Cenderung Mendukung |
| [0, Pls] where $0 < Pls < 1$ | Cenderung Menolak |
| [Bel, Pls] where $0 < Bel \leq Pls < 1$ | Cenderung Mendukung dan Menolak |

Pada teori Dempster-Shafer juga dikenal adanya frame of discernment yang dinotasikan dengan Θ . FOD ini merupakan semesta pembicaraan dari sekumpulan hipotesis sehingga sering disebut dengan environment yang di formulasikan dengan Persamaan

$$\Theta = \{\theta_1, \theta_2, \dots, \theta_n\}$$

dimana:

Θ = FOD atau environment

$\theta_1, \dots, \theta_n$ = elemen/unsur bagian dalam environment

Environment mengandung elemen-elemen yang menggambarkan kemungkinan sebagai jawaban dan hanya ada satu yang akan sesuai dengan jawaban yang

dibutuhkan. Kemungkinan ini dalam teori Dempster- Shafer disebut dengan power set dan dinotasikan dengan $P(\Theta)$, setiap elemen dalam power set ini memiliki nilai interval antara 0 sampai 1.

$$m = P(\Theta) \rightarrow [0,1]$$

sehingga dapat dirumuskan dengan Persamaan

$$\sum_{X \in P(\Theta)} m(X) = 1 \approx \sum_{X \in P(\theta)} m(X) = 1$$

dengan $P(\Theta)$ = power set dan $m(X)$ = mass function dari (X), sebagai

contoh:

$$P(\text{hostile}) = 0,7$$

$$P(\text{non-hostile}) = 1 - 07 = 0,3$$

Pada contoh di atas belief dari hostile adalah 0,7 sedangkan disbelief hostile adalah 0,3. dalam teori Dempster-Shafer, disbelief dalam environment biasanya dinotasikan $m(\Theta)$. Sedangkan mass function (m) dalam teori Dempster-Shafer adalah tingkat kepercayaan dari suatu evidence (gejala), sering disebut dengan evidence measure sehingga dinotasikan dengan (m).

Pada aplikasi sistem pakar dalam satu penyakit terdapat sejumlah evidence yang akan digunakan pada faktor ketidakpastian dalam pengambilan keputusan untuk diagnosa suatu penyakit. Untuk mengatasi sejumlah evidence tersebut pada teori Dempster-Shafer menggunakan aturan yang lebih dikenal dengan Dempster's Rule of Combination

$$m1 \oplus m2(Z) = \sum_{X \cap Y = Z} m1(X)m2(Y)$$

dimana:

$m1 (+) m2(Z)$ = mass function dari evidence (Z)

$m1(X)$ = mass function dari evidence (X)

$m2(Y)$ = mass function dari evidence (Y)

(+) = operator direct sum

Dempster's Rule of Combination diformulasikan pada persamaan

$$m1 \oplus m2(Z) = \frac{\sum_{X \cap Y = Z} m1(X)m2(Y)}{1 - k}$$

dimana: k = Jumlah evidential conflict.

Besarnya jumlah evidential conflict (k) dirumuskan Persamaan

$$k = \sum_{X \cap Y = \theta} m1(X)m2(Y)$$

sehingga bila persamaan ini disubstitusikan ke persamaan sebelumnya akan menjadi rumus seperti pada Persamaan

$$m1 \oplus m2(Z) = \frac{\sum_{X \cap Y = Z} m1(X)m2(Y)}{1 - \sum_{X \cap Y = \theta} m1(X)m2(Y)}$$

dimana:

$m1(+)$ $m2(Z)$ = mass function dari evidence (Z)

$m1(X)$ = mass function dari evidence (X)

$m2(Y)$ = mass function dari evidence (Y)

k= jumlah evidential conflict

Contoh:

$$\Theta = \{Ko, Kr, Ks, Kv\}$$

dimana:

Ko = Kanker Ovarium

Kr = Kanker Rahim

Ks = Kanker Serviks

Kv = Kanker Vulva

Andaikan seorang pasien mengalami 2 gejala :

- 1) Tercium bau amis pada vagina dengan nilai probabilitas 0,8. Gejala tersebut adalah gejala dari penyakit : Kanker Ovarium, Kanker Rahim dan Kanker Serviks.

$$m1 \{Ko, Kr, Ks\} = 0.8$$

$$m(\theta) = 1 - 0.8 = 0.2$$

- 2) Nyeri pinggang dengan nilai probabilitas 0.9. Gejala tersebut adalah gejala dari penyakit : Kanker Rahim, Kanker Serviks dan Kanker Vulva.

$$m_2 \{Kr, Ks, Kv\} = 0.9$$

$$m(\theta) = 1 - 0.9 = 0.1$$

Dengan munculnya gejala kedua yaitu nyeri pada pinggang, maka harus dilakukan penghitungan densitas baru untuk beberapa kombinasi (m_3). Untuk memudahkan perhitungan maka himpunan-himpunan bagian yang terbentuk dimasukkan ke dalam Tabel Kolom pertama diisi dengan gejala yang pertama (m_1). Sedangkan baris pertama diisi dengan gejala yang kedua (m_2). Sehingga diperoleh nilai m_3 sebagai hasil kombinasi m_1 dan m_2 .

| | | | |
|--------------------|--|---------------------|---------------------|
| | | {Kr, Ks, Kv} (0.9) | θ (0.1) |
| {Ko, Kr, Ks} (0.8) | | {Kr, Ks} (0.72) | {Ko, Kr, Ks} (0.08) |
| θ (0.2) | | {Kr, Ks, Kv} (0.18) | θ (0.02) |

Sehingga dapat dihitung :

$$m_3 \{Kr, Ks\} = \frac{0.72}{1 - 0.02} = 0.73$$

$$m_3 \{Kr, Ks, Kv\} = \frac{0.18}{1 - 0.02} = 0.18$$

$$m_3 \{Ko, Kr, Ks\} = \frac{0.08}{1 - 0.02} = 0.08$$

$$m_3 \{ \theta \} = \frac{0.02}{1 - 0.02} = 0.02$$

Dari hasil perhitungan nilai densitas m_3 kombinasi di atas dapat dilihat bahwa terdapat gejala pertama yaitu $\{Ko, Kr, Ks\} = 0.8$, namun setelah ada gejala kedua yaitu $\{Kr, Ks, Kv\} = 0.9$. Sedangkan nilai $\{Kr, Ks\} = 0.73$. Jadi dari perhitungan di atas, nilai densitas yang paling tinggi adalah $m\{Kr, Ks\} = 0.73$. Jika kemudian terdapat gejala lain yaitu :

- 3) Siklus menstruasi tidak teratur dengan nilai probabilitas 0,6. Gejala tersebut merupakan gejala penyakit Kanker Serviks.

$$m_4 = 0.6$$

$$m(\theta) = 1 - 0.6 = 0.4$$

Dengan munculnya gejala baru yaitu menstruasi tidak teratur, maka harus dilakukan perhitungan densitas baru, untuk beberapa kombinasi (m5). Untuk

| | | | |
|-------------------|--------------|------------|---------|
| | {Ks} (0.6) | θ | (0.4) |
| {Kr,Ks} (0.73) | {Ks} (0.438) | {Kr,Ks} | (0.292) |
| {Kr,Ks,Kv} (0.18) | {Ks} (0.108) | {Kr,Ks,Kv} | (0.072) |
| {Ko,Kr,Ks} (0.08) | {Ks} (0.048) | {Ko,Kr,Ks} | (0.032) |
| θ (0.02) | {Ks} (0.012) | θ | (0.008) |

memudahkan perhitungan maka himpunan-himpunan bagian yang terbentuk dimasukkan ke dalam Tabel 4.3. Kolom pertama berisi semua himpunan bagian pada m3 (1) sebagai fungsi densitas. Sedangkan baris pertama berisi semua himpunan bagian pada gejala menstruasi yang tidak teratur dengan m4 sebagai fungsi densitas. Sehingga diperoleh nilai m5 sebagai hasil m kombinasi

Sehingga dapat dihitung :

Dari hasil perhitungan nilai densitas m5 kombinasi di atas dapat dilihat bahwa dengan hanya terdapat gejala tercium bau amis dari vagina dan nyeri pada

$$\begin{aligned}
 m5 \{Ks\} &= \frac{0.438 + 0.108 + 0.048 + 0.012}{1 - 0.008} = 0.6 \\
 m5 \{Kr, Ks\} &= \frac{0.292}{1 - 0.008} = 0.294 \\
 m5 \{Kr, Ks, Kv\} &= \frac{0.072}{1 - 0.008} = 0.072 \\
 m5 \{Ko, Kr, Ks\} &= \frac{0.032}{1 - 0.008} = 0.032 \\
 m5 \{\theta\} &= \frac{0.008}{1 - 0.008} = 0.008
 \end{aligned}$$

pinggang, $m\{Ko,Kr,Ks\} = 0.08$, namun setelah ada gejala lain yaitu menstruasi tidak teratur maka nilai $m\{Ko,Kr,Ks\} = 0.032$.

Nilai $m\{Kr,Ks,Kv\} = 0.18$ setelah ada gejala menstruasi tidak teratur menjadi 0.073. Dengan perhitungan 3 gejala ini maka nilai densitas yang paling kuat adalah $m\{Ks\} = 0.6$. Dari perhitungan ketiga gejala di atas didapatkan hasil penyakit Kanker Serviks dengan nilai probabilitas 0,6 atau bila di persentasekan 60%.¹⁵

¹⁵ Ibid.

BAB VI

SISTEM PAKAR

Sistem pakar (expert system) adalah sistem yang berusaha mengapdosikan pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para ahli.

Jadi sistem pakar → Kepakaran ditransfer dari seorang pakar (atau sumber kepakaran yang lain) ke komputer, pengetahuan yang ada disimpan dalam komputer, dan pengguna dapat berkonsultasi pada komputer itu untuk suatu nasehat, lalu komputer dapat mengambil inferensi (menyimpulkan, mendeduksi, dll.) seperti layaknya seorang pakar, kemudian menjelaskannya ke pengguna tersebut, bila perlu dengan alasan-alasannya.

Sistem Pakar terkadang lebih baik unjuk kerjanya daripada seorang pakar manusia. Dengan sistem pakar, orang awam pun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli. Bagi para ahli, sistem pakar juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman.¹⁶

Sistem pakar dikembangkan pertama kali tahun 1960. Sistem pakar yang terkenal :

MYCIN

- ✓ Paling terkenal, dibuat oleh Edward Shortliffe of Stanford University tahun 70-an
- ✓ Sistem pakar medical yang bisa mendiagnosa penyakit infeksi dan merekomendasi pengobatan
- ✓ MYCIN membantu dokter mengidentifikasi pasien yang menderita penyakit. Dokter duduk di depan komputer dan memasukkan data pasien: umur, riwayat kesehatan, hasil laboratorium dan informasi terkait lainnya. Dengan informasi ini ditambah pengetahuan yang sudah ada dalam komputer, MYCIN mendiagnosa selanjutnya
- ✓ merekomendasi obat dan dosis yang harus dimakan.

¹⁶ Kusumadewi, *Artificial Intelligence (Teknik Dan Aplikasinya)*.

- ✓ MYCIN sebagai penasehat medis, tidak dimaksudkan untuk menggantikan kedudukan seorang dokter. Tetapi membantu dokter yang belum berpengalaman dalam penyakit tertentu. Juga untuk membantu dokter dalam mengkonfirmasi diagnosa dan terapi yang diberikan kepada pasien apakah sesuai dengan diagnosa dan terapi yang ada dalam basis pengetahuan yang sudah dimasukkan ke dalam MYCIN, karena MYCIN dirancang oleh dokter-dokter yang ahli di bidang penyakit tersebut.
- ✓ Kesimpulan : sistem pakar seperti MYCIN bisa digunakan sebagai bahan pembanding dalam pengambilan solusi dan pemecahan masalah. Keputusan terakhir atas pengobatan tersebut tetap menjadi tanggung jawab dokter.

DENDRAL

Mengidentifikasi struktur molekular campuran kimia yang tak dikenal

XCON & XSEL XCON

- ✓ Merupakan sistem pakar untuk membantu konfigurasi sistem komputer besar, membantu melayani order langganan sistem komputer DEC VAX 11/780 ke dalam sistem spesifikasi final yang lengkap
- ✓ Komputer besar seperti VAX terbuat dari ratusan komponen yang berbeda digabung dan disesuaikan dengan konfigurasi tertentu yang diinginkan oleh para pelanggan.
- ✓ Ada ribuan cara dimana asosiasi Pcbboard, kabel, disk drive, periperal, perangkat lunak, dan lainnya bisa dirakit ke dalam konfigurasi yang sangat rapih. Untuk mengidentifikasi hal-hal tersebut diperlukan waktu sehari-hari/berminggu-minggu agar bisa memenuhi spesifikasi yang diinginkan pemesan, tapi dengan XCON bisa dalam beberapa menit.

XSEL

- ✓ Dirancang untuk membantu karyawan bagian penjualan dalam memilih komponen sistem VAX. Karena banyaknya pilihan karyawan tersebut sering menghadapi kesulitan dalam memilih suatu komponen yang paling tepat.
- ✓ Basis pengetahuan yang ada pada XSEL membantu mengarahkan para pemesan serius untuk memilih konfigurasi yang dikehendaki, kemudian XSEL memilih CPU,

memori, periperal dan menyarankan paket software tertentu yang paling tepat dengan konfigurasinya.

PROSPECTOR

- ✓ Adalah sistem pakar yang membantu ahli geologi dalam mencari dan menemukan deposit
- ✓ Basis pengetahuan berisi bermacam-macam mineral dan batu-batuan. Banyak pakar geologi diwawancarai dan pengetahuan mereka tentang berbagai bentuk biji deposit dimasukkan ke dalam sistem pakar.
- ✓ Ahli geologi melacak biji deposit dengan pergi ke lapangan untuk meninjau medan dan mengumpulkan bukti yang ada seperti ciri-ciri geologi dicatat, sampel tanah dan batu-batuan. Sistem pakar mengevaluasi areal dalam bentuk pertanyaan dan data-data tersebut dimasukkan, kemudian Prospector memberikan rekomendasi yang menunjukkan jumlah deposit yang ada dan apakah menguntungkan atau tidak bila dieksplorasi atau di bor lebih lanjut.

DELTA

Dibuat oleh perusahaan General Electric (GE) membantu karyawan bagian pemeliharaan mesin lokomotif diesel dalam memantau mesin-mesin yang tidak berfungsi dengan baik dan membimbing ke arah prosedur perbaikan.

FOLIO

- ✓ Sistem pakar yang menolong stock broker dan tugas manajer dalam menangani investasi bagi kepentingan para langganannya. Stock broker mewawancarai langganan untuk menentukan tujuan sumber dan investasi mereka.
- ✓ FOLIO bisa memberikan rekomendasi tentang keamanan investasi, mengevaluasi stock beresiko tinggi, menghitung pengembalian modal, dan membuat keputusan dalam hal pemasaran suatu komoditi.
- ✓ Membantu para perencana keuangan untuk memperkecil kerugian karena pajak, inflasi atau faktor lain misal turun naiknya nilai mata uang.

EL

- ✓ Digunakan untuk menganalisa dan membantu rekayasa rancangan sirkuit elektronik yang terbuat dari transistor, dioda dan resistor.

- ✓ Diagram skematik dari sirkuit ini dimasukkan ke dalam komputer dan EL menganalisis menentukan karakteristik sirkuit, nilai voltase, dan strum yang ada pada semua titik sirkuit.
- ✓ Basis pengetahuan pada EL merupakan prinsip umum elektronik seperti hukum OHM, hukum kirchoff, karakteristik komponen, teori operasi transistor.¹⁷

6.1. Manfaat Sistem Pakar

- ✓ Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli
- ✓ Bisa melakukan proses secara berulang secara otomatis
- ✓ Menyimpan pengetahuan dan keahlian para pakar
- ✓ Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka)
- ✓ Mampu beroperasi dalam lingkungan yang berbahaya
- ✓ Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian. Pengguna bisa merespon dengan jawaban 'tidak tahu' atau 'tidak yakin' pada satu atau lebih pertanyaan selama konsultasi dan sistem pakar tetap akan memberikan jawaban.
- ✓ Tidak memerlukan biaya saat tidak digunakan, sedangkan pada pakar manusia memerlukan biaya sehari-hari.
- ✓ Dapat digandakan (diperbanyak) sesuai kebutuhan dengan waktu yang minimal dan sedikit biaya
- ✓ Dapat memecahkan masalah lebih cepat daripada kemampuan manusia dengan catatan menggunakan data yang sama.
- ✓ Menghemat waktu dalam pengambilan keputusan
- ✓ Meningkatkan kualitas dan produktivitas karena dapat memberi nasehat yang konsisten dan mengurangi kesalahan
- ✓ Meningkatkan kapabilitas sistem terkomputerisasi yang lain. Integrasi Sistem Pakar dengan sistem komputer lain membuat lebih efektif, dan bisa mencakup lebih banyak aplikasi .
- ✓ Mampu menyediakan pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman. Fasilitas penjelas dapat berfungsi sebagai guru.

¹⁷ Ibid.

6.2. Kelemahan Sistem Pakar

- ✓ Biaya yang diperlukan untuk membuat, memelihara, dan mengembangkannya sangat mahal
- ✓ Sulit dikembangkan, hal ini erat kaitannya dengan ketersediaan pakar di bidangnya dan kepakaran sangat sulit diekstrak dari manusia karena sangat sulit bagi seorang pakar untuk menjelaskan langkah mereka dalam menangani masalah.
- ✓ Sistem pakar tidak 100% benar karena seseorang yang terlibat dalam pembuatan sistem pakar tidak selalu benar. Oleh karena itu perlu diuji ulang secara teliti sebelum digunakan.
- ✓ Pendekatan oleh setiap pakar untuk suatu situasi atau problem bisa berbeda-beda, meskipun sama-sama benar.
- ✓ Transfer pengetahuan dapat bersifat subjektif dan bias
- ✓ Kurangnya rasa percaya pengguna dapat menghalangi pemakaian sistem pakar.

6.3. Konsep Dasar Sistem Pakar

Konsep dasar sistem pakar mengandung keahlian, ahli/pakar, pengalihan keahlian, mengambil keputusan, aturan, kemampuan menjelaskan.

➤ Keahlian

Keahlian bersifat luas dan merupakan penguasaan pengetahuan dalam bidang khusus yang diperoleh dari pelatihan, membaca atau pengalaman. Contoh bentuk pengetahuan yang termasuk keahlian :

- ✓ Teori, fakta, aturan-aturan pada lingkup permasalahan tertentu
- ✓ Strategi global untuk menyelesaikan masalah

➤ Ahli / Pakar

Seorang ahli adalah seseorang yang mampu menjelaskan suatu tanggapan, mempelajari hal-hal baru seputar topik permasalahan, menyusun kembali pengetahuan jika dipandang perlu, memecahkan masalah dengan cepat dan tepat

➤ Pengalihan keahlian

Tujuan dari sistem pakar adalah untuk mentransfer keahlian dari seorang pakar ke dalam komputer kemudian ke masyarakat. Proses ini meliputi 4 kegiatan, yaitu perolehan pengetahuan (dari para ahli atau sumber-sumber lainnya), representasi

pengetahuan ke komputer, kesimpulan dari pengetahuan dan pengalihan pengetahuan ke pengguna.

➤ Mengambil keputusan

Hal yang unik dari sistem pakar adalah kemampuan untuk menjelaskan dimana keahlian tersimpan dalam basis pengetahuan. Kemampuan komputer untuk mengambil kesimpulan dilakukan oleh komponen yang dikenal dengan mesin inferensi yaitu meliputi prosedur tentang pemecahan masalah.

➤ Aturan

Sistem pakar yang dibuat merupakan sistem yang berdasarkan pada aturan – aturan dimana program disimpan dalam bentuk aturan-aturan sebagai prosedur pemecahan masalah. Aturan tersebut biasanya berbentuk IF – THEN.

➤ Kemampuan menjelaskan

Keunikan lain dari sistem pakar adalah kemampuan dalam menjelaskan atau memberi saran/rekomendasi serta juga menjelaskan mengapa beberapa tindakan/saran tidak direkomendasikan.

6.4. Perbedaan Sistem Konvensional Dengan Sistem Pakar

| Sistem Konvensional | Sistem Pakar |
|---|--|
| Informasi dan pemrosesannya biasanya jadi satu dengan program | Basis pengetahuan merupakan bagian terpisah dari mekanisme inferensi |
| Program tidak pernah salah (kecuali pemrogramnya yang salah) | Program bisa saja melakukan kesalahan |
| Biasanya tidak bisa menjelaskan mengapa suatu input data itu dibutuhkan atau bagaimana output itu diperoleh | Penjelasan adalah bagian terpenting dari sistem pakar |
| Pengubahan program cukup sulit dan merepotkan | Pengubahan pada aturan/kaidah dapat dilakukan dengan mudah |
| Sistem hanya akan bekerja jika sistem tersebut sudah lengkap | Sistem dapat bekerja hanya dengan beberapa aturan |
| Eksekusi dilakukan langkah demi langkah secara algoritmik | Eksekusi dilakukan pada keseluruhan basis pengetahuan secara heuristik dan logis |
| Menggunakan data | Menggunakan pengetahuan |
| Tujuan utamanya adalah efisiensi | Tujuan utamanya adalah efektivitas |

6.5. Elemen Yang Terkait Pengembangan Sistem Pakar

1) Pakar

Pakar adalah orang yang memiliki pengetahuan khusus, pendapat, pengalaman dan metode, serta kemampuan untuk mengaplikasikan keahliannya tersebut guna menyelesaikan masalah.

2) Perakayasa pengetahuan

Perakayasa pengetahuan adalah orang yang membantu pakar dalam menyusun area permasalahan dengan menginterpretasikan dan mengintegrasikan jawaban-jawaban pakar atas pertanyaan yang diajukan, menggambarkan analogi, mengajukan counter example dan menerangkan kesulitan-kesulitan konseptual.

3) Pemakai

- ✓ Pemakai awam : dalam hal ini sistem pakar bertindak sebagai konsultan untuk memberikan saran dan solusi kepada pemakai
- ✓ Pelajar yang ingin belajar : sistem pakar bertindak sebagai instruktur
- ✓ Pembuat sistem pakar : sistem pakar sebagai partner dalam pengembangan basis pengetahuan.
- ✓ Pakar : sistem pakar bertindak sebagai mitra kerja/asisten

6.6. Area Permasalahan Aplikasi Sistem Pakar

1) Interpretasi

Yaitu pengambilan keputusan dari hasil observasi, diantaranya : pengawasan, pengenalan ucapan, analisis citra, interpretasi sinyal, dan beberapa analisis kecerdasan

2) Prediksi

Memprediksi akibat-akibat yang dimungkinkan dari situasi-situasi tertentu, diantaranya : peramalan, prediksi demografis, peralihan ekonomi, prediksi lalu lintas, estimasi hasil, militer, pemasaran, atau peramalan keuangan.

3) Diagnosis

Menentukan sebab malfungsi dalam situasi kompleks yang didasarkan pada gejala-gejala yang teramati, diantaranya : medis, elektronis, mekanis, dan diagnosis perangkat lunak

4) Desain

Menentukan konfigurasi komponen-komponen sistem yang cocok dengan tujuan-tujuan kinerja tertentu dan kendala-kendala tertentu, diantaranya : layout sirkuit, perancangan bangunan

5) Perencanaan

Merencanakan serangkaian tindakan yang akan dapat mencapai sejumlah tujuan dengan kondisi awal tertentu, diantaranya : perencanaan keuangan, komunikasi, militer, pengembangan politik, routing dan manajemen proyek.

6) Monitoring

Membandingkan tingkah laku suatu sistem yang teramati dengan tingkah laku yang diharapkan darinya, diantaranya : Computer Aided Monitoring System

7) Debugging dan repair

Menentukan dan mengimplementasikan cara-cara untuk mengatasi malfungsi, diantaranya memberikan resep obat terhadap suatu kegagalan.

8) Instruksi

Melakukan instruksi untuk diagnosis, debugging dan perbaikan kinerja.

9) Kontrol

Mengatur tingkah laku suatu environment yang kompleks seperti kontrol terhadap interpretasi-interpretasi, prediksi, perbaikan, dan monitoring kelakuan sistem

10)Seleksi

Mengidentifikasi pilihan terbaik dari sekumpulan (list) kemungkinan.

11)Simulasi

Pemodelan interaksi antara komponen-komponen sistem.

6.7. Bentuk / Tipe Sistem Pakar

- 1) Mandiri : sistem pakar yang murni berdiri sendiri, tidak digabung dengan software lain, bisa dijalankan pada komputer pribadi, mainframe.
- 2) Terkait/Tergabung : dalam bentuk ini sistem pakar hanya merupakan bagian dari program yang lebih besar. Program tersebut biasanya menggunakan teknik

algoritma konvensional tapi bisa mengakses sistem pakar yang ditempatkan sebagai subrutin, yang bisa dimanfaatkan setiap kali dibutuhkan.

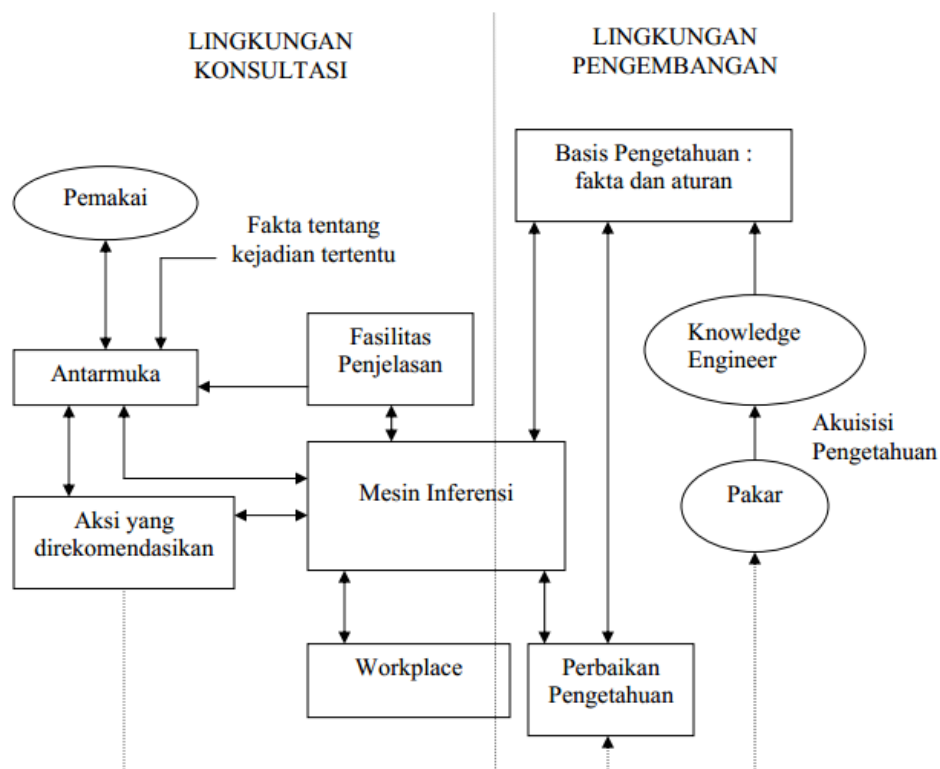
- 3) Terhubung : merupakan sistem pakar yang berhubungan dengan software lain, misal : spreadsheet, DBMS, program grafik. Pada saat proses inferensi, sistem pakar bisa mengakses data dalam spreadsheet atau DBMS atau program grafik bisa dipanggil untuk menayangkan output visual.
- 4) Sistem Mengabdikan: Merupakan bagian dari komputer khusus yang diabdikan kepada fungsi tunggal. Sistem tersebut bisa membantu analisa data radar dalam pesawat tempur atau membuat keputusan intelejen tentang bagaimana memodifikasi pembangunan kimiawi, dll.

6.8. Struktur Sistem Pakar

2 bagian utama sistem pakar :

- ✓ lingkungan pengembangan (development environment) : digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar
- ✓ lingkungan konsultasi (consultation environment) digunakan oleh pengguna yang bukan pakar untuk memperoleh pengetahuan pakar

6.9. Arsitektur Sistem Pakar



Komponen yang terdapat dalam arsitektur/struktur sistem pakar :

1) Antarmuka Pengguna (User Interface)

Merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi. Antarmuka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu antarmuka menerima dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai.

2) Basis Pengetahuan

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi, dan penyelesaian masalah. Komponen sistem pakar ini disusun atas 2 elemen dasar, yaitu :

- fakta : informasi tentang obyek dalam area permasalahan tertentu
- aturan : informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui.

3) Akuisisi Pengetahuan (Knowledge Acquisition)

Akuisisi pengetahuan adalah akumulasi, transfer, dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program komputer. Dalam tahap ini knowledge engineer berusaha menyerap pengetahuan untuk selanjutnya ditransfer ke dalam basis pengetahuan. Pengetahuan diperoleh dari pakar, dilengkapi dengan buku, basis data, laporan penelitian dan pengalaman pemakai. Metode akuisisi pengetahuan :

- Wawancara

Metode yang paling banyak digunakan, yang melibatkan pembicaraan dengan pakar secara langsung dalam suatu wawancara

- Analisis protokol

Dalam metode ini pakar diminta untuk melakukan suatu pekerjaan dan mengungkapkan proses pemikirannya dengan menggunakan kata-kata. Pekerjaan tersebut direkam, dituliskan, dan dianalisis.

- Observasi pada pekerjaan pakar

Pekerjaan dalam bidang tertentu yang dilakukan pakar direkam dan diobservasi

- Induksi aturan dari contoh

Induksi adalah suatu proses penalaran dari khusus ke umum. Suatu sistem induksi aturan diberi contoh-contoh dari suatu masalah yang hasilnya telah diketahui. Setelah diberikan beberapa contoh, sistem induksi aturan tersebut dapat membuat aturan yang benar untuk kasus-kasus contoh. Selanjutnya aturan dapat digunakan untuk menilai kasus lain yang hasilnya tidak diketahui.

4) Mesin/Motor Inferensi (inference engine)

Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin inferensi adalah program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam workplace, dan untuk memformulasikan kesimpulan.

5) Workplace / Blackboard

Workplace merupakan area dari sekumpulan memori kerja (working memory), digunakan untuk merekam kejadian yang sedang berlangsung termasuk keputusan sementara. Ada 3 keputusan yang dapat direkam :

- Rencana : bagaimana menghadapi masalah
- Agenda : aksi-aksi yang potensial yang sedang menunggu untuk dieksekusi
- Solusi : calon aksi yang akan dibangkitkan

6) Fasilitas Penjelasan

Adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Digunakan untuk melacak respon dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif melalui pertanyaan :

- mengapa suatu pertanyaan ditanyakan oleh sistem pakar ?
- bagaimana konklusi dicapai ?
- mengapa ada alternatif yang dibatalkan ?
- rencana apa yang digunakan untuk mendapatkan solusi ?

7) Perbaikan Pengetahuan

Pakar memiliki kemampuan untuk menganalisis dan meningkatkan kinerjanya serta kemampuan untuk belajar dari kinerjanya. Kemampuan tersebut adalah penting dalam pembelajaran terkomputerisasi, sehingga program akan mampu menganalisis penyebab kesuksesan dan kegagalan yang dialaminya dan juga mengevaluasi apakah pengetahuan-pengetahuan yang ada masih cocok untuk digunakan di masa mendatang

6.10. Basis Pengetahuan (Knowledge Base)

Basis pengetahuan berisi pengetahuan-pengetahuan dalam penyelesaian masalah.

Ada 2 bentuk pendekatan basis pengetahuan :

1) Penalaran berbasis aturan (rule-based reasoning)

Pada penalaran berbasis aturan, pengetahuan direpresentasikan dengan menggunakan aturan berbentuk IF-THEN. Bentuk ini digunakan apabila kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu, dan si pakar dapat menyelesaikan masalah tersebut secara berurutan. Disamping itu, bentuk ini juga digunakan apabila dibutuhkan penjelasan tentang jejak (langkah-langkah) pencapaian solusi.

Contoh : aturan identifikasi hewan

Rule 1 : IF hewan berambut dan menyusui THEN hewan mamalia

Rule 2 : IF hewan mempunyai sayap dan bertelur THEN hewan jenis burung

Rule 3 : IF hewan mamalia dan memakan daging THEN hewan

karnivora Dst...

2) Penalaran berbasis kasus (case-based reasoning)

Pada penalaran berbasis kasus, basis pengetahuan akan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang terjadi sekarang (fakta yang ada). Bentuk ini digunakan apabila user menginginkan untuk tahu lebih banyak lagi pada kasus-kasus yang hampir sama (mirip). Selain itu bentuk ini juga digunakan bila kita telah memiliki sejumlah situasi atau kasus tertentu dalam basis pengetahuan.

6.11. Mesin Inferensi (Inference Engine)

Ada 2 cara penalaran yang dapat dikerjakan dalam melakukan inferensi :

1) Forward Chaining

Pencocokan fakta atau pernyataan dimulai dari bagian sebelah kiri dulu (IF dulu). Dengan kata lain penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis.

2) Backward Chaining

Pencocokan fakta atau pernyataan dimulai dari bagian sebelah kanan (THEN dulu). Dengan kata lain penalaran dimulai dari hipotesis terlebih dahulu, dan untuk menguji kebenaran hipotesis tersebut harus dicari fakta-fakta yang ada dalam basis pengetahuan.

Langkah-Langkah Pembuatan Sistem Pakar

- 1) Mengidentifikasi masalah dan kebutuhan
- 2) Menentukan problema yang cocok
- 3) mempertimbangkan alternatif
- 4) menghitung pengembalian investasi
- 5) memilih alat pengembangan
- 6) merencanakan pengetahuan
- 7) merancang sistem
- 8) melengkapi pengembangan
- 9) menguji dan mencari kesalahan sistem
- 10) memelihara sistem ¹⁸

¹⁸ Ibid.

BAB VII

FUZZY LOGIC

7.1. Pengertian Fuzzy Logic

Konsep *Fuzzy Logic* pertama kali diperkenalkan pada tahun 1962 oleh seorang professor dari University of California yang bernama Lotfi. A. Zadeh¹⁹. *Fuzzy* secara bahasa diartikan sebagai kabur atau samar samar. Suatu nilai dapat bernilai besar atau salah secara bersamaan. Dalam fuzzy dikenal derajat keanggotaan yang memiliki rentang nilai 0 (nol) hingga 1(satu). Berbeda dengan himpunan tegas yang memiliki nilai 1 atau 0 (ya atau tidak).

Fuzzy Logic atau selanjutnya disebut dengan logika fuzzy merupakan suatu logika yang memiliki nilai kekaburan atau kesamaran (fuzzyness) antara benar atau salah. Dalam teori logika fuzzy suatu nilai bias bernilai benar atau salah secara bersama. Namun berapa besar keberadaan dan kesalahan suatu tergantung pada bobot keanggotaan yang dimilikinya. Logika fuzzy memiliki derajat keanggotaan dalam rentang 0 hingga 1. Berbeda dengan logika digital yang hanya memiliki dua nilai 1 atau 0. Logika fuzzy digunakan untuk menterjemahkan suatu besaran yang diekspresikan menggunakan bahasa (linguistic), misalkan besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat, dan sangat cepat. Dan logika fuzzy menunjukkan sejauh mana suatu nilai itu benar dan sejauh mana suatu nilai itu salah. Tidak seperti logika klasik (scrisp)/tegas, suatu nilai hanya mempunyai 2 kemungkinan yaitu merupakan suatu anggota himpunan atau tidak. Derajat keanggotaan 0 (nol) artinya nilai bukan merupakan anggota himpunan dan 1 (satu) berarti nilai tersebut adalah anggota himpunan.

Lotfi Zadeh mengatakan Integrasi Logika Fuzzy kedalam sistem informasi dan rekayasa proses adalah menghasilkan aplikasi seperti sistem kontrol, alat alat rumah tangga, dan sistem pengambil keputusan yang lebih fleksibel, mantap, dan canggih dibandingkan dengan sistem konvensional. Dalam hal ini kami dapat mengatakan bahwa logika fuzzy memimpin dalam pengembangan kecerdasan mesin yang lebih tinggi (machine Intelligency Quotient / MIQ) Produk produk berikut telah menggunakan logika fuzzy dalam alat alat rumah tangga seperti mesin cuci, video dan kamera refleksi lensa tunggal, pendingin ruangan, oven microwave, dan banyak

¹⁹ T. Sutojo, E. Mulyanto and V. Suhartono, Kecerdasan Buatan, Yogyakarta: Penerbit ANDI, 2011.

sistem diagnosa mandiri. Fuzzy Logic merupakan kecerdasan buatan yang pertama kali dipublikasikan oleh Prof.Dr. Lotfi Zadeh yang berasal dari Pakistan. Melalui fuzzy logic ini sistem dapat membuat keputusan sendiri dan terkesan seperti memiliki perasaan, karena memiliki keputusan lain selain iya (logika 1) dan tidak (logika 0). Oleh karena itu fuzzy logic sangat berbeda jauh dari alur algoritma pemrograman.

7.2. Kelebihan Fuzzy Logic

Logika fuzzy memiliki beberapa keunggulan, antara lain sebagai berikut:

1. Konsep logika fuzzy mudah dimengerti. Konsep matematis yang mendasari penalaran logika fuzzy sangat sederhana dan mudah dimengerti.
2. Logika fuzzy sangat fleksibel.
3. Logika fuzzy memiliki toleransi terhadap data-data yang tidak tepat.
4. Logika fuzzy mampu memodelkan fungsi nonlinear yang kompleks.
5. Logika fuzzy dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
6. Logika fuzzy dapat bekerja sama dengan teknik-teknik kendali secara konvensional.
7. Logika fuzzy didasarkan pada bahas alami.

Sementara itu, dalam pengaplikasiannya, logika fuzzy juga memiliki beberapa kelebihan, antara lain sebagai berikut:

1. Daya gunanya dianggap lebih baik daripada teknik kendali yang pernah ada.
2. Pengendali fuzzy terkenal karena keandalannya.
3. Mudah diperbaiki.
4. Pengendali fuzzy memberikan pengendalian yang sangat baik dibandingkan teknik lain
5. Usaha dan dana yang dibutuhkan kecil.

Selain itu, logika fuzzy juga memiliki kekurangan, terutama dalam penerapannya. Kekurangan-kekurangan tersebut antara lain:

1. Para enjiner dan ilmuwan generasi sebelumnya dan sekarang banyak yang tidak mengenal teori kendali fuzzy, meskipun secara teknik praktis mereka

memiliki pengalaman untuk menggunakan teknologi dan perkakas kontrol yang sudah ada.

2. Belum banyak terdapat kursus/balai pendidikan dan buku-buku teks yang menjangkau setiap tingkat pendidikan (undergraduate, postgraduate, dan on site training)
3. Hingga kini belum ada pengetahuan sistematik yang baku dan seragam tentang metodologi pemecahan problema kendali menggunakan pengendali fuzzy.
4. Belum adanya metode umum untuk mengembangkan dan implementasi pengendali fuzzy.²⁰

7.3. Fungsi Keanggotaan

Dalam logika tegas, fungsi keanggotaan menyatakan keanggotaan pada suatu himpunan. Fungsi keanggotaan $\chi_A(x)$ bernilai 1 jika x anggota himpunan A , dan bernilai 0 jika x bukan anggota himpunan A . Jadi, fungsi keanggotaan ini hanya bisa bernilai 0 atau 1.

$$\chi_A : x \rightarrow \{0,1\} \quad (1)$$

Sedangkan dalam logika fuzzy, fungsi keanggotaan menyatakan derajat keanggotaan pada suatu himpunan. Nilai dari fungsi keanggotaan ini berada dalam selang $[0,1]$, dan dinyatakan dengan μ_A .

$$\mu_A : x \rightarrow [0,1] \quad (2)$$

Fungsi keanggotaan $\mu_A(x)$ bernilai 1 jika x anggota penuh himpunan A , dan bernilai 0 jika x bukan anggota himpunan A . Sedangkan jika derajat keanggotaan berada dalam selang $(0,1)$, misalnya $\mu_A(x) = \mu$, menyatakan x sebagian anggota himpunan A dengan derajat keanggotaan sebesar μ . Ada 3 cara mendefinisikan himpunan fuzzy:

1. Sebagai himpunan pasangan berurutan Misalkan himpunan fuzzy A didefinisikan dalam semesta $X = \{x_1, x_2, \dots, x_n\}$, maka himpunan pasangan berurutan yang menyatakan himpunan fuzzy-nya adalah

$$A = \{ (x_1, \mu_A(x_1)), (x_2, \mu_A(x_2)), \dots, (x_n, \mu_A(x_n)) \} \quad (3)$$

²⁰ Athia Saelan, "Logika Fuzzy," *Struktur Diskrit* 1, no. 13508029 (2009): 1–5.

Cara ini hanya dapat digunakan pada himpunan fuzzy yang anggotanya bernilai diskrit.

2. Dengan menyebut fungsi keanggotaan Misalkan himpunan fuzzy A didefinisikan dalam semesta X yang anggotanya bernilai kontinu, maka himpunan pasangan berurutan yang menyatakan himpunan fuzzy-nya adalah

$$A = \{(x, \mu_A(x)) \mid \mu_A(x) = \dots, x \in X\} \quad (4)$$

3. Menuliskan sebagai

$$A = \{\mu_A(x_1)/x_1 + \mu_A(x_2)/x_2 + \dots + \mu_A(x_n)/x_n\} = \{\sum_{i=1}^n \mu_A(x_i)/x_i\} \quad (5)$$

untuk X diskrit, atau

$$A = \{\int_X \mu_A(x)/x\} \quad (6)$$

Untuk X kontinu. Lambang \int bukan berarti integral.

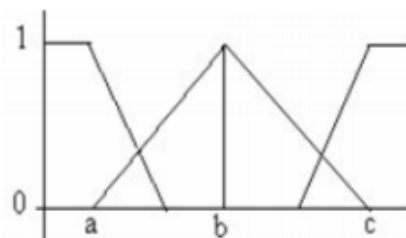
Fungsi keanggotaan suatu himpunan fuzzy dapat ditentukan dengan fungsi segitiga (triangel), trapesium (trapezoidal), atau Fungsi Gauss (Gaussian).

1. Fungsi keanggotaan segitiga

Persamaan fungsi keanggotaan segitiga adalah

$$\mu(x; a, b, c) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b < x \leq c \\ 1 & x > c \end{cases} \quad (7)$$

Persamaan tersebut direpresentasikan dalam bentuk grafik sebagai berikut.

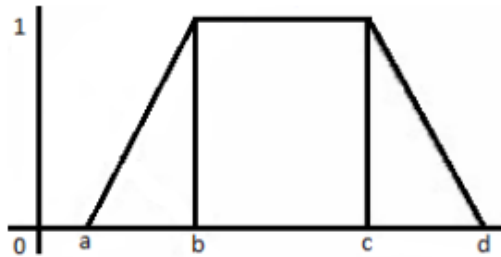


2. Fungsi keanggotaan trapesium

Persamaan fungsi keanggotaan segitiga adalah

$$\mu(x; a, b, c, d) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \leq x < b \\ 1 & b \leq x \leq c \\ \frac{d-x}{d-c} & c < x \leq d \\ 0 & x > d \end{cases} \quad (8)$$

Persamaan tersebut direpresentasikan dalam bentuk grafik sebagai berikut.

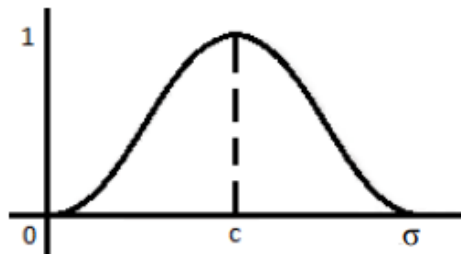


3. Fungsi keanggotaan Gaussian

Persamaan fungsi keanggotaan segitiga adalah

$$\mu(x; c, \sigma) = \frac{1}{1 + \left(\frac{x-c}{\sigma}\right)^2} \quad (9)$$

Persamaan tersebut direpresentasikan dalam bentuk grafik sebagai berikut.



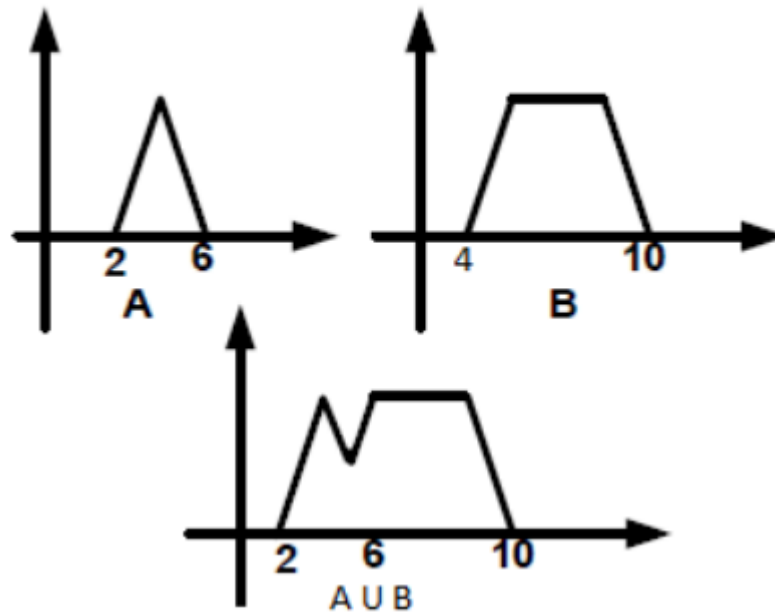
7.4. Operasi Logika Fuzzy

Operasi-operasi yang dapat dilakukan dalam logika dan himpunan fuzzy sama dengan dalam logika dan himpunan biasa. Namun definisinya agak berbeda.

1. Gabungan

Gabungan antara himpunan A dan himpunan B dapat diartikan sebagai himpunan yang dekat dengan A atau dekat dengan B.

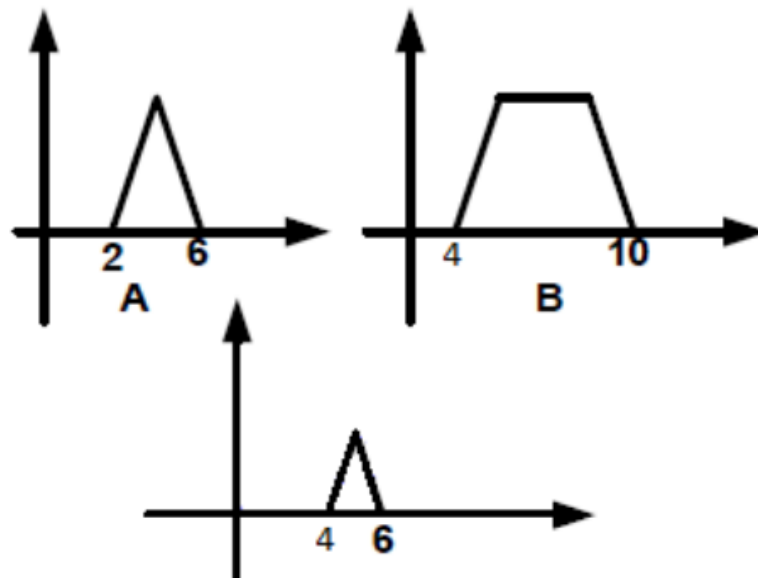
$$A \cup B \rightarrow \mu_{A \cup B} = \mu_A(x) \vee \mu_B(x) = \max(\mu_A(x), \mu_B(x))$$



2. Irisan

Irisan antara himpunan A dan himpunan B dapat diartikan sebagai himpunan yang dekat dengan A dan dekat dengan B.

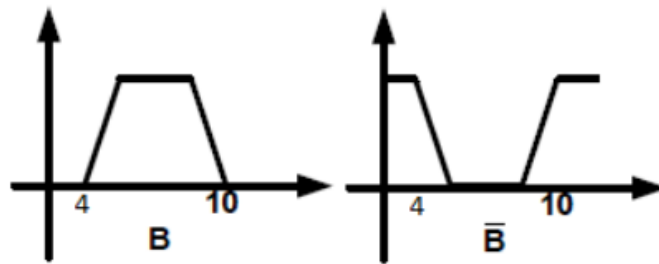
$$A \cap B \rightarrow \mu_{A \cap B} = \mu_A(x) \wedge \mu_B(x) = \min(\mu_A(x), \mu_B(x)) \quad (10)$$



3. Komplemen

Komplemen dari himpunan A dapat diartikan sebagai himpunan yang tidak dekat dengan A.

$$\bar{A} \rightarrow \mu_{\bar{A}} = 1 - \mu_A(x) \quad (11)$$

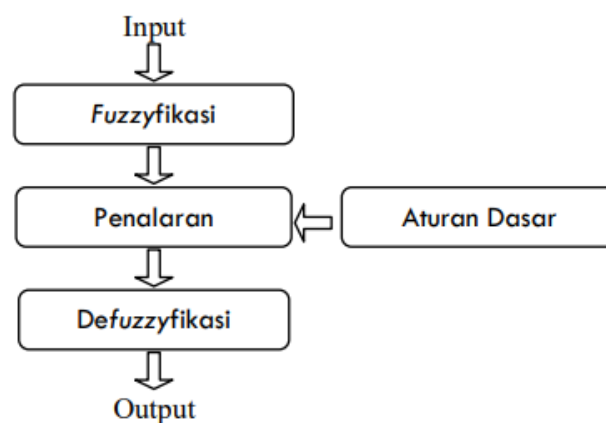


7.5. Kendali Logika Fuzzy

Sistem kendali logika fuzzy disebut juga sistem Inferensi Fuzzy (Fuzzy Inference System/FIS) atau fuzzy inference engine adalah sistem yang dapat melakukan penalaran dengan prinsip serupa seperti manusia melakukan penalaran dengan nalurinya.

Terdapat beberapa jenis FIS yang dikenal yaitu Mamdani, Sugeno dan Tsukamoto. FIS yang paling mudah dimengerti, karena paling sesuai dengan naluri manusia adalah FIS Mamdani. FIS tersebut bekerja berdasarkan kaidah-kaidah linguistik dan memiliki algoritma fuzzy yang menyediakan sebuah aproksimasi untuk dimasuki analisa matematik.

Sistem kendali logika fuzzy terdiri dari beberapa tahapan seperti pada diagram berikut.



Proses dalam kendali logika fuzzy ditunjukkan pada Gambar di atas. Input yang diberikan kepada adalah berupa bilangan tertentu dan output yang dihasilkan juga harus berupa bilangan tertentu. Aturan-aturan dalam bahasa linguistik dapat digunakan sebagai input yang bersifat teliti harus dikonversikan terlebih dahulu, lalu melakukan penalaran berdasarkan aturan-aturan dan mengkonversi hasil penalaran tersebut menjadi output yang bersifat teliti.

7.5.1. Fuzzyfikasi

Fuzzyfikasi adalah pemetaan nilai input yang merupakan nilai tegas ke dalam fungsi keanggotaan himpunan fuzzy, untuk kemudian diolah di dalam mesin penalaran.

$$\text{fuzzyfikasi : } x \rightarrow \mu(x) \quad (12)$$

7.5.2. Aturan Dasar

Aturan dasar dalam kendali logika fuzzy adalah aturan implikasi dalam bentuk “jika ... maka ...”. Aturan dasar tersebut ditentukan dengan bantuan seorang pakar yang mengetahui karakteristik objek yang akan dikendalikan.

Contoh bentuk implikasi yang digunakan adalah sebagai berikut.

$$\text{Jika } X = A \text{ dan } Y = B \text{ maka } Z = C$$

7.5.3. Penalaran

Pada tahapan ini sistem menalar nilai masukan untuk menentukan nilai keluaran sebagai bentuk pengambil keputusan. Sistem terdiri dari beberapa aturan, maka kesimpulan diperoleh dari kumpulan dan korelasi antar aturan.

Ada 3 metode yang digunakan dalam melakukan inferensi sistem fuzzy, yaitu max, additive dan probabilistik OR.

Pada metode max, solusi himpunan fuzzy diperoleh dengan cara mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah fuzzy, dan mengaplikasikannya ke output dengan menggunakan operator OR (union). Secara umum dapat ditulis

$$\mu_{df}(x_i) \leftarrow \max (\mu_{df}(x_i), \mu_{kf}(x_i)) \quad (13)$$

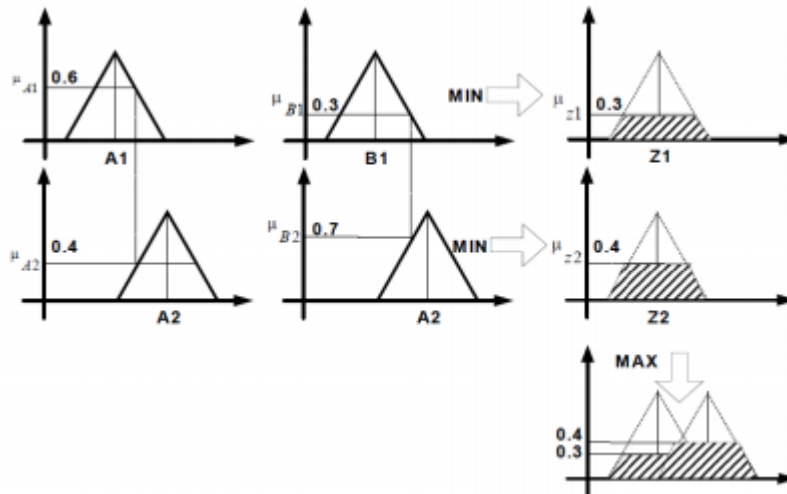
Selain itu, salah satu model penalaran yang banyak digunakan adalah max-min. Dalam penalaran ini, pertama-tama dilakukan proses operasi min sinyal keluaran lapisan fuzzyfikasi, kemudian diteruskan dengan operasi max untuk mencari nilai keluaran yang selanjutnya akan difuzzyfikasikan sebagai bentuk keluaran pengendali. Operasi max-min tersebut dapat dinyatakan sebagai berikut.

Operasi min atau irisan

$$\begin{aligned} a \wedge b &= \min(a, b) = a \text{ if } a \leq b \\ &= b \text{ if } a > b \end{aligned} \quad (14)$$

Operasi max atau gabungan

$$\begin{aligned} a \vee b &= \max(a, b) = a \text{ if } a \geq b \\ &= b \text{ if } a < b \end{aligned} \quad (15)$$



7.6. Defuzzyfikasi

Defuzzyfikasi merupakan kebalikan dari fuzzyfikasi, yaitu pemetaan dari himpunan fuzzy ke himpunan tegas. Input dari proses defuzzyfikasi adalah suatu himpunan fuzzy yang diperoleh dari komposisi aturanaturan fuzzy. Hasil dari defuzzyfikasi ini merupakan output dari sistem kendali logika fuzzy.

Defuzzyfikasi dideskripsikan sebagai

$$Z^* = \text{defuzzyfier}(Z)$$

dengan

Z = hasil penalaran fuzzy

Z^* = keluaran kendali logika fuzzy

defuzzyfier = fungsi defuzzyfikasi

Metode defuzzyfikasi antara lain:

1. Metode Maximum

Metode ini juga dikenal dengan metode puncak, yang nilai keluarannya dibatasi oleh fungsi $\mu_c(z^*) > \mu_c(z)$.

2. Metode titik tengah

Metode titik tengah juga disebut metode pusat area. Metode ini lazim dipakai dalam proses defuzzifikasi. Keluaran dari metode ini adalah titik tengah dari hasil proses penalaran.

3. Metode rata-rata

Metode ini digunakan untuk fungsi keanggotaan keluaran yang simetris. Keluaran dari metode ini adalah nilai rata-rata dari hasil proses penalaran.

4. Metode penjumlahan titik tengah

Keluaran dari metode ini adalah penjumlahan titik tengah dari hasil proses penalaran.

5. Metode titik tengah area terbesar

Dalam metode ini, keluarannya adalah titik pusat dari area terbesar yang ada.

7.7. Aplikasi Fuzzy Logic

Jika diamati pengalaman pada negara-negara berteknologi tinggi, khususnya di negara Jepang, pengendali fuzzy sudah sejak lama dan luas digunakan di industri-industri dan alat-alat elektronika. Beberapa contoh aplikasi yang menggunakan pengendali fuzzy antara lain:

- ✓ Dalam teknologi otomotif : sistem transmisi otomatis fuzzy dan pengendali kecepatan idle fuzzy.
- ✓ Dalam teknologi transportasi :
Pengendali fuzzy anti-slip untuk kereta listrik, sistem pengaturan dan perencanaan parkir, sistem pengaturan lampu lalu lintas, dan pengendalian kecepatan kendaraan di jalan bebas hambatan.
- ✓ Dalam peralatan sehari-hari : mesin cuci fuzzy dan vacuum cleaner fuzzy dan lain-lain.
- ✓ Dalam aplikasi industri di antaranya : industri kimia, sistem pengolahan kertas, dan lain-lain.
- ✓ Dalam power stations : sistem diagnosis kebocoran-H₂²¹

²¹ Ibid.

BAB VIII

MACHINE LEARNING

Learning Machine adalah suatu aplikasi dalam AI yang memiliki kemampuan beradaptasi dengan dunia luar dan dapat memanfaatkan informasi dari dunia luar untuk menambah pengetahuan dan meningkatkan kemampuannya.

Kata mesin digunakan untuk membedakan dengan manusia (mahluk hidup) yang secara alami memiliki kemampuan belajar.

8.1. Rote Learning

Metode learning ini menggunakan hasil penelusuran atau hasil perhitungan sebelumnya yang tersimpan dalam cache memori komputer untuk menentukan strategi ke langkah berikutnya.

Metode ini memiliki kemampuan untuk :

- 1) Mengorganisir penyimpanan informasi adalah lebih cepat mengambil nilai yang sudah tersimpan dari pada menghitung ulang
- 2) Generalisasi hal ini akan mencegah terlalu besarnya informasi atau nilai yang disimpan

8.2. Learning by Taking Advice

Metode learning ini menggunakan advice tingkat tinggi (dalam bahasa manusia) untuk menghasilkan suatu aturan operasional.

Advice mana yang akan digunakan dari sekian banyak yang ada diproses/dipilih menggunakan operator-operator seperti : analisis kasus, pencocokan, dsb

8.3. Learning from example

Metode ini menggunakan semua contoh dari kasus-kasus yang pernah diselesaikan atau data yang dimasukkan ke sistem.

Hal terpenting dari metode ini klasifikasi, untuk memilah atau mengklasifikasi menjadi positif dan negatif. Hasil dari metode ini adalah suatu deskripsi konsep.

Metode ini menggunakan Algoritma search untuk mengeliminasi dan menghasilkan pohon keputusan.

8.4. Learning in Problem Solving

Metode ini berusaha untuk memperbaiki pemecahan masalah dari pemecahan masalah yang sudah ada atau sudah pernah diaplikasikan.

Metode ini menggunakan solusi dari contoh masalah sebagai masukan dan akan menghasilkan penemuan cara baru untuk menyelesaikan masalah secara lebih efisien.

Metode ini menggunakan heuristic search seperti : generalisasi, learning berdasarkan penjelasan dan pertimbangan yang menyeluruh.

8.5. Discovery

Metode ini berusaha untuk menemukan pengetahuan-pengetahuan baru yang belum terungkap sebelumnya.

Metode ini menggunakan heuristic search yang berdasarkan kepada analogy, ketertarikan (minat) atau bahkan suatu misteri.

Hasil atau keluaran dari metode ini cenderung tidak diketahui atau sulit diperkirakan, karena biasanya berdasarkan informasi atau pengetahuan yang minim. ²²

²² Kusumadewi, *Artificial Intelligence (Teknik Dan Aplikasinya)*.

BAB IX

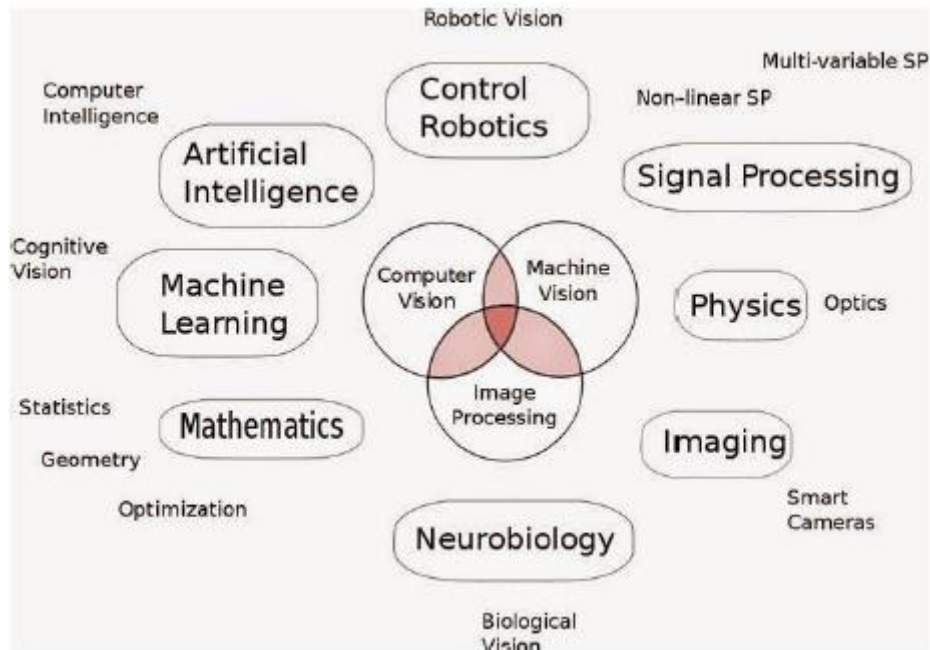
COMPUTER VISION DAN APLIKASINYA

Dalam istilah sederhana, Computer Vision adalah bagaimana komputer/mesin dapat melihat. Computer vision adalah bidang yang mencakup metode untuk memperoleh, mengolah, menganalisis, dan memahami data visual seperti gambar dan video. Tujuan utama dari Computer Vision adalah agar komputer atau mesin dapat meniru kemampuan perseptual mata manusia dan otak, atau bahkan dapat mengunggulinya untuk tujuan tertentu.

“For Your Information, Perseptual adalah kemampuan memahami dan menginterpretasikan informasi sensorik atau kemampuan intelek untuk mencari makna yang diterima oleh panca indera.”

Bidang yang berkaitan erat dengan computer vision adalah image processing (pengolahan citra) dan machine vision (visi mesin). Ada tumpang tindih yang signifikan dalam berbagai teknik dan aplikasi yang mencakup tiga bidang tersebut. Hal ini menunjukkan teknik dasar yang digunakan dan dikembangkan kurang lebih sama (identik). Computer vision mencakup teknologi utama untuk menganalisis citra (visual) secara otomatis yang digunakan dalam bidang lain. Sedangkan machine vision biasanya mengacu pada proses menggabungkan analisis citra otomatis dengan metode dan/atau teknologi lain baik berupa software maupun hardware untuk mencapai tujuan tertentu.

Secara luas computer vision berhubungan dan dapat diterapkan/dikombinasikan dengan bidang lain seperti artificial intelligence (kecerdasan buatan), robotika, otomasi industri, pengolahan sinyal, optic fisik, neurobiology, dll.



Dalam zaman yang semakin modern semakin banyak aplikasi dari computer vision yang dapat ditemui pada berbagai bidang, seperti:

- ✓ Industri : biasanya CV digunakan membantu proses otomasi industri, misalnya untuk quality control dimana computer vision berfungsi untuk melakukan pemeriksaan akhir terhadap produk untuk menemukan apakah ada cacat produksi. Dalam bidang industri terkadang disebut sebagai machine vision karena biasanya dihubungkan dengan perangkat lain.
- ✓ Robotika : digunakan untuk mengenali lingkungan sekitar, misalnya mobile robot, lengan robot yang dapat mengenali objek, drone robot (pesawat robot tanpa awak), robot humanoid, robot penjelajah, dll.
- ✓ Otomotif : misalnya pada autonomus vehicle, fitur keselamatan rem otomatis untuk menghindari tabrakan dengan penyeberang jalan ataupun objek lain.
- ✓ Pendeteksi : misalnya alat presensi yang dapat mengenali wajah, alat pendeteksi tanda tangan, alat penghitung jumlah kendaraan yang melintasi jalan, dll.
- ✓ Medis : biasanya digunakan untuk menangkap, mengolah, dan menganalisis gambar yang ditangkap dari pasien untuk mendiagnosis penyakit. Umumnya data gambar diambil dari mikroskop, x-ray, angiography, ultrasonik, dan tomography. Aplikasi di bidang medis juga termasuk peningkatan kualitas gambar yang

- ✓ diinterpretasikan oleh manusia, misalnya gambar ultrasonik atau gambar X-ray, untuk mengurangi pengaruh noise.²³

²³ Dana H. Ballard and Christopher M. Brown, *Visi Komputer* (Prentice Hall, 1982).

BAB X

NATURAL LANGUAGE PROCESSING

Secara mendasar, komunikasi adalah salah satu hal paling penting yang dibutuhkan manusia sebagai makhluk sosial. Ada lebih dari trilyunan halaman berisi informasi pada Website, dimana kebanyakan diantaranya menggunakan bahasa natural. Isu yang sering muncul dalam pengolahan bahasa adalah ambiguitas, dan bahasa yang berantakan/tidak formal (tidak sesuai aturan bahasa).

Natural Language Processing (NLP) merupakan salah satu cabang ilmu AI yang berfokus pada pengolahan bahasa natural. Bahasa natural adalah bahasa yang secara umum digunakan oleh manusia dalam berkomunikasi satu sama lain. Bahasa yang diterima oleh komputer butuh untuk diproses dan dipahami terlebih dahulu supaya maksud dari user bisa dipahami dengan baik oleh komputer.

Ada berbagai terapan aplikasi dari NLP. Diantaranya adalah Chatbot (aplikasi yang membuat user bisa seolah-olah melakukan komunikasi dengan computer), Stemming atau Lemmatization (pemotongan kata dalam bahasa tertentu menjadi bentuk dasar pengenalan fungsi setiap kata dalam kalimat), Summarization (ringkasan dari bacaan), Translation Tools (menterjemahkan bahasa) dan aplikasi-aplikasi lain yang memungkinkan komputer mampu memahami instruksi bahasa yang diinputkan oleh user.²⁴

10.1. NLP Area

Pustejovsky dan Stubbs (2012) menjelaskan bahwa ada beberapa area utama penelitian pada field NLP, diantaranya:

- 1) Question Answering Systems (QAS). Kemampuan komputer untuk menjawab pertanyaan yang diberikan oleh user. Daripada memasukkan keyword ke dalam browser pencarian, dengan QAS, user bisa langsung bertanya dalam bahasa natural yang digunakannya, baik itu Inggris, Mandarin, ataupun Indonesia.

²⁴ D. Jurafsky and J.H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. 2nd Edition.* (New Jersey: Prentice Hall, 2008).

- 2) Summarization. Pembuatan ringkasan dari sekumpulan konten dokumen atau email. Dengan menggunakan aplikasi ini, user bisa dibantu untuk mengkonversikan dokumen teks yang besar ke dalam bentuk slide presentasi.
- 3) Machine Translation. Produk yang dihasilkan adalah aplikasi yang dapat memahami bahasa manusia dan menterjemahkannya ke dalam bahasa lain. Termasuk di dalamnya adalah Google Translate yang apabila dicermati semakin membaik dalam penterjemahan bahasa. Contoh lain lagi adalah BabelFish yang menterjemahkan bahasa pada real time.
- 4) Speech Recognition. Field ini merupakan cabang ilmu NLP yang cukup sulit. Proses pembangunan model untuk digunakan telpon/komputer dalam mengenali bahasa yang diucapkan sudah banyak dikerjakan. Bahasa yang sering digunakan adalah berupa pertanyaan dan perintah.
- 5) Document classification. Sedangkan aplikasi ini adalah merupakan area penelitian NLP Yang paling sukses. Pekerjaan yang dilakukan aplikasi ini adalah menentukan dimana tempat terbaik dokumen yang baru diinputkan ke dalam sistem. Hal ini sangat berguna pada aplikasi spam filtering, news article classification, dan movie review.

10.2. Terminologi NLP

Perkembangan NLP menghasilkan kemungkinan dari interface bahasa natural menjadi knowledge base dan penterjemahan bahasa natural. Poole dan Mackworth (2010) menjelaskan bahwa ada 3 (tiga) aspek utama pada teori pemahaman mengenai natural language:

- 1) Syntax: menjelaskan bentuk dari bahasa. Syntax biasa dispesifikasikan oleh sebuah grammar. Natural language jauh lebih daripada formal language yang digunakan untuk logika kecerdasan buatan dan program komputer
- 2) Semantics: menjelaskan arti dari kalimat dalam satu bahasa. Meskipun teori semantics secara umum sudah ada, ketika membangun sistem natural language understanding untuk aplikasi tertentu, akan digunakan representasi yang paling sederhana.
- 3) Pragmatics: menjelaskan bagaimana pernyataan yang ada berhubungan dengan dunia. Untuk memahami bahasa, agen harus mempertimbangan lebih dari hanya sekedar kalimat. Agen harus melihat lebih ke dalam konteks kalimat, keadaan dunia, tujuan dari speaker dan listener, konvensi khusus, dan sejenisnya.

Contoh kalimat di bawah ini akan membantu untuk memahami perbedaan diantara ketiga aspek tersebut di atas. Kalimat-kalimat ini adalah kalimat yang mungkin muncul pada bagian awal dari sebuah buku Artificial Intelligence (AI):

1. This book is about Artificial Intelligence
2. The green frogs sleep soundly
3. Colorless green ideas sleep furiously
4. Furiously sleep ideas green colorless

Kalimat pertama akan tepat jika diletakkan pada awal sebuah buku, karena tepat secara sintaks, semantik, dan pragmatik. Kalimat kedua tepat secara sintaks dan semantic, namun kalimat tersebut akan menjadi aneh apabila diletakkan pada awal sebuah buku AI, sehingga kalimat ini tidak tepat secara pragmatik. Kalimat ketiga tepat secara sintaks, tetapi tidak secara semantik. Sedangkan pada kalimat keempat, tidak tepat secara sintaks, semantik, dan pragmatik.

Selain daripada ketiga istilah tersebut ada beberapa istilah yang terkait dengan NLP, yaitu:

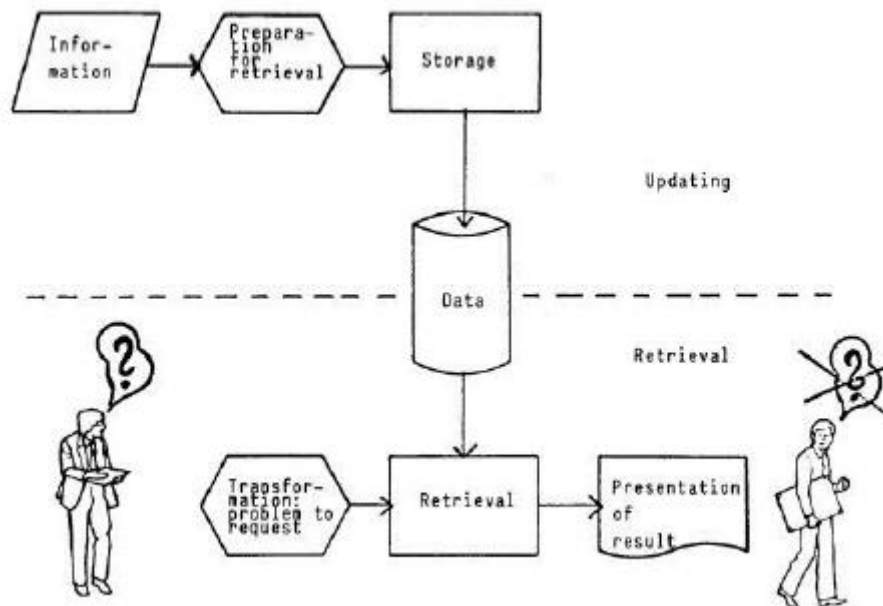
- Morfologi. Adalah pengetahuan tentang kata dan bentuknya sehingga bisa dibedakan antara yang satu dengan yang lainnya. Bisa juga didefinisikan asal usul sebuah kata itu bisa terjadi. Contoh : membangunkan → bangun (kata dasar), mem- (prefix), -kan (suffix)
- Fonetik. Adalah segala hal yang berhubungan dengan suara yang menghasilkan kata yang dapat dikenali. Fonetik digunakan dalam pengembangan NLP khususnya bidang speech based system

10.3. Information Retrieval

Information Retrieval (IR) adalah pekerjaan untuk menemukan dokumen yang relevan dengan kebutuhan informasi yang dibutuhkan oleh user. Contoh sistem IR yang paling populer adalah search engine pada World Wide Web. Seorang pengguna Web bisa menginputkan query berupa kata apapun ke dalam sebuah search engine dan melihat hasil dari pencarian yang relevan. Karakteristik dari sebuah sistem IR (Russel & Norvig, 2010) diantaranya adalah:

- ✓ A corpus of documents. Setiap sistem harus memutuskan dokumen yang ada akan diperlakukan sebagai apa. Bisa sebagai sebuah paragraf, halaman, atau teks multipage.

- ✓ Queries posed in a query language. Sebuah query menjelaskan tentang apa yang user ingin peroleh. Query language dapat berupa list dari kata-kata, atau bisa juga menspesifikasikan sebuah frase dari kata-kata yang harus berdekatan
- ✓ A result set. Ini adalah bagian dari dokumen yang dinilai oleh sistem IR sebagai yang relevan dengan query.
- ✓ A presentation of the result set. Maksud dari bagian ini adalah tampilan list judul dokumen yang sudah di ranking.



10.4. Morphological Analysis

Proses dimana setiap kata yang berdiri sendiri (individual words) dianalisis kembali ke komponen pembentuk mereka dan token nonword seperti tanda baca dsb dipisahkan dari kata tersebut.

Contohnya apabila terdapat kalimat:

“I want to print Bill’s .init file”

Jika morphological analysis diterapkan ke dalam kalimat di atas, maka:

- Pisahkan kata “Bill’s” ke bentuk proper noun “Bill” dan possessive suffix “’s”
- Kenali sequence “.init” sebagai sebuah extension file yang berfungsi sebagai adjective dalam kalimat.

Syntactic analysis harus menggunakan hasil dari morphological analysis untuk membangun sebuah deskripsi yang terstruktur dari kalimat. Hasil akhir dari proses ini adalah yang sering disebut sebagai parsing. Parsing adalah mengkonversikan daftar kata yang berbentuk kalimat ke dalam bentuk struktur yang mendefinisikan unit yang diwakili oleh daftar tadi.

Hampir semua sistem yang digunakan untuk syntactic processing memiliki dua komponen utama, yaitu:

- Representasi yang deklaratif, yang disebut juga sebagai Grammar, dari fakta sintaktis mengenai bahasa yang digunakan
- Procedure, yang disebut juga sebagai Parser, yang membandingkan grammar dengan kalimat yang diinputkan untuk menghasilkan struktur kalimat yang telah di parsing

Cara yang paling umum digunakan untuk merepresentasikan grammar adalah dengan sekumpulan production rule. Rule yang paling pertama bisa diterjemahkan sebagai “Sebuah Sentence terdiri dari sebuah Noun Phrase, diikuti oleh Verb Phrase”, garis vertical adalah OR, sedangkan ϵ mewakili string kosong.

Proses parsing menggunakan aturan-aturan yang ada pada Grammar, kemudian membandingkannya dengan kalimat yang diinputkan. Struktur paling sederhana dalam melakukan parsing adalah Parse Tree, yang secara sederhana menyimpan rule dan bagaimana mereka dicocokkan satu sama lain. Setiap node pada Parse Tree berhubungan dengan kata yang dimasukkan atau pada nonterminal pada Grammar yang ada. Setiap level pada Parse Tree berkorespondensi dengan penerapan dari satu rule pada Grammar.

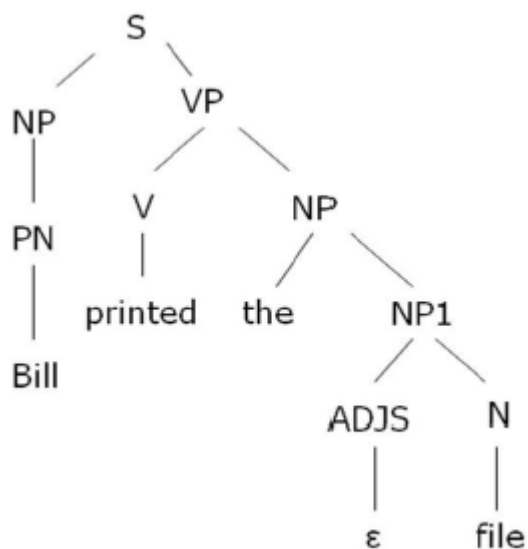
Contoh:

Terdapat Grammar sebagai berikut:

- $S \rightarrow NP VP$
- $NP \rightarrow the NP1$
- $NP \rightarrow PRO$
- $NP \rightarrow PN$
- $NP \rightarrow NP1$
- $NP1 \rightarrow ADJS N$

- $ADJS \rightarrow \epsilon \mid ADJ ADJS$
- $VP \rightarrow V$
- $P \rightarrow V NP$
- $N \rightarrow \text{file} \mid \text{printer}$
- $PN \rightarrow \text{Bill}$
- $PRO \rightarrow I$
- $ADJ \rightarrow \text{short} \mid \text{long} \mid \text{fast}$
- $V \rightarrow \text{printed} \mid \text{created} \mid \text{want}$

Maka, apabila terdapat kalimat "Bill printed the file", representasi Parse Tree nya akan menjadi:



Pembangunan Parse Tree ini didasarkan pada Grammar yang digunakan. Apabila Grammar yang digunakan berbeda, maka Parse Tree yang dibangun harus tetap berdasarkan pada Grammar yang berlaku.

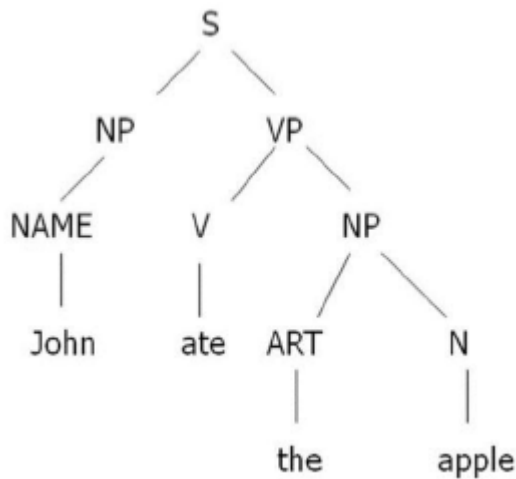
Contoh:

Terdapat Grammar sebagai berikut:

- $S \rightarrow NP VP$
- $VP \rightarrow V NP$
- $NP \rightarrow \text{NAME}$
- $NP \rightarrow \text{ART N}$
- $\text{NAME} \rightarrow \text{John}$
- $V \rightarrow \text{ate}$

- ART → the
- N → apple

Maka Parse Tree untuk kalimat “John ate the apple” akan menjadi:



10.5. Stemming & Lemmatization

Stemming merupakan sebuah proses yang bertujuan untuk mereduksi jumlah variasi dalam representasi dari sebuah kata (Kowalski, 2011). Resiko dari proses stemming adalah hilangnya informasi dari kata yang di-stem. Hal ini menghasilkan menurunnya akurasi atau presisi. Sedangkan untuk keuntungannya adalah, proses stemming bisa meningkatkan kemampuan untuk melakukan recall. Tujuan dari stemming sebenarnya adalah untuk meningkatkan performace dan mengurangi penggunaan resource dari sistem dengan mengurangi jumlah unique word yang harus diakomodasikan oleh sistem. Jadi, secara umum, algoritma stemming mengerjakan transformasi dari sebuah kata menjadi sebuah standar representasi morfologi (yang dikenal sebagai stem).

Contoh:

“comput” adalah stem dari “computable, computability, computation, computational, computed, computing, compute, computerize”

Ingason dkk. (2008) mengemukakan bahwa lemmatization adalah sebuah proses untuk menemukan bentuk dasar dari sebuah kata. Nirenburg (2009) mendukung teori ini dengan kalimatnya yang menjelaskan bahwa lemmatization adalah proses yang bertujuan untuk melakukan normalisasi pada teks/kata dengan berdasarkan pada

bentuk dasar yang merupakan bentuk lemma-nya. Normalisasi disini adalah dalam artian mengidentifikasi dan menghapus prefiks serta suffiks dari sebuah kata. Lemma adalah bentuk dasar dari sebuah kata yang memiliki arti tertentu berdasar pada kamus.

Contoh:

- Input: "The boy's cars are different colors"
- Transformation: am, is, are à be
- Transformation: car, cars, car's, cars' à car
- Hasil: "The boy car be differ color"

Algoritma Stemming dan Lemmatization berbeda untuk bahasa yang satu dengan bahasa yang lain.

10.6. Contoh Aplikasi NLP

Penelitian yang dikerjakan oleh Suhartono, Christiandy, dan Rolando (2013) adalah merancang sebuah algoritma lemmatization untuk Bahasa Indonesia. Algoritma ini dibuat untuk menambahkan fungsionalitas pada algoritma Stemming yang sudah pernah dikerjakan sebelumnya yaitu Enhanced Confix-Stripping Stemmer (ECS) yang dikerjakan pada tahun 2009. ECS sendiri merupakan pengembangan dari algoritma Confix-Stripping Stemmer yang dibuat pada tahun 2007. Pengembangan yang dikerjakan terdiri dari beberapa rule tambahan dan modifikasi dari rule sebelumnya. Langkah untuk melakukan suffix backtracking juga ditambahkan. Hal ini untuk menambah akurasi.

Secara mendasar, algoritma lemmatization ini tidak bertujuan untuk mengembangkan dari metode ECS, karena tujuannya berbeda. Algoritma lemmatization bertujuan untuk memodifikasi ECS, supaya lebih tepat dengan konsep lemmatization. Namun demikian, masih ada beberapa kemiripan pada proses yang ada pada ECS. Ada beberapa kasus yang mana ECS belum berhasil untuk digunakan, namun bisa diselesaikan pada algoritma lemmatization ini.

Pengujian validitas pada algoritma ini adalah dengan menggunakan beberapa artikel yang ada di Kompas, dan diperoleh hasil sebagai berikut:

T = Total data count

V = Valid test data count

S = Successful lemmatization

E = Error / Kegagalan

P = Precision

Aplikasi NLP yang lainnya adalah seperti penerjemah bahasa, chatting dengan komputer, meringkas satu bacaan yang panjang, pengecekan grammar dan lain sebagainya.²⁵

²⁵ Ibid.

BAB XI

JARINGAN SYARAF TIRUAN

Jaringan saraf tiruan (JST) (Bahasa Inggris: artificial neural network (ANN), atau juga disebut simulated neural network (SNN), atau umumnya hanya disebut neural network (NN)), adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan sistem saraf manusia. JST merupakan sistem adaptif yang dapat mengubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan tersebut. Oleh karena sifatnya yang adaptif, JST juga sering disebut dengan jaringan adaptif.

Secara sederhana, JST adalah sebuah alat pemodelan data statistik non-linier. JST dapat digunakan untuk memodelkan hubungan yang kompleks antara input dan output untuk menemukan pola-pola pada data. Menurut suatu teorema yang disebut "teorema penaksiran universal", JST dengan minimal sebuah lapis tersembunyi dengan fungsi aktivasi non-linear dapat memodelkan seluruh fungsi terukur Boreal apapun dari suatu dimensi ke dimensi lainnya.²⁶

11.1. Sejarah

Saat ini bidang kecerdasan buatan dalam usahanya menirukan intelegensi manusia, belum mengadakan pendekatan dalam bentuk fisiknya melainkan dari sisi yang lain. Pertama-tama diadakan studi mengenai teori dasar mekanisme proses terjadinya intelegensi. Bidang ini disebut Cognitive Science. Dari teori dasar ini dibuatlah suatu model untuk disimulasikan pada komputer, dan dalam perkembangannya yang lebih lanjut dikenal berbagai sistem kecerdasan buatan yang salah satunya adalah jaringan saraf tiruan. Dibandingkan dengan bidang ilmu yang lain, jaringan saraf tiruan relatif masih baru. Sejumlah literatur menganggap bahwa konsep jaringan saraf tiruan bermula pada makalah Waffen McCulloch dan Walter Pitts pada tahun 1943. Dalam makalah tersebut mereka mencoba untuk memformulasikan model matematis sel-sel otak. Metode yang dikembangkan berdasarkan sistem saraf biologi ini, merupakan suatu langkah maju dalam industri komputer.

²⁶ C.M. Bishop, *Neural Networks for Pattern Recognition* (Oxford: Oxford University Press, 1995).

11.2. Model

Model pada JST pada dasarnya merupakan fungsi model matematika yang mendefinisikan fungsi. Istilah "jaringan" pada JST merujuk pada interkoneksi dari beberapa neuron yang diletakkan pada lapisan yang berbeda. Secara umum, lapisan pada JST dibagi menjadi tiga bagian:

- Lapis masukan (input layer) terdiri dari neuron yang menerima data masukan dari variabel X . Semua neuron pada lapis ini dapat terhubung ke neuron pada lapisan tersembunyi atau langsung ke lapisan luaran jika jaringan tidak menggunakan lapisan tersembunyi.
- Lapisan tersembunyi (hidden layer) terdiri dari neuron yang menerima data dari lapisan masukan.
- Lapisan luaran (output layer) terdiri dari neuron yang menerima data dari lapisan tersembunyi atau langsung dari lapisan masukan yang nilai luarannya melambangkan hasil kalkulasi dari X menjadi nilai Y .

Secara matematis, neuron merupakan sebuah fungsi yang menerima masukan dari lapisan sebelumnya (lapisan ke-). Fungsi ini pada umumnya mengolah sebuah vektor untuk kemudian diubah ke nilai skalar melalui komposisi nonlinear weighted sum, dimana , merupakan fungsi khusus yang sering disebut dengan fungsi aktivasi dan merupakan beban atau weight.²⁷

²⁷ Ibid.

DAFTAR PUSTAKA

- Ballard, Dana H., and Christopher M. Brown. *Visi Komputer*. Prentice Hall, 1982.
- Bishop, C.M. *Neural Networks for Pattern Recognition*. Oxford: Oxford University Press, 1995.
- Davis, Lawrence. *Handbook Of Genetic Algorithms*. New York: Van Nostrand Reinhold, 1991.
- Giarratano, J., and G Riley. *Expert Systems; Principles and Programming*. Boston: PWS Publishing Company, 2005.
- Hermawan, Arief. *Jaringan Syaraf Tiruan (Teori Dan Aplikasi)*. Penerbit Andi, 2006.
- Hidayanto, Akhmad. *Knowledge Management*. Departemen Teknik Industri Universitas Indonesia, 2006.
- Jurafsky, D., and J.H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. 2nd Edition*. New Jersey: Prentice Hall, 2008.
- Kristanto, Andri. *Kecerdasan Buatan*. Yogyakarta: Graha Ilmu, 2004.
- Kusumadewi, Sri. *Artificial Intelligence (Teknik Dan Aplikasinya)*. Yogyakarta: Graha Ilmu, 2003.
- Saelan, Athia. "Logika Fuzzy." *Struktur Diskrit* 1, no. 13508029 (2009): 1–5.
- Sri, Kusuma Dewi, and Hari Purnomo. *Aplikasi Logika Fuzzy Untuk Mendukung Keputusan*. Yogyakarta: Graha Ilmu, 2004.
- Turban, E. *Decision Support and Expert Systems; Management Support Systems*. New Jersey: Prentice Hall, 1995.