

V. 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.

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.
- 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 pembandingan 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 aseosri Pcbboard, kabel, disk drive, periperak, perangkat lunak, dan lainnya bisa dirakit ke dalam konfigurasi yang sangat rapih. Untuk mengidentifikasi hal-hal tersebut diperlukan waktu berhari-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 istem 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

- = 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.

RAMALAN CUACA

Dengan diberi input tentang situasi cuaca yang sedang berlangsung, baik lokal maupun ditempat lain, maka sistem pakar bisa menyajikan ramalan yang akurat tentang cuaca yang akan terjadi dalam suatu periode tertentu.

CONTOH LAIN SISTEM PAKAR

Sistem pakar :

- Digunakan untuk konsultasi
- Sistem pakar selalu tersedia di organisasi, sedang pakar belum tentu selalu berada di tempat. Misal suatu keputusan harus diambil oleh manajer yang pakar dalam suatu bidang, karena manajer ini pergi dan tidak berada di kantor, maka keputusan yang harus diambil tertunda.
- Sistem pakar dapat menyimpan dan mengingat pengetahuan yang sangat tidak terbatas dan tidak kenal lelah. Oleh karena itu pekerjaan dokter akan sangat terbantu sekali dengan SP yang diisi dengan sejumlah pengetahuan (misal semua jenis obat dan efeknya) yang pakarnya sendiri belum tentu dapat mengingatnya.
- Pak A nasabah bank X. Pak A akan meminjam uang untuk membeli rumah. Di bank X pak A menanyakan ke bagian informasi dan disarankan untuk menuju ke lantai 3 di kantor manajer installment loan. Di kantor ini, pak A mengutarakan maksudnya untuk meminjam uang dan akan

dibayar angsuran tiap bulannya bervariasi besarnya tergantung dari penghasilannya. Manajer installment loan menolak karena pinjaman di bagian ini harus dibayar angsuran yang nilainya sudah tetap ditentukan di muka. Manajer ini mengatakan bahwa pak A salah tempat menemui dia dan menyarankan ke lantai 5 di kantor manajer mortgage loan. Sesampainya disana, pak A mengutarakan kembali maksudnya untuk meminjam uang dengan membayar secara angsuran. Manajer ini setuju tetapi pak A harus meninggalkan sertifikat tanahnya sebagai agunan. Sebaliknya pak A tidak setuju karena dia akan menggunakan sertifikat tanah ini untuk keperluan yang lain. Manajer mortgage loan menyarankan pak A untuk menemui kepala cabang di lantai 1. Pak A turun ke lantai 1 bukannya menemui kepala cabang tetapi bank lain yaitu Y di seberang jalan.

Di bank Y pak A ditemui oleh seorang pegawai bank yang mempersilahkan dia duduk. Setelah pak A mengutarakan maksudnya, tak lama kemudian setelah pegawai menggunakan komputer dihadapannya, pegawai tersebut menyetujui pinjaman pak A.

Beda bank X dan bank Y.

Untuk bank X kepandaian atau pengetahuan (knowledge) dimiliki di masing-masing manajer. Dan jika pak A menemui manajer yang pengetahuannya lain yang tidak sesuai dengan permasalahannya, pak A salah alamat dan harus menemui manajer lain yang sesuai.

Untuk bank Y, pengetahuan dari manajer-manajer bersangkutan dimasukkan dan berada di dalam sistem komputer dalam bentuk basis data pengetahuan (knowledge base).

- Sistem pakar yang digunakan di dinas sosial negara bagian California, Amerika Serikat. Sebelum SP digunakan, pemberian tunjangan sosial kurang efektif karena beragamnya macam tunjangan yang diberikan dan banyaknya aturan yang ada untuk mendapatkan tunjangan sosial. Lebih dari 3000 aturan dibukukan untuk tunjangan sosial ini. Pada waktu seseorang melamar untuk meminta tunjangan sosial, orang ini akan dilayani dengan pekerja sosial dan pekerja sosial harus mengetahui aturan-aturan yang ada. Jika ada kasus khusus dan pekerja sosial tidak memahaminya tetapi memutuskan hasilnya, maka hasil keputusan dapat tidak efektif. Menyadari kelemahan-kelemahan ini maka dinas sosial kemudian menerapkan sistem pakar yang berisi dengan knowledge base berupa ribuan aturan-aturan ini.

Bedanya : sebelum ada SP, yang pakar adalah pekerja sosialnya dan jika pekerja sosial kurang pakar maka dapat mengakibatkan kesalahan keputusan. Setelah ada SP, pekerja sosial tidak harus pakar karena yang pakar adalah sistemnya karena sistemnya berisi dengan semua aturan, sehingga mengurangi kesalahan pengambilan keputusan.

MANFAAT SISTEM PAKAR :

1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli
2. Bisa melakukan proses secara berulang secara otomatis
3. Menyimpan pengetahuan dan keahlian para pakar
4. Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka)
5. Mampu beroperasi dalam lingkungan yang berbahaya
6. 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.
7. Tidak memerlukan biaya saat tidak digunakan, sedangkan pada pakar manusia memerlukan biaya sehari-hari.
8. Dapat digandakan (diperbanyak) sesuai kebutuhan dengan waktu yang minimal dan sedikit biaya
9. Dapat memecahkan masalah lebih cepat daripada kemampuan manusia dengan catatan menggunakan data yang sama.
10. Menghemat waktu dalam pengambilan keputusan
11. Meningkatkan kualitas dan produktivitas karena dapat memberi nasehat yang konsisten dan mengurangi kesalahan
12. Meningkatkan kapabilitas sistem terkomputerisasi yang lain. Integrasi Sistem Pakar dengan sistem komputer lain membuat lebih efektif, dan bisa mencakup lebih banyak aplikasi .
13. Mampu menyediakan pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman. Fasilitas penjelas dapat berfungsi sebagai guru.

KELEMAHAN SISTEM PAKAR

1. Biaya yang diperlukan untuk membuat, memelihara, dan mengembangkannya sangat mahal
2. 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.
3. 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.
4. Pendekatan oleh setiap pakar untuk suatu situasi atau problem bisa berbeda-beda, meskipun sama-sama benar.
5. Transfer pengetahuan dapat bersifat subjektif dan bias
6. Kurangnya rasa percaya pengguna dapat menghalangi pemakaian sistem pakar.

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.

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

ELEMEN MANUSIA YANG TERKAIT DALAM PENGGUNAAN DAN 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

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, peralaman 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.

BENTUK / TIPE SISTEM PAKAR

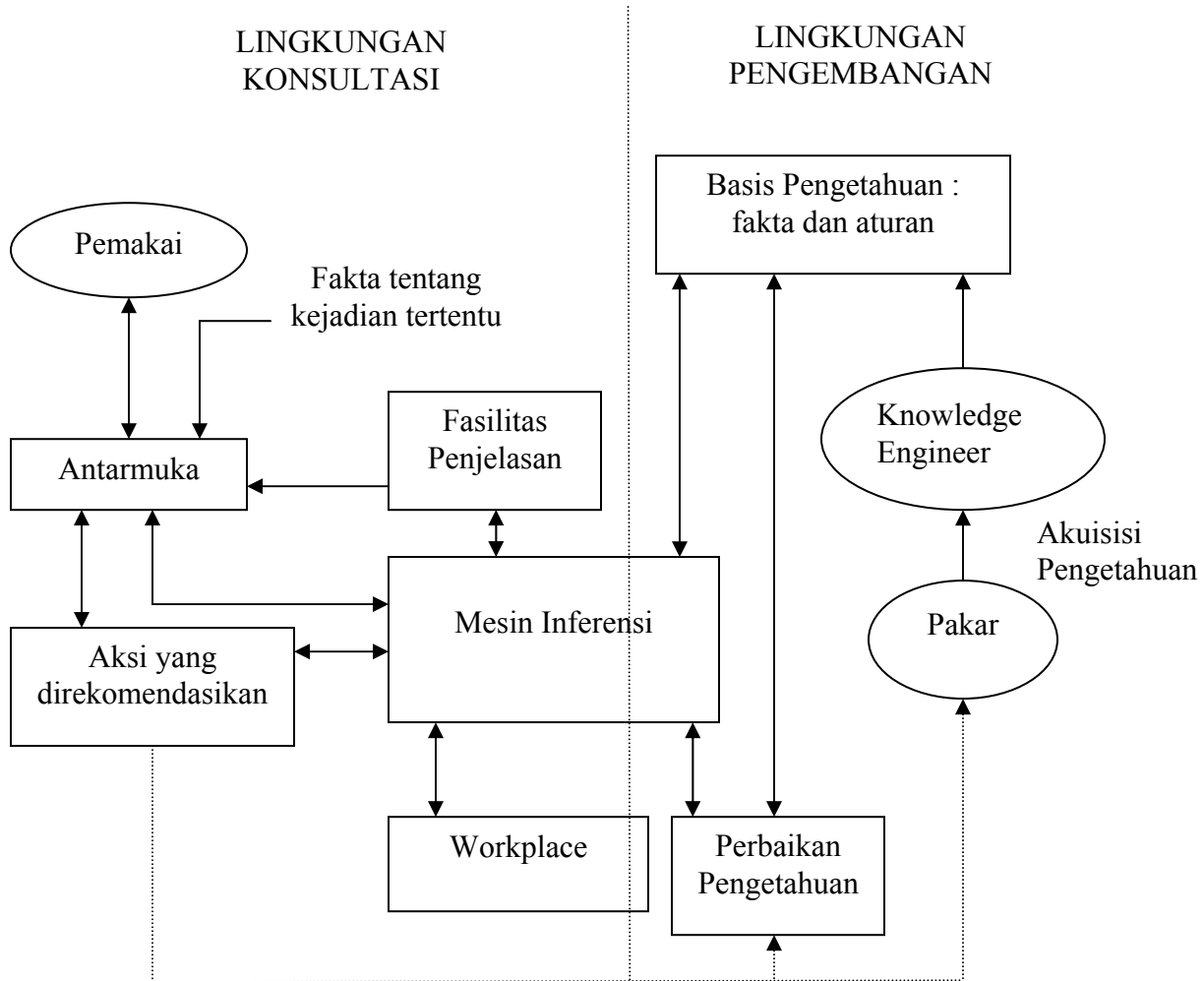
1. Mandiri : sistem pakar yang murni berdiri sendiri, tidak digabung dengan software lain, bisa dijalankan pada komputer pribadi, mainframe.
2. Terkait/Tergabung : dalam bentuk ini sistem pakar hanya merupakan bagian dari program yang lebih besar. Program tersebut biasanya menggunakan teknik algoritma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

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

Arsitektur sistem pakar :



Komponen-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.

2. 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.

3. 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.

4. 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

5. 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 ?

6. 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

BASIS PENGETAHUAN (KNOWLEDGE BASE)

Basis pengetahuan berisi pengetahuan-pengetahuan dalam penyelesaian masalah. Ada 2 bentuk pendekatan basis pengetahuan :

a. 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...

b. 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.

MESIN INFERENSI (INFERENCE ENGINE)

Ada 2 cara penalaran yang dapat dikerjakan dalam melakukan inferensi :

a. 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.

b. 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.

Contoh :

R1 : IF suku bunga turun THEN harga obligasi naik

R2 : IF suku bunga naik THEN harga obligasi turun

R3 : IF suku bunga tidak berubah THEN harga obligasi tidak berubah

R4 : IF dolar naik THEN suku bunga turun

R5 : IF dolar turun THEN suku bunga naik

R6 : IF harga obligasi turun THEN beli obligasi

Apabila diketahui bahwa dolar turun, maka untuk memutuskan apakah akan membeli obligasi atau tidak dapat ditunjukkan sebagai berikut :

Forward Chaining

Dari fakta dolar turun, berdasarkan Rule 5, diperoleh konklusi suku bunga naik. Dari Rule 2 suku bunga naik menyebabkan harga obligasi turun. Dengan Rule 6, jika harga obligasi turun, maka kesimpulan yang diambil adalah membeli obligasi.

Backward Chaining

Dari solusi yaitu membeli obligasi, dengan menggunakan Rule 6 diperoleh anteseden harga obligasi turun. Dari Rule 2 dibuktikan harga obligasi turun bernilai benar jika suku bunga naik bernilai benar . Dari Rule 5 suku bunga naik bernilai memang bernilai benar karena diketahui fakta dolar turun.

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. merekayasa pengetahuan
7. merancang sistem
8. melengkapi pengembangan
9. menguji dan mencari kesalahan sistem
10. memelihara sistem

JENIS KERUSAKAN

A1	=	MONITOR RUSAK
A2	=	MEMORI RUSAK
A3	=	HDD RUSAK
A4	=	VGA RUSAK
A5	=	SOUND CARD RUSAK
A6	=	OS BERMASALAH
A7	=	APLIKASI RUSAK
A8	=	PSU RUSAK
A9	=	PROSESOR RUSAK
A10	=	MEMORY KURANG (PERLU UPGRADE MEMORY)
A11	=	MEMORY VGA KURANG (PERLU UPGRADE VGA)
A12	=	CLOCK PROSOR KURANG TINGGI (PERLU UPGRADE PROSESOR)
A13	=	KABEL IDE RUSAK
A14	=	KURANG DAYA PADA PSU (PERLU UPGRADE PSU)
A15	=	PERANGKAT USB RUSAK
A16	=	KEYBOARD RUSAK
A17	=	MOUSE RUSAK

RULE	IF	THEN
1	B1,B2	A1
2	B3,B4,B5,B11,B12	A2
3	B6,B7,B8,B10,B21,B22	A3
4	B1,B3,B5,B9,B10,B12,B13	A4
5	B10,B13,B14	A5
6	B11,B11,B15	A6
7	B7,B12	A7
8	B16,B17	A8
9	B1,B3,B4,B5	A9
10	B18,B19	A10
11	B9,B20	A11
12	B19	A12
13	B21	A13
14	B5,B23	A14
15	B10	A15
16	B10,B24	A16
17	B10,B25	A17

VI. KETIDAKPASTIAN

Dalam kenyataan sehari-hari banyak masalah didunia ini tidak dapat dimodelkan secara lengkap dan konsisten. Suatu penalaran dimana adanya penambahan fakta baru mengakibatkan ketidakkonsistenan, 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 digunakan penalaran statistik.

PROBABILITAS & TEOREMA BAYES

PROBABILITAS

Probabilitas menunjukkan kemungkinan sesuatu akan terjadi atau tidak.

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

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

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

TEOREMA BAYES

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

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 :

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

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

Maka :

- probabilitas Asih terkena cacar karena ada bintik-bintik di wajahnya :

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

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

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

- probabilitas Asih 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 Asih jerawat karena ada bintik-bintik di wajahnya :

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

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

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

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

dengan:

e = evidencelama

E = evidenceatau observasibaru

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

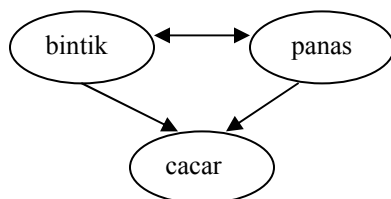
$p(H | E)$ = probabilitas hipotesis H benar jika diberikanevidence E

$p(e | E, H)$ = 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.



Asih ada bintik-bintik di wajahnya. Dokter menduga bahwa Asih terkena cacar dengan probabilitas terkena cacar bila ada bintik-bintik di wajah $\rightarrow p(\text{cacar} | \text{bintik}) = 0.8$

Ada observasi bahwa orang terkena cacar pasti mengalami panas badan. Jika diketahui probabilitas orang terkena cacar bila panas badan $\rightarrow p(\text{cacar} | \text{panas}) = 0.5$

Keterkaitan antara adanya bintik-bintik di wajah dan panas badan bila seseorang terkena cacar $\rightarrow p(\text{bintik} | \text{panas, cacar}) = 0.4$

Keterkaitan antara adanya bintik-bintik di wajah dan panas badan $\rightarrow p(\text{bintik} | \text{panas}) = 0.6$
 Maka :

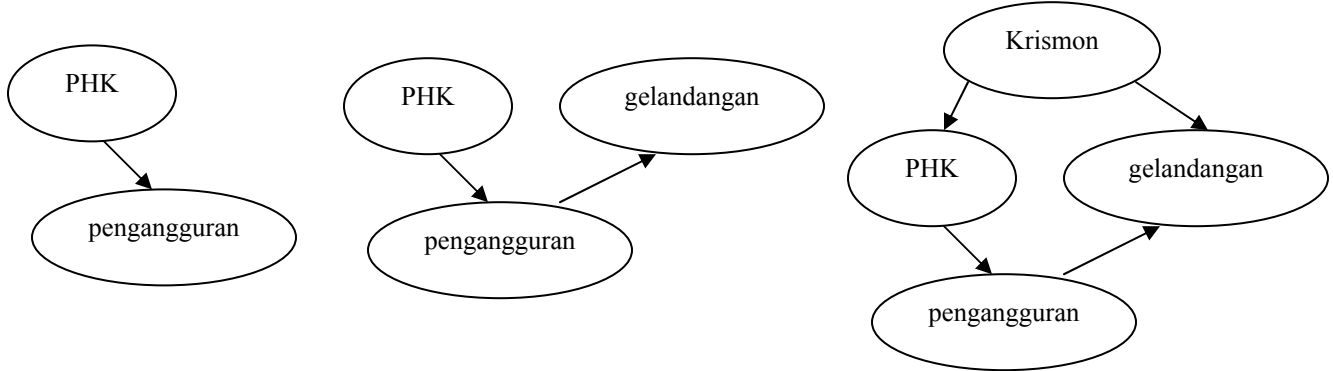
$$p(H | E, e) = p(H | E) * \frac{p(e | E, H)}{p(e | E)}$$

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

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

Pengembangan lebih jauh dari Teorema Bayes adalah Jaringan Bayes.

Contoh : hubungan antara krismon, PHK, pengangguran, gelandangan dalam suatu jaringan.



Munculnya pengangguran disebabkan PHK

Munculnya pengangguran dapat digunakan sebagai evidence untuk membuktikan adanya gelandangan

Probabilitas terjadinya PHK jika terjadi krismon, probabilitas munculnya gelandangan jika terjadi krismon

Probabilitas untuk jaringan bayes

Atribut	Prob	Keterangan
$p(\text{pengangguran} \text{PHK, gelandangan})$	0.95	Keterkaitan antara pengangguran & PHK, jika muncul gelandangan
$p(\text{pengangguran} \text{PHK, } \sim \text{gelandangan})$	0.20	Keterkaitan antara pengangguran & PHK, jika tidak ada gelandangan
$p(\text{pengangguran} \sim \text{PHK, gelandangan})$	0.75	Keterkaitan antara pengangguran & tidak ada PHK, jika muncul gelandangan
$p(\text{pengangguran} \sim \text{PHK, } \sim \text{gelandangan})$	0.40	Keterkaitan antara pengangguran & tidak ada PHK, jika tidak ada gelandangan
$p(\text{PHK} \text{krismon})$	0,50	Probabilitas orang diPHK jika terjadi krismon
$p(\text{PHK} \sim \text{krismon})$	0.10	Probabilitas orang diPHK jika tidak terjadi krismon
$p(\text{pengangguran} \text{krismon})$	0.90	Probabilitas muncul pengangguran jika terjadi krismon
$p(\text{pengangguran} \sim \text{krismon})$	0.30	Probabilitas muncul pengangguran jika tidak terjadi krismon
$p(\text{krismon})$	0.80	

FAKTOR KEPASTIAN (CERTAINTY FACTOR)

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

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

CF[h,e] = faktor kepastian

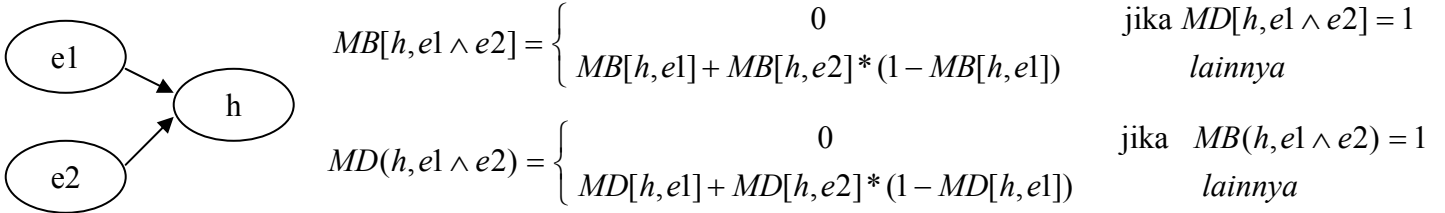
MB[h,e] = ukuran kepercayaan/tingkat keyakinan terhadap hipotesis h, jika diberikan/dipengaruhi evidence e (antara 0 dan 1)

MD[h,e] = ukuran ketidakpercayaan/tingkat ketidakyakinan 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 :



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 :

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

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

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

- Asih menderita bintik-bintik di wajahnya. Dokter memperkirakan Asih terkena cacar dengan kepercayaan MB[cacar,bintik]=0,80 dan MD[cacar,bintik]=0,01 maka :

$$CF[cacar,bintik] = 0,80 - 0,01 = 0,79$$

Jika ada observasi baru bahwa Asih juga panas badan dengan kepercayaan, MB[cacar,panas]=0,7 dan MD[cacar,panas]=0,08 maka :

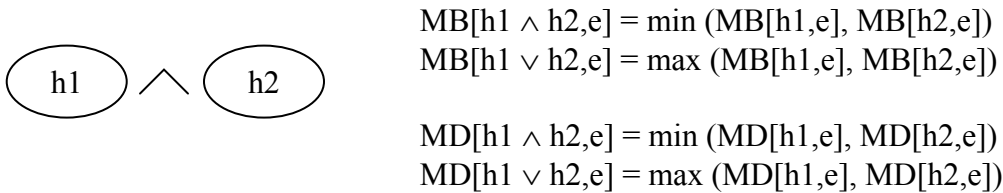
$$MB[cacar,bintik \wedge panas] = 0,8 + 0,7 * (1 - 0,8) = 0,94$$

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

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

2. CF dihitung dari kombinasi beberapa hipotesis

Jika h1 dan h2 adalah hipotesis maka :



Contoh :

- 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 ∧ 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 ∨ h2,e] diperoleh dari

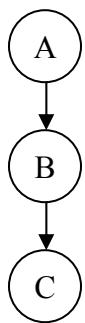
- $MB[h1 \vee h2, e] = \max(0,5 ; 0,8) = 0,8$
 $MD[h1 \vee h2, e] = \max(0,2 ; 0,1) = 0,2$
 $CF[h1 \vee h2, e] = 0,8 - 0,2 = 0,6$
- Asih menderita bintik-bintik di wajahnya. Dokter memperkirakan Asih terkena cacar dengan kepercayaan $MB[\text{cacar}, \text{bintik}] = 0,80$ dan $MD[\text{cacar}, \text{bintik}] = 0,01$ maka
 $CF[\text{cacar}, \text{bintik}] = 0,80 - 0,01 = 0,79$
 Jika observasi tersebut juga memberikan kepercayaan bahwa Asih mungkin juga terkena alergi dengan kepercayaan $MB[\text{alergi}, \text{bintik}] = 0,4$ dan $MD[\text{alergi}, \text{bintik}] = 0,3$ maka
 $CF[\text{alergi}, \text{bintik}] = 0,4 - 0,3 = 0,1$
 Untuk mencari $CF[\text{cacar} \wedge \text{alergi}, \text{bintik}]$ diperoleh dari
 $MB[\text{cacar} \wedge \text{alergi}, \text{bintik}] = \min(0,8 ; 0,4) = 0,4$
 $MD[\text{cacar} \wedge \text{alergi}, \text{bintik}] = \min(0,01 ; 0,3) = 0,01$
 $CF[\text{cacar} \wedge \text{alergi}, \text{bintik}] = 0,4 - 0,01 = 0,39$
 Untuk mencari $CF[\text{cacar} \vee \text{alergi}, \text{bintik}]$ diperoleh dari
 $MB[\text{cacar} \vee \text{alergi}, \text{bintik}] = \max(0,8 ; 0,4) = 0,8$
 $MD[\text{cacar} \vee \text{alergi}, \text{bintik}] = \max(0,01 ; 0,3) = 0,3$
 $CF[\text{cacar} \vee \text{alergi}, \text{bintik}] = 0,8 - 0,3 = 0,5$
 Kesimpulan : semula faktor kepercayaan bahwa Asih terkena cacar dari gejala munculnya bintik-bintik di wajahnya adalah 0,79. Demikian pula faktor kepercayaan bahwa Ani 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 :
 Asih menderita cacar dan alergi = 0,39
 Asih menderita cacar atau alergi = 0,5
 - Pertengahan tahun 2002, ada indikasi bahwa turunnya devisa Indonesia disebabkan oleh permasalahan TKI di Malaysia. Apabila diketahui $MB[\text{devisaturun}, \text{TKI}] = 0,8$ dan $MD[\text{devisaturun}, \text{TKI}] = 0,3$ maka $CF[\text{devisaturun}, \text{TKI}]$:
 $CF[\text{devisaturun}, \text{TKI}] = MB[\text{devisaturun}, \text{TKI}] - MD[\text{devisaturun}, \text{TKI}]$
 $0,8 - 0,3 = 0,5$
 Akhir September 2002 kemarau berkepanjangan mengakibatkan gagal panen yang cukup serius, berdampak pada turunnya ekspor Indonesia. Bila diketahui $MB[\text{devisaturun}, \text{eksporturun}] = 0,75$ dan $MD[\text{devisaturun}, \text{eksporturun}] = 0,1$, maka $CF[\text{devisaturun}, \text{eksporturun}]$ dan $CF[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}]$:
 $CF[\text{devisaturun}, \text{eksporturun}] = MB[\text{devisaturun}, \text{eksporturun}] - MD[\text{devisaturun}, \text{eksporturun}]$
 $= 0,75 - 0,1 = 0,65$
 $MB[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] =$
 $MB[\text{devisaturun}, \text{TKI}] + MB[\text{devisaturun}, \text{eksporturun}] * (1 - MB[\text{devisaturun}, \text{TKI}])$
 $= 0,8 + 0,75 * (1 - 0,8) = 0,95$
 $MD[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] =$
 $MD[\text{devisaturun}, \text{TKI}] + MD[\text{devisaturun}, \text{eksporturun}] * (1 - MD[\text{devisaturun}, \text{TKI}])$
 $= 0,3 + 0,1 * (1 - 0,3) = 0,37$
 $CF[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] =$
 $MB[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}] - MD[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun}]$
 $= 0,95 - 0,37 = 0,58$
 - 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 $MD[\text{devisaturun}, \text{bombali}] = 0,3$, maka $CF[\text{devisaturun}, \text{bombali}]$ dan $CF[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun} \wedge \text{bombali}]$:
 $CF[\text{devisaturun}, \text{bombali}] = MB[\text{devisaturun}, \text{bombali}] - MD[\text{devisaturun}, \text{bombali}]$
 $= 0,5 - 0,3 = 0,2$
 $MB[\text{devisaturun}, \text{TKI} \wedge \text{eksporturun} \wedge \text{bombali}] =$

$$\begin{aligned} & MB[\text{devisaturun,TKI} \wedge \text{eksporturun}] + MB[\text{devisaturun,bombali}] * (1 - MB[\text{devisaturun,} \\ & \text{TKI} \wedge \text{eksporturun}]) \\ & = 0,95 + 0,5 * (1 - 0,95) = 0,975 \end{aligned}$$

$$\begin{aligned} & MD[\text{devisaturun, TKI} \wedge \text{eksporturun} \wedge \text{bombali}] = \\ & MD[\text{devisaturun,TKI} \wedge \text{eksporturun}] + MD[\text{devisaturun,bombali}] * \\ & (1 - MD[\text{devisaturun,TKI} \wedge \text{eksporturun}]) \\ & = 0,37 + 0,3 * (1 - 0,37) = 0,559 \end{aligned}$$

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

3. Beberapa aturan saling bergandengan, ketidakpastian dari suatu aturan menjadi input untuk aturan yang lainnya



Maka :

$$MB[h,s] = MB'[h,s] * \max(0, CF[s,e])$$

$MB'[h,s]$ = ukuran kepercayaan h berdasarkan keyakinan penuh terhadap validitas s

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

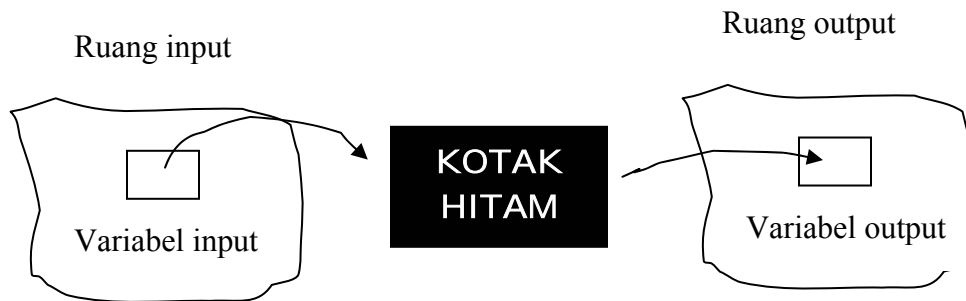
$$MB[\text{gelandangan, pengangguran}] = 0,7$$

Maka =

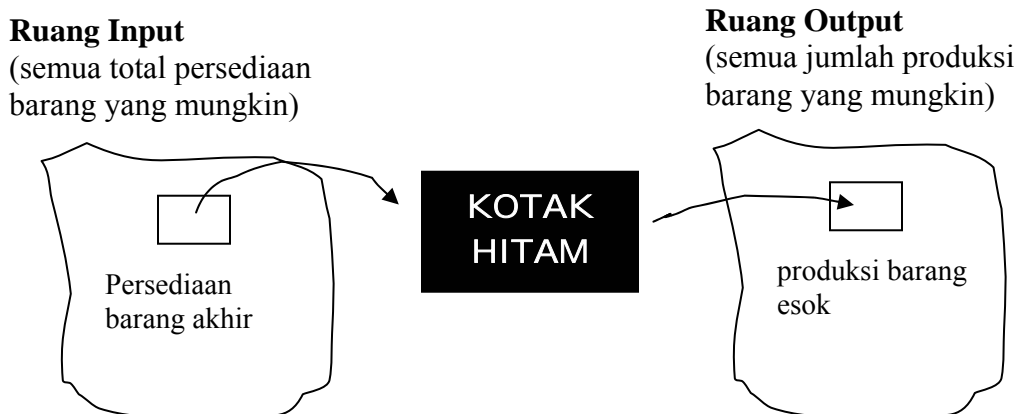
$$MB[\text{gelandangan, pengangguran}] = [0,7] * [0,9] = 0,63$$

VII. LOGIKA FUZZY

Logika fuzzy adalah suatu cara untuk memetakan suatu ruang input ke dalam suatu ruang output.
Skema logika fuzzy :



Antara input dan output terdapat suatu kotak hitam yang harus memetakan input ke output yang sesuai.
Misal :



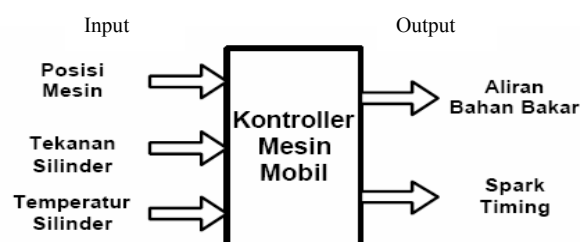
Pemetaan input-output pada masalah produksi : “diberikan data persediaan barang, berapa jumlah barang yang harus diproduksi ?”

Ada beberapa cara/metode yang mampu bekerja di kotak hitam tersebut, misal : sistem fuzzy, jaringan syaraf tiruan, sistem linear, sistem pakar, persamaan diferensial, dll.

Namun menurut Prof. Lotfi A. Zadeh, penemu teori logika fuzzy di tahun 1960-an :



“Pada hampir semua kasus kita dapat menghasilkan suatu produk tanpa menggunakan logika fuzzy, namun menggunakan fuzzy akan lebih cepat dan lebih murah”



ALASAN MENGGUNAKAN FUZZY

1. Konsep logika fuzzy mudah dimengerti
2. Logika fuzzy sangat fleksibel
3. Memiliki toleransi terhadap data-data yang tidak tepat
4. Dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan
5. Logika fuzzy didasarkan pada bahasa alami

APLIKASI LOGIKA FUZZY

1. Tahun 1990 pertama kali mesin cuci dengan logika fuzzy di Jepang (Matsushita Electric Industrial Company). Sistem fuzzy digunakan untuk menentukan putaran yang tepat secara otomatis berdasarkan jenis dan banyaknya kotoran serta jumlah yang akan dicuci. Input yang digunakan : seberapa kotor, jenis kotoran, banyaknya yang dicuci. Mesin ini 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.
2. Transmisi otomatis pada mobil Nissan, menghemat bensin 12 – 17 %
3. Kereta bawah tanah Sendai mengontrol pemberhentian otomatis pada area tertentu
4. Ilmu kedokteran dan biologi, seperti sistem diagnosis kanker
5. Manajemen dan pengambilan keputusan, misal tata letak pabrik berdasarkan logika fuzzy, pembuatan games berdasarkan logika fuzzy, dll
6. Ilmu lingkungan, misal kendali kualitas air, prediksi cuaca
7. Teknik, misal perancangan jaringan komputer, prediksi adanya gempa bumi, dll
8. dsb

KONSEP DASAR LOGIKA FUZZY

HIMPUNAN TEGAS (CRISP)

= nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A[x]$, memiliki 2 kemungkinan, yaitu :

- 1, yang berarti bahwa item tersebut (x) anggota himpunan A
- 0, yang berarti bahwa item tersebut (x) bukan anggota himpunan A

contoh :

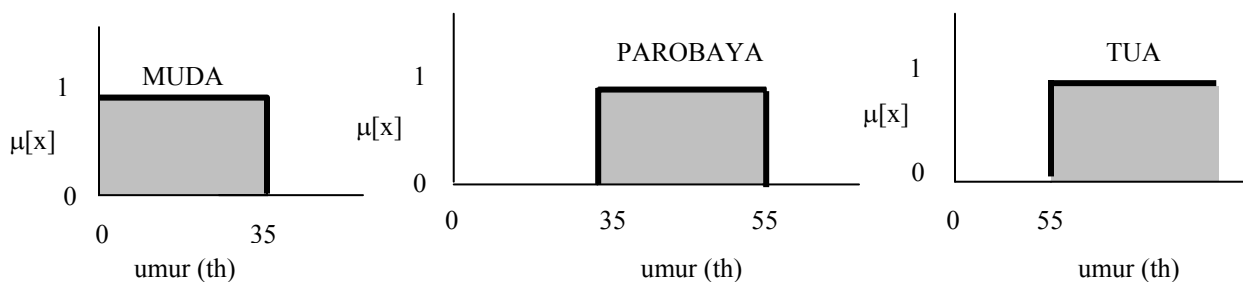
- $S = [1,2,3,4,5,6]$ adalah semesta pembicaraan
 $A = [1,2,3]$
 $B = [3,4,5]$

Jadi :

nilai keanggotaan 2 pada himpunan $A \rightarrow \mu_A[2] = 1$, karena $2 \in A$
 nilai keanggotaan 3 pada himpunan $A \rightarrow \mu_A[3] = 1$, karena $3 \in A$
 nilai keanggotaan 4 pada himpunan $A \rightarrow \mu_A[4] = 0$, karena $4 \notin A$
 nilai keanggotaan 2 pada himpunan $B \rightarrow \mu_B[2] = 0$, karena $2 \notin B$
 nilai keanggotaan 3 pada himpunan $B \rightarrow \mu_B[3] = 1$, karena $3 \in B$

- misal variabel umur dibagi menjadi 3 kategori, yaitu :
 MUDA umur < 35 tahun
 PAROBAYA $35 \leq \text{umur} \leq 55$ tahun
 TUA umur > 55 tahun

Nilai keanggotaan secara grafis, himpunan MUDA, PAROBAYA, TUA :



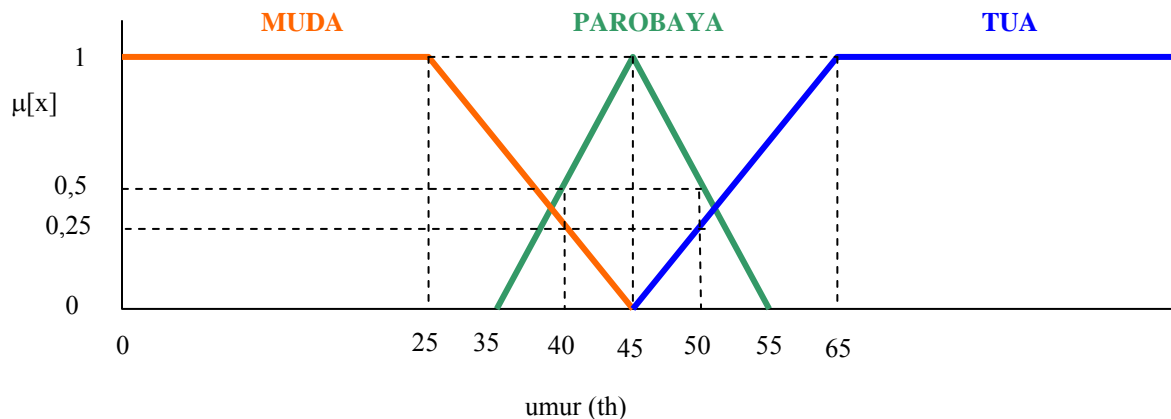
usia 34 tahun maka dikatakan MUDA $\rightarrow \mu_{MUDA}[34] = 1$
 usia 35 tahun maka dikatakan TIDAKMUDA $\rightarrow \mu_{MUDA}[35] = 0$
 usia 35 tahun maka dikatakan PAROBAYA $\rightarrow \mu_{PAROBAYA}[35] = 1$
 usia 34 tahun maka dikatakan TIDAKPAROBAYA $\rightarrow \mu_{PAROBAYA}[34] = 0$
 usia 35 tahun kurang 1 hari maka dikatakan TIDAKPAROBAYA $\rightarrow \mu_{PAROBAYA}[35 \text{ th} - 1 \text{ hari}] = 0$
 usia 35 tahun lebih 1 hari maka dikatakan TIDAKMUDA $\rightarrow \mu_{MUDA}[35 \text{ th} + 1 \text{ hari}] = 0$

Himpunan crisp untuk menyatakan umur bisa tidak adil karena adanya perubahan kecil saja pada suatu nilai mengakibatkan perbedaan kategori yang cukup signifikan.

HIMPUNAN FUZZY

Himpunan fuzzy digunakan untuk mengantisipasi hal tersebut diatas. Seseorang dapat masuk dalam 2 himpunan yang berbeda, MUDA dan PAROBAYA, PAROBAYA dan TUA, dsb. Seberapa besar eksistensinya dalam himpunan tersebut dapat dilihat pada nilai/derajat keanggotaannya.

Himpunan fuzzy untuk variabel UMUR :



usia 40 tahun termasuk dalam himpunan MUDA dengan $\mu_{MUDA}[40] = 0,25$

termasuk juga dalam himpunan PAROBAYA dengan $\mu_{PAROBAYA}[40] = 0,5$

usia 50 tahun termasuk dalam himpunan TUA dengan $\mu_{TUA}[50] = 0,25$

termasuk juga dalam himpunan PAROBAYA dengan $\mu_{PAROBAYA}[50] = 0,5$

Himpunan crisp, nilai keanggotaan hanya 0 dan 1.

Himpunan fuzzy, derajat/nilai keanggotaan terletak pada rentang 0 sampai 1 sehingga :

Bila x memiliki derajat keanggotaan fuzzy $\mu_A[x] = 0 \rightarrow x$ bukan anggota himpunan A

Bila x memiliki derajat keanggotaan fuzzy $\mu_A[x] = 1 \rightarrow x$ anggota penuh himpunan A

FUNGSI KEANGGOTAAN (Membership function)

= suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai/derajat keanggotaannya yang memiliki interval antara 0 sampai 1.

Contoh grafik diatas adalah suatu fungsi keanggotaan untuk variabel UMUR yang dibagi menjadi 3 kategori atau 3 himpunan fuzzy yaitu MUDA, PAROBAYA, TUA, dimana dapat direpresentasikan sebagai berikut :

$$\mu_{MUDA}[x] = \begin{cases} 1, & x \leq 25 \\ \frac{45-x}{45-25}, & 25 < x < 45 \\ 0, & x \geq 45 \end{cases} \quad \mu_{TUA}[x] = \begin{cases} 0, & x \leq 45 \\ \frac{x-45}{65-45}, & 45 < x < 65 \\ 1, & x \geq 65 \end{cases}$$

$$\mu_{PAROBAYA}[x] = \begin{cases} 0, & x \leq 35 \text{ atau } x \geq 55 \\ \frac{x-35}{45-35}, & 35 < x < 45 \\ \frac{55-x}{55-45}, & 45 \leq x \leq 55 \end{cases}$$

WATAK KEKABURAN

Perhatikan pernyataan dibawah ini :

Mesin yang digunakan terus-menerus akan cepat panas

→ kita tidak dapat menentukan dengan tepat batasan terus-menerus, cepat, dan panas

Jika air pancuran terlalu panas maka naikkan aliran air dingin perlahan-lahan

→ kita tidak dapat menentukan dengan tepat batasan terlalu panas, menaikkan, air yang dingin, dan perlahan-lahan



maka solusinya dengan menggunakan LOGIKA FUZZY (logika samar)

VARIABEL LINGUSTIK

- Variabel linguistik = sebuah variabel yang memiliki nilai berupa kata-kata dalam bahasa alamiah bukan angka.
- Mengapa menggunakan kata/kalimat daripada angka ? → karena peranan linguistik memang kurang spesifik dibandingkan angka, namun informasi yang disampaikan lebih informatif.

Contoh, jika “KECEPATAN” adalah variabel linguistik, maka nilai linguistik untuk variabel kecepatan adalah, misalnya “LAMBAT”, “SEDANG”, “CEPAT”. Hal ini sesuai dengan kebiasaan manusia sehari-hari dalam menilai sesuatu, misalnya : “Ia mengendarai mobil dengan cepat”, tanpa memberikan nilai berapa kecepatannya.

- Setiap variabel linguistik berkaitan dengan sebuah fungsi keanggotaan.
- Menurut Wang (1997) definisi formal dari variabel linguistik diberikan sebagai berikut:

Sebuah variabel linguistik dikarakterisasi oleh (X, T(x), U, M), dimana :

X = Nama variabel (variabel linguistik) yang menjadi objek

T(x) = Himpunan semua istilah (nilai-nilai) linguistik yang terkait dengan (nama) variabel (X) yang menggambarkan objek tersebut

U = Domain fisik aktual/ruang lingkup dimana variabel linguistik X mengambil nilai-nilai kuantitatifnya/nilai numeris (crisp) → himpunan semesta

M = Suatu aturan semantik yang menghubungkan setiap nilai linguistik dalam T dengan suatu himpunan fuzzy dalam U.

Dari contoh diatas, maka diperoleh:

X = kecepatan

U = [0 , 100] → maksudnya domain/ruang lingkup kecepatan misal dari 0 sampai 100 km/jam

T(kecepatan) = {lambat, sedang, cepat} → maksudnya variabel kecepatan terbagi menjadi 3 himpunan fuzzy yaitu lambat, sedang, cepat

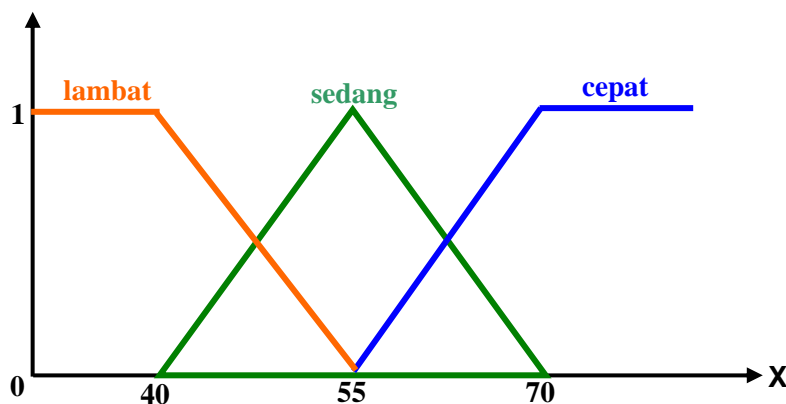
Maka M untuk setiap X, M(x) adalah: M(lambat), M(sedang), M(cepat)

M(lambat) = himpunan fuzzynya “kecepatan dibawah 40 Km/jam” dengan fungsi keanggotaan

μ_{lambat}
M(sedang) = himpunan fuzzynya “kecepatan mendekati 55 Km/jam” dengan fungsi keanggotaan μ_{sedang} .

μ_{cepat}
M(cepat) = himpunan fuzzynya “kecepatan diatas 70 Km/jam ” dengan fungsi keanggotaan

Gambar grafik fungsi keanggotaannya sebagai berikut :



Grafik fungsi keanggotaan kecepatan

Sehingga himpunan fuzzy untuk :

M(lambat) = {(0,1),(1,1),(2,1), ..., (40,1), ..., (47,0.533), ..., (55,0), (56,0), ..., (100,0)}

M(sedang) = {(0,0),(1,0),(2,0), ..., (40,0), ..., (47,0.533), ..., (55,1), (56,0.933), ..., (100,0)}

M(cepat) = {(0,0),(1,1),(2,1), ..., (40,1), ..., (47,0), ..., (55,0), (56,0.066), ..., (68,0.866) (70,1),... ,(100,1)}

OPERASI DASAR HIMPUNAN FUZZY (Operator Zadeh)

Digunakan untuk mengkombinasi dan memodifikasi himpunan fuzzy. Nilai keanggotaan sebagai hasil dari operasi 2 himpunan disebut *fire strength* atau *a predikat*.

Operator	Operasi	Fungsi keanggotaan
AND	Intersection	$\mu_{(A \cap B)}(x) = \min[\mu_A(x), \mu_B(x)]$
OR	Union	$\mu_{(A \cup B)}(x) = \max[\mu_A(x), \mu_B(x)]$
NOT	Complement	$\mu_A^c(x) = 1 - \mu_A(x)$

Fuzzifikasi

= mengambil masukan nilai crisp dan menentukan derajat dimana nilai-nilai tersebut menjadi anggota dari setiap himpunan fuzzy yang sesuai → membuat fungsi keanggotaan

Contoh : masukan crisp 75 derajat ditransformasikan sebagai panas dalam bentuk fuzzy dengan derajat keanggotaan 0.80.

Inferensi

- mengaplikasikan aturan pada masukan fuzzy yang dihasilkan dalam proses fuzzyfikasi
- mengevaluasi tiap aturan dengan masukan yang dihasilkan dari proses fuzzyfikasi dengan mengevaluasi hubungan atau derajat keanggotaan antecedent/premis setiap aturan.
- derajat keanggotaan/nilai kebenaran dari premis digunakan untuk menentukan nilai kebenaran bagian consequent/kesimpulan

Proses penentuan Output Crisp

Tergantung teori/metode yang digunakan

METODE TSUKAMOTO

Suatu perusahaan makanan kaleng akan memproduksi makanan jenis ABC.

Dari data 1 bulan terakhir, permintaan terbesar mencapai 5000 kemasan/hari, dan permintaan terkecil sampai 1000 kemasan/hari.

Persediaan barang di gudang terbanyak sampai 600 kemasan/hari, dan terkecil pernah 100 kemasan/hari.

Dengan segala keterbatasannya, sampai saat ini, perusahaan baru mampu memproduksi barang maksimum 7000 kemasan/hari, serta demi efisiensi mesin dan SDM perusahaan memproduksi paling tidak 2000 kemasan.

Apabila proses produksi perusahaan tersebut menggunakan 4 aturan fuzzy :

R1 : IF permintaan turun AND persediaan banyak THEN produksi barang berkurang

R2 : IF permintaan turun AND persediaan sedikit THEN produksi barang berkurang

R3 : IF permintaan naik AND persediaan banyak THEN produksi barang bertambah

R4 : IF permintaan naik AND persediaan sedikit THEN produksi barang bertambah

Berapa kemasan makanan jenis ABC yang harus diproduksi, jika jumlah permintaan sebanyak 4000 kemasan dan persediaan di gudang masih 300 kemasan?

Solusi :

FUZZIFIKASI → membuat fungsi keanggotaan

Ada 3 variabel fuzzy yang akan dimodelkan, yaitu :

1. PERMINTAAN

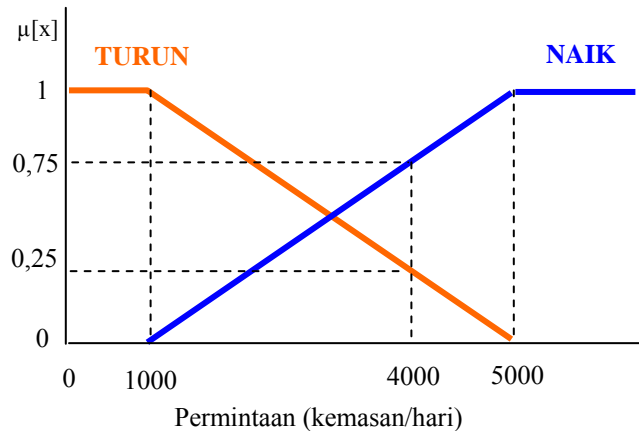
Terdiri atas 2 himpunan fuzzy yaitu NAIK dan TURUN

$$\mu_{\text{permintaanTURUN}}[x] = \begin{cases} 1, & x \leq 1000 \\ \frac{5000 - x}{5000 - 1000}, & 1000 < x < 5000 \\ 0, & x \geq 5000 \end{cases}$$

$$\mu_{\text{permintaanNAIK}}[x] = \begin{cases} 0, & x \leq 1000 \\ \frac{x - 1000}{5000 - 1000}, & 1000 < x < 5000 \\ 1, & x \geq 5000 \end{cases}$$

Derajat/nilai keanggotaan : $\mu_{\text{permintaanTURUN}}[4000]$ $= (5000 - 4000) / 4000$ $= 0,25$

$\mu_{\text{permintaanNAIK}}[4000]$ $= (4000 - 1000) / 4000$ $= 0,75$



2. PERSEDIAAN

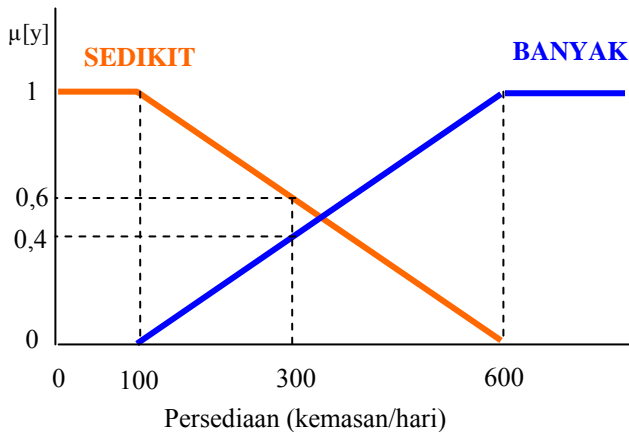
Terdiri atas 2 himpunan fuzzy yaitu SEDIKIT dan BANYAK

$$\mu_{\text{persediaanSEDIKIT}}[y] = \begin{cases} 1, & y \leq 100 \\ \frac{600 - y}{600 - 100}, & 100 < x < 600 \\ 0, & y \geq 600 \end{cases}$$

$$\mu_{\text{persediaanBANYAK}}[y] = \begin{cases} 0, & y \leq 100 \\ \frac{y - 100}{600 - 100}, & 100 < y < 600 \\ 1, & y \geq 600 \end{cases}$$

Derajat/nilai keanggotaan :
 $\mu_{\text{persediaanSEDIKIT}}[300]$
 $= (600 - 300) / 500$
 $= 0,6$

$\mu_{\text{persediaanBANYAK}}[300]$
 $= (300 - 100) / 500$
 $= 0,4$

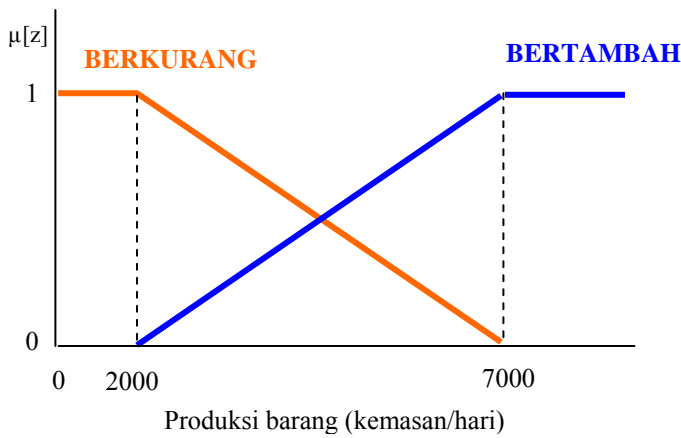


3. PRODUKSI BARANG

Terdiri atas 2 himpunan fuzzy yaitu BERKURANG dan BERTAMBAH

$$\mu_{\text{produksiBERKURANG}}[z] = \begin{cases} 1, & z \leq 2000 \\ \frac{7000 - z}{7000 - 2000}, & 2000 < z < 7000 \\ 0, & z \geq 7000 \end{cases}$$

$$\mu_{\text{produksiBERTAMBAH}}[z] = \begin{cases} 0, & z \leq 2000 \\ \frac{z - 2000}{7000 - 2000}, & 2000 < z < 7000 \\ 1, & z \geq 7000 \end{cases}$$



INFERENSI → evaluasi rule, operator Zadeh digunakan sehingga didapat α predikat untuk menentukan z

Aturan 1

R1 : IF permintaan turun AND persediaan banyak THEN produksi barang berkurang

$$\begin{aligned}\alpha \text{ predikat1} &= \mu_{\text{permintaanTURUN}} \cap \mu_{\text{persediaanBANYAK}} \\ &= \min(\mu_{\text{permintaanTURUN}}[4000] \cap \mu_{\text{persediaanBANYAK}}[300]) \\ &= \min(0.25; 0.4) = 0,25\end{aligned}$$

Untuk α predikat1 = 0,25 pada himpunan produksi BERKURANG

$$\begin{aligned}&= (7000 - z) / (7000 - 2000) = 0,25 \\ &\Rightarrow z1 = 5750\end{aligned}$$

Aturan 2

R1 : IF permintaan turun AND persediaan sedikit THEN produksi barang berkurang

$$\begin{aligned}\alpha \text{ predikat2} &= \mu_{\text{permintaanTURUN}} \cap \mu_{\text{persediaanSEDIKIT}} \\ &= \min(\mu_{\text{permintaanTURUN}}[4000] \cap \mu_{\text{persediaanSEDIKIT}}[300]) \\ &= \min(0.25; 0.6) = 0,25\end{aligned}$$

Untuk α predikat2 = 0,25 pada himpunan produksi BERKURANG

$$\begin{aligned}&= (7000 - z) / (7000 - 2000) = 0,25 \\ &\Rightarrow z2 = 5750\end{aligned}$$

Aturan 3

R1 : IF permintaan naik AND persediaan banyak THEN produksi barang bertambah

$$\begin{aligned}\alpha \text{ predikat3} &= \mu_{\text{permintaanNAIK}} \cap \mu_{\text{persediaanBANYAK}} \\ &= \min(\mu_{\text{permintaanNAIK}}[4000] \cap \mu_{\text{persediaanBANYAK}}[300]) \\ &= \min(0.75; 0.4) = 0,4\end{aligned}$$

Untuk α predikat3 = 0,4 pada himpunan produksi BERTAMBAH

$$\begin{aligned}&= (z - 2000) / (7000 - 2000) = 0,4 \\ &\Rightarrow z3 = 4000\end{aligned}$$

Aturan 4

R1 : IF permintaan naik AND persediaan sedikit THEN produksi barang bertambah

$$\begin{aligned}\alpha \text{ predikat4} &= \mu_{\text{permintaanNAIK}} \cap \mu_{\text{persediaanSEDIKIT}} \\ &= \min(\mu_{\text{permintaanNAIK}}[4000] \cap \mu_{\text{persediaanSEDIKIT}}[300]) \\ &= \min(0.75; 0.6) = 0,6\end{aligned}$$

Untuk α predikat4 = 0,6 pada himpunan produksi BERTAMBAH

$$\begin{aligned}&= (z - 2000) / (7000 - 2000) = 0,6 \\ &\Rightarrow z4 = 5000\end{aligned}$$

MENENTUKAN OUTPUT CRISP

Pada metode TSUKAMOTO untuk menentukan output crisp menggunakan *rata-rata berbobot* yaitu :

$$z = \frac{\alpha_1 z_1 + \alpha_2 z_2 + \alpha_3 z_3 + \alpha_4 z_4}{\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4}$$

$$= \frac{0,25 * 5750 + 0,25 * 5750 + 0,4 * 4000 + 0,6 * 5000}{0,25 + 0,25 + 0,4 + 0,6}$$

$$= 7475 / 1,5 = 4983$$

Jadi jumlah makanan kaleng yang harus diproduksi 4983 kemasan

BASIS DATA FUZZY METODE TAHANIBasis Data Standar

Data karyawan mentah

NIP	Nama	Tgl Lahir	Thn Masuk	Gaji/bl
01	Lia	03-06-1972	1996	750000
02	Iwan	23-09-1954	1985	1500000
03	Sari	12-12-1966	1988	1255000
04	Andi	06-03-1965	1998	1040000
05	Budi	04-12-1960	1990	950000
06	Amir	18-11-1963	1989	1600000
07	Rian	28-05-1965	1997	1250000
08	Kiki	09-07-1971	2001	550000
09	Alda	14-08-1967	1999	735000
10	Yoga	17-09-1977	2000	860000

Data karyawan setelah diolah

NIP	Nama	Umur(th)	Masa kerja(th)	Gaji/bl
01	Lia	30	6	750000
02	Iwan	48	17	1500000
03	Sari	36	14	1255000
04	Andi	37	4	1040000
05	Budi	42	12	950000
06	Amir	39	13	1600000
07	Rian	37	5	1250000
08	Kiki	32	1	550000
09	Alda	35	3	735000
10	Yoga	25	2	860000

- Query untuk mendapatkan informasi karyawan umurnya kurang dari 35 tahun :
SELECT NAMA FROM KARYAWAN WHERE (UMUR < 35)
Hasil : Lia, Kiki, Yoga
- Query untuk mendapatkan informasi karyawan gajinya lebih dari 1 juta :
SELECT NAMA FROM KARYAWAN WHERE (GAJI > 1000000)
Hasil : Iwan, Sari, Andi, Amir, Rian
- Query untuk mendapatkan informasi karyawan masa kerja kurang dari atau sama dengan 5 tahun tetapi gaji sudah lebih dari 1 juta :
SELECT NAMA FROM KARYAWAN WHERE (MASAKERJA <= 5) AND (GAJI > 1000000)
Hasil : Andi, Rian

Basis Data Fuzzy Model Tahani

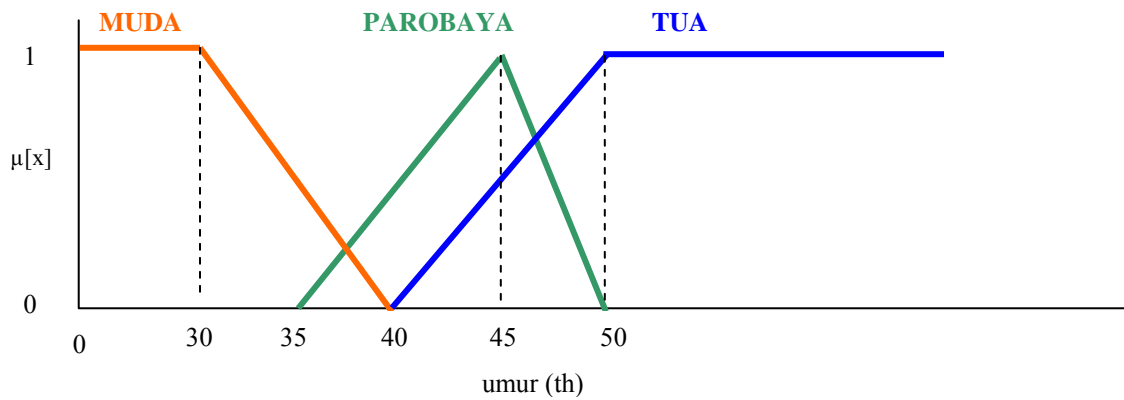
UMUR

Umur karyawan dikategorikan ke himpunan MUDA, PAROBAYA, TUA

$$\mu \text{ MUDA}[x] = \begin{cases} 1, & x \leq 30 \\ \frac{40 - x}{10}, & 30 < x < 40 \\ 0, & x \geq 40 \end{cases}$$

$$\mu \text{ TUA}[x] = \begin{cases} 0, & x \leq 40 \\ \frac{x - 40}{50 - 40}, & 40 < x < 50 \\ 1, & x \geq 50 \end{cases}$$

$$\mu \text{ PAROBAYA}[x] = \begin{cases} 1, & x \leq 35 \text{ atau } x \geq 50 \\ \frac{x - 35}{45 - 35}, & 35 < x < 45 \\ \frac{50 - x}{50 - 45}, & 45 \leq x \leq 50 \end{cases}$$



Karyawan berdasarkan umur

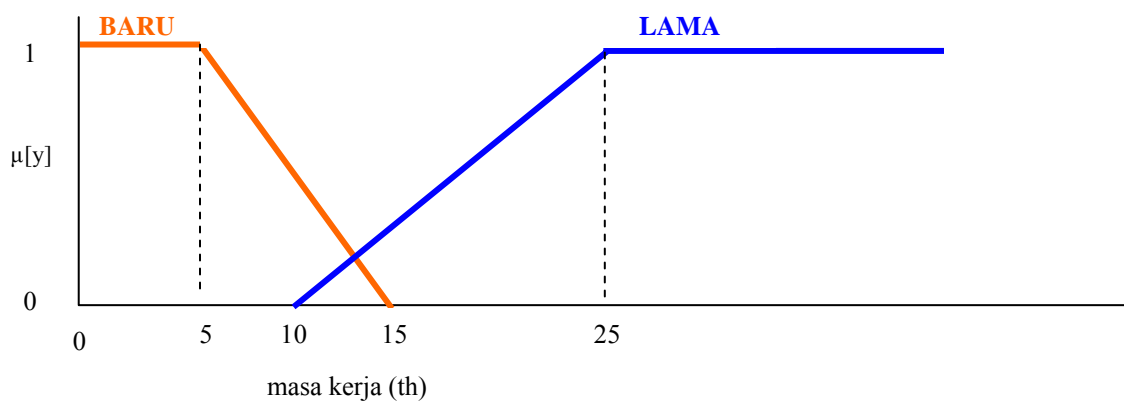
NIP	Nama	Umur(th)	Derajat keanggotaan (μ[x])		
			MUDA	PAROBAYA	TUA
01	Lia	30	1	0	0
02	Iwan	48	0	0,4	0,8
03	Sari	36	0,4	0,1	0
04	Andi	37	0,3	0,2	0
05	Budi	42	0	0,7	0,2
06	Amir	39	0,1	0,4	0
07	Rian	37	0,3	0,2	0
08	Kiki	32	0,8	0	0
09	Alda	35	0,5	0	0
10	Yoga	25	1	0	0

MASA KERJA

Masa kerja karyawan dikategorikan ke himpunan BARU, LAMA

$$\mu \text{ BARU}[y] = \begin{cases} 1, & y \leq 5 \\ \frac{15 - y}{15 - 5}, & 5 < y < 15 \\ 0, & x \geq 15 \end{cases}$$

$$\mu \text{ LAMA}[y] = \begin{cases} 0, & y \leq 10 \\ \frac{y - 10}{15 - 10}, & 10 < y < 25 \\ 1, & y \geq 25 \end{cases}$$



Karyawan berdasarkan masa kerja

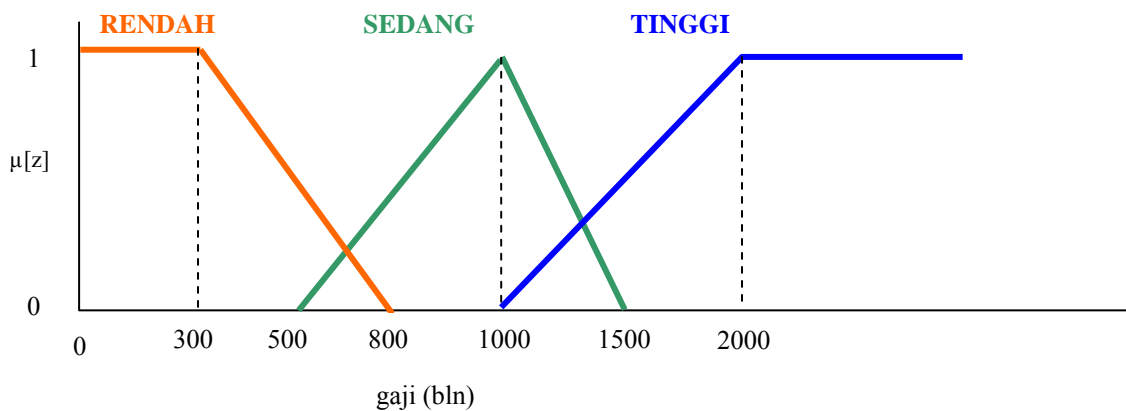
NIP	Nama	Masa kerja(th)	Derajat keanggotaan ($\mu[x]$)	
			BARU	LAMA
01	Lia	6	0,9	0
02	Iwan	17	0	0,467
03	Sari	14	0,1	0,267
04	Andi	4	1	0
05	Budi	12	0,3	0,133
06	Amir	13	0,2	0,200
07	Rian	5	1	0
08	Kiki	1	1	0
09	Alda	3	1	0
10	Yoga	2	1	0

GAJI

Gaji karyawan dikategorikan ke himpunan RENDAH, SEDANG, TINGGI

$$\mu \text{ RENDAH}[z] = \begin{cases} 1, & z \leq 300 \\ \frac{800 - z}{800 - 300}, & 300 < z < 800 \\ 0, & z \geq 800 \end{cases} \quad \mu \text{ TINGGI}[z] = \begin{cases} 0, & z \leq 1000 \\ \frac{z - 1000}{2000 - 1000}, & 1000 < z < 2000 \\ 1, & z \geq 2000 \end{cases}$$

$$\mu \text{ SEDANG}[z] = \begin{cases} 0, & z \leq 500 \text{ atau } z \geq 1500 \\ \frac{z - 500}{1000 - 500}, & 500 < z < 1000 \\ \frac{1500 - z}{1500 - 1000}, & 1000 \leq z \leq 1500 \end{cases}$$



Karyawan berdasarkan gaji

NIP	Nama	Gaji(bln)	Derajat keanggotaan ($\mu[x]$)		
			RENDAH	SEDANG	TINGGI
01	Lia	750000	0,1	0,50	0
02	Iwan	1500000	0	0,49	0,255
03	Sari	1255000	0	0	0,500
04	Andi	1040000	0	0,92	0,040
05	Budi	950000	0	0,90	0
06	Amir	1600000	0	0	0,600
07	Rian	1250000	0	0,50	0,250
08	Kiki	550000	0,5	0	0
09	Alda	735000	0,13	0	0
10	Yoga	860000	0	0	0

- Query untuk mendapatkan informasi karyawan yang masih muda tapi gaji tinggi :
SELECT NAMA FROM KARYAWAN WHERE (UMUR = "MUDA") AND (GAJI = "TINGGI")

NIP	Nama	Umur	Gaji	Derajat keanggotaan ($\mu[x]$)		
				MUDA	TINGGI	MUDA & TINGGI
03	Sari	36	1500000	0,4	0,5	0,4
07	Rian	37	1250000	0,3	0,25	0,25
06	Amir	39	1600000	0,1	0,6	0,1
04	Andi	37	1040000	0,3	0,04	0,04
01	Lia	30	750000	1	0	0
02	Iwan	48	1255000	0	0,255	0
05	Budi	42	950000	0	0	0
08	Kiki	32	550000	0,8	0	0
09	Alda	35	735000	0,5	0	0
10	Yoga	25	860000	1	0	0

Hasil : Sari, Rian, Amir, Andi
max = Sari

- Query untuk mendapatkan informasi karyawan yang masih muda atau karyawan gaji tinggi :
SELECT NAMA FROM KARYAWAN WHERE (UMUR = "MUDA") OR (GAJI = "TINGGI")

NIP	Nama	Umur	Gaji	Derajat keanggotaan ($\mu[x]$)		
				MUDA	TINGGI	MUDA OR TINGGI
01	Lia	30	750000	1	0	1
10	Yoga	25	860000	1	0	1
08	Kiki	32	550000	0,8	0	0,8
06	Amir	39	1600000	0,1	0,6	0,6
03	Sari	36	1500000	0,4	0,5	0,5
09	Alda	35	735000	0,5	0	0,5
04	Andi	37	1040000	0,3	0,04	0,3
07	Rian	37	1250000	0,3	0,25	0,3
02	Iwan	48	1255000	0	0,255	0,255
05	Budi	42	950000	0	0	0

Hasil : Hanya 1 yang tidak memenuhi kategori tersebut yaitu Budi

- Query untuk mendapatkan informasi karyawan yang masih muda tapi masa kerja sudah lama :
SELECT NAMA FROM KARYAWAN WHERE (UMUR = "MUDA") AND
(MASAKERJA = "LAMA")

NIP	Nama	Umur	Masa kerja	Derajat keanggotaan ($\mu[x]$)		
				MUDA	LAMA	MUDA & LAMA
03	Sari	36	14	0,4	0,267	0,267
07	Amir	39	13	0,1	0,2	0,1
06	Lia	30	6	1	0	0
04	Iwan	48	17	0	0,467	0
01	Andi	37	4	0,3	0	0
02	Budi	42	12	0	0,133	0
05	Rian	37	5	0,3	0	0
08	Kiki	32	1	0,8	0	0
09	Alda	35	3	0,5	0	0
10	Yoga	25	2	1	0	0

Hasil : Sari, Amir
max = Sari

- Query untuk mendapatkan informasi karyawan yang parobaya dan gajinya sedang, atau karyawan yang parobaya tapi masa kerja sudah lama :
SELECT NAMA FROM KARYAWAN WHERE
((UMUR = "PAROBAYA") AND (GAJI = "SEDANG")) OR
((UMUR = "PAROBAYA") AND (MASAKERJA = "LAMA"))

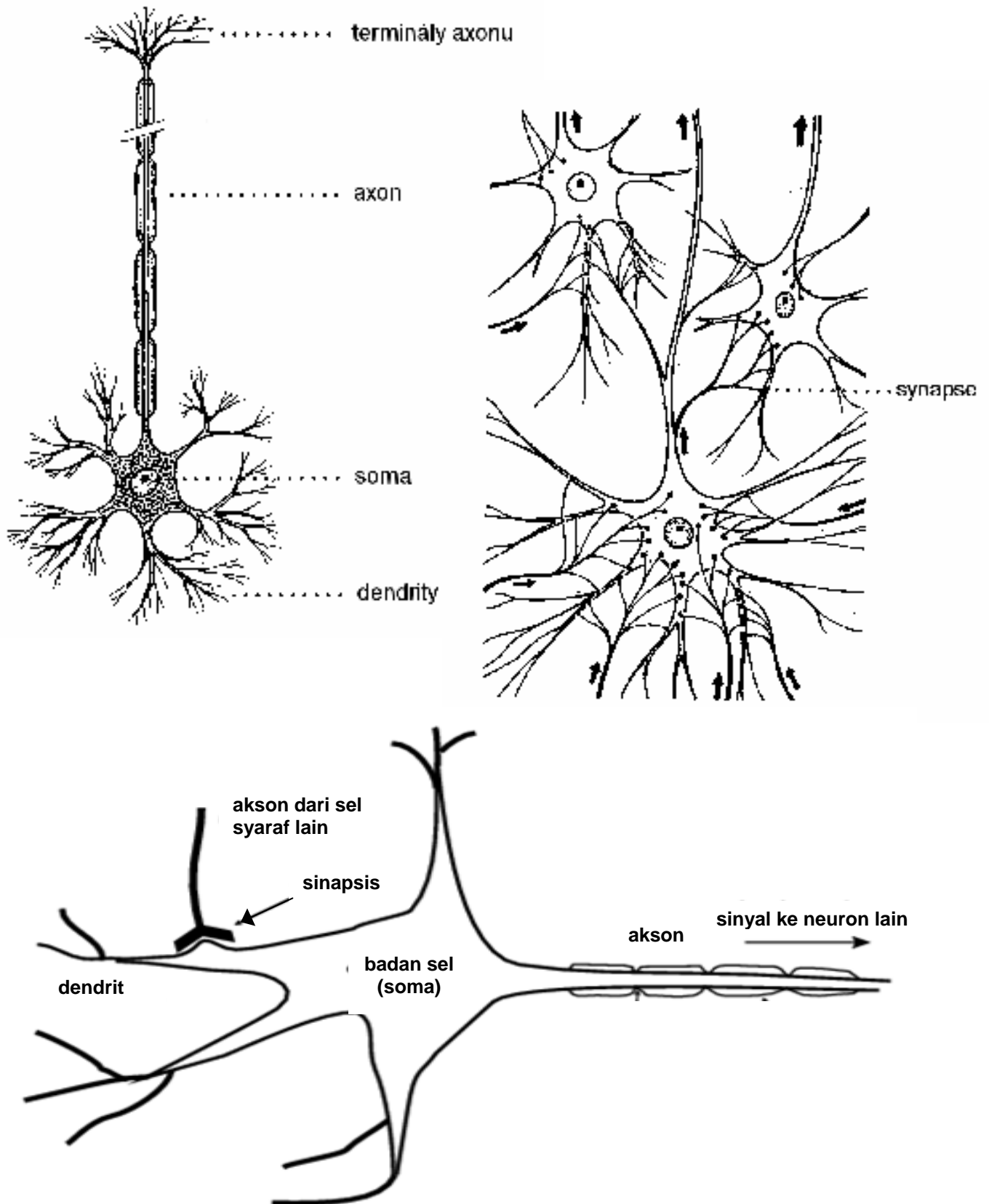
NIP	Nama	Derajat Keanggotaan						(1) OR (2)
		PAROBAYA	SEDANG	PAROBAYA & SEDANG (1)	PAROBAYA	LAMA	PAROBAYA & LAMA (2)	
05	Budi	0,7	0,9	0,7	0,7	0,133	0,133	0,7
02	Iwan	0,4	0,49	0,4	0,4	0,467	0,4	0,4
04	Andi	0,2	0,92	0,2	0,2	0	0	0,2
06	Amir	0,4	0	0	0,4	0,2	0,2	0,2
07	Rian	0,2	0,5	0,2	0,2	0	0	0,2
03	Sari	0,1	0	0	0,1	0,267	0,1	0,1
01	Lia	0	0,5	0	0	0	0	0
08	Kiki	0	0	0	0	0	0	0
09	Alda	0	0	0	0	0	0	0
10	Yoga	0	0	0	0	0	0	0

Hasil : Budi, Iwan, Andi, Amir, Rian, Sari
max = Budi

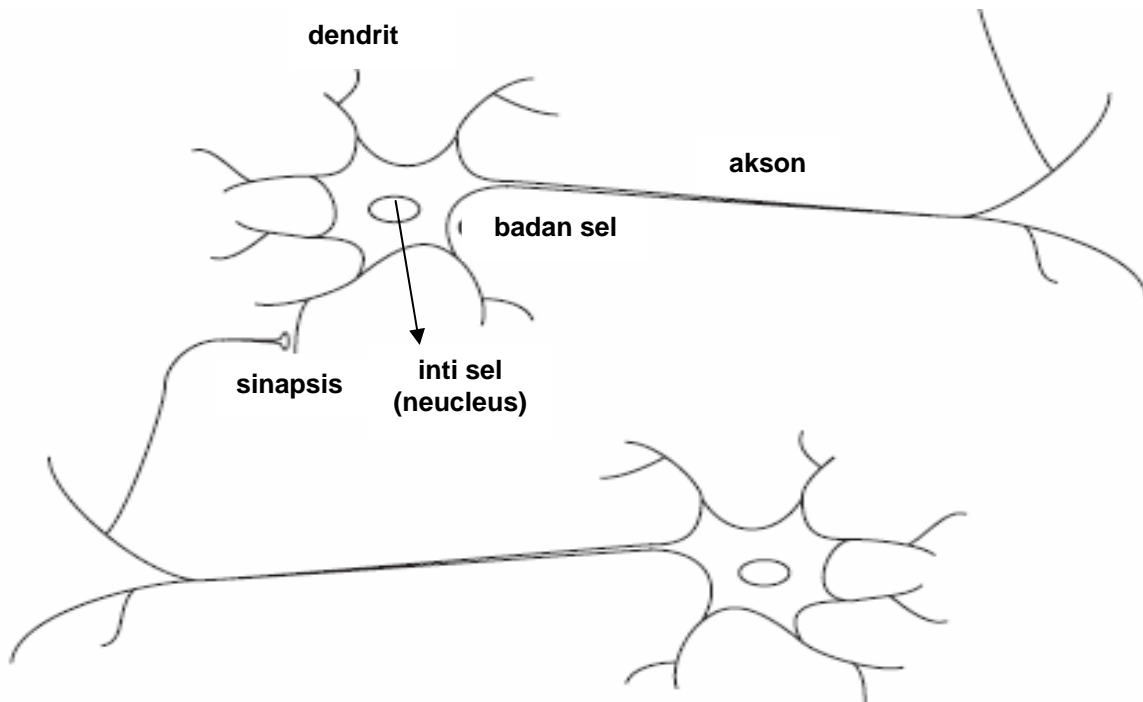
VIII.PENGANTAR JARINGAN SYARAF TIRUAN (JST)

JARINGAN SYARAF BIOLOGIS (JSB)

- Otak manusia berisi sekitar 10^{11} sel syaraf (neuron) yang bertugas untuk memproses informasi yang masuk. Tiap sel syaraf dihubungkan dengan sel syaraf lain hingga sekitar 10^4 sinapsis. Tiap sel bekerja seperti suatu prosesor sederhana. Masing-masing sel tersebut saling berinteraksi sehingga mendukung kemampuan kerja otak manusia.



Gambar NEURON

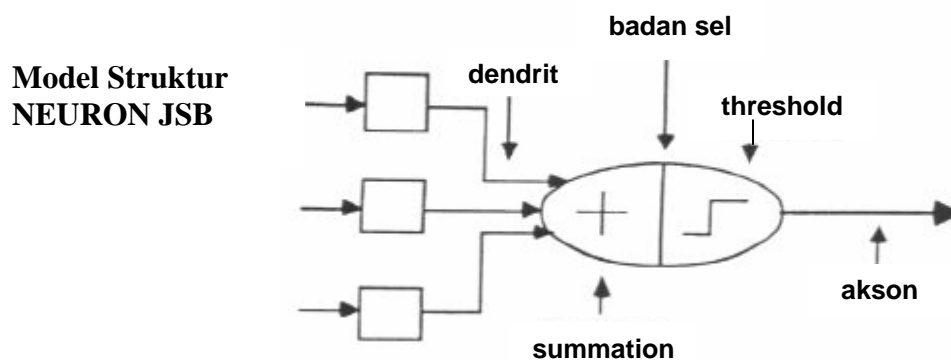


Komponen utama neuron dapat dikelompokkan menjadi 3 bagian :

1. Dendrit = bertugas menerima informasi = jalur input bagi soma
2. Badan sel (soma) = tempat pengolahan informasi
3. Akson = bertugas mengirimkan impuls-impuls sinyal ke sel syaraf lain = jalur output bagi soma

Perhatikan gambar-gambar diatas :

- Sebuah neuron menerima impuls-impuls sinyal (informasi) dari neuron lain melalui dendrit dan mengirimkan sinyal yang dibangkitkan (hasil penjumlahan) oleh badan sel melalui akson.
- Akson dari sel syaraf ini bercabang-cabang dan berhubungan dengan dendrit dari sel syaraf lain dengan cara mengirimkan impuls melalui sinapsis.
- Sinapsis adalah unit fungsional antara 2 buah sel syaraf, misal A dan B, dimana yang satu adalah serabut akson dari neuron A dan satunya lagi adalah dendrit dari neuron B.
- Kekuatan sinapsis bisa menurun / meningkat tergantung seberapa besar tingkat propagasi (penyiaran) sinyal yang diterimanya.
- Impuls-impuls sinyal (informasi) akan diterima oleh neuron lain jika memenuhi batasan tertentu, yang sering disebut dengan nilai ambang (threshold).



JARINGAN SYARAF TIRUAN (JST)

- JST didefinisikan sebagai suatu sistem pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan syaraf manusia (JSB)
- JST tercipta sebagai suatu generalisasi model matematis dari pemahaman manusia (human cognition) yang didasarkan atas asumsi sebagai berikut :
 1. Pemrosesan informasi terjadi pada elemen sederhana yang disebut neuron
 2. Sinyal mengalir diantara sel saraf/neuron melalui suatu sambungan penghubung
 3. Setiap sambungan penghubung memiliki bobot yang bersesuaian. Bobot ini akan digunakan untuk menggandakan / mengalikan sinyal yang dikirim melaluinya.
 4. Setiap sel syaraf akan menerapkan fungsi aktivasi terhadap sinyal hasil penjumlahan berbobot yang masuk kepadanya untuk menentukan sinyal keluarannya.

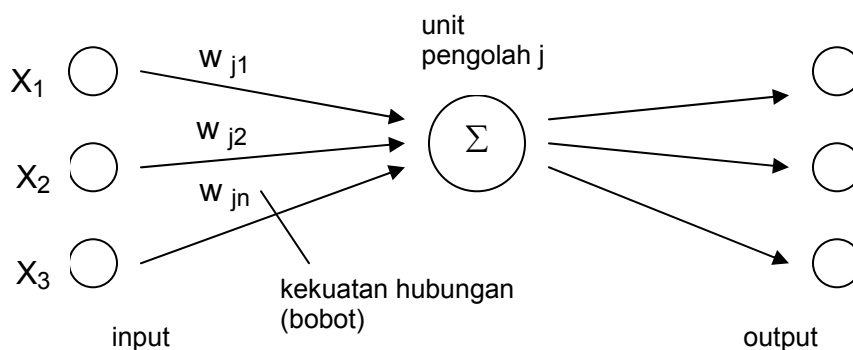
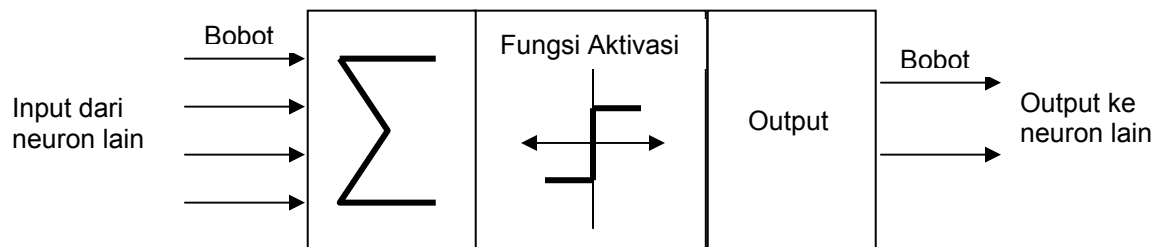
- Perbandingan kemampuan otak manusia dengan CPU

Parameter	Otak manusia	CPU
Elemen pengolah	10^{11} sinapsis	10^8 transistor
Ukuran elemen	10^{-6} m	10^{-6} m
Energi yang digunakan	30 W	30 W (CPU)
Kecepatan pengolah	100 Hz	10^9 Hz
Bentuk komputasi	Paralel terdistribusi	Serial terpusat
Fault tolerant	Ya	Tidak
Proses belajar	Ya	Tidak
Kepandaian	Selalu	Tidak (kadang-kadang)

Analogi JST dengan JSB

JST	J S Biologis
Node / input	Badan sel (soma)
Input	Dendrit
Output	Akson
Bobot	Sinapsis

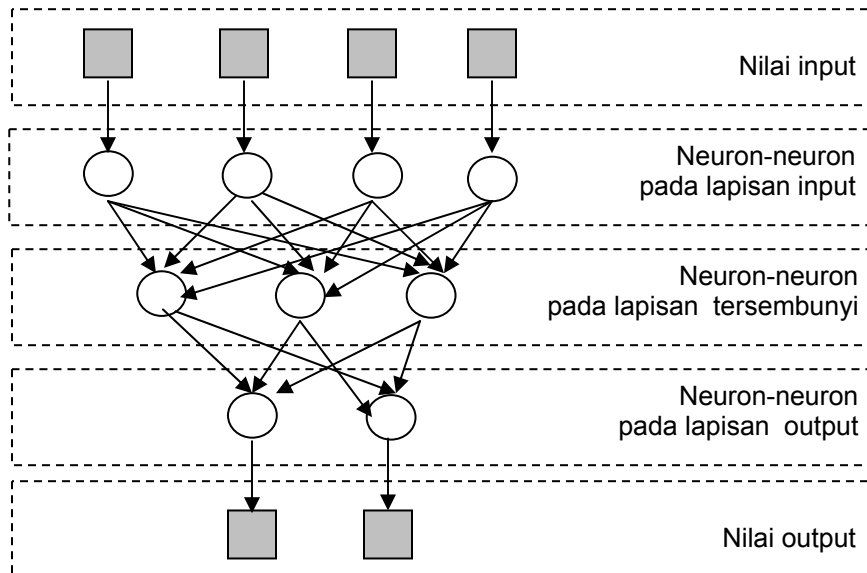
Model Struktur NEURON JST



- Jaringan syaraf tiruan dapat belajar dari pengalaman, melakukan generalisasi atas contoh-contoh yang diperolehnya dan mengabstraksi karakteristik esensial input bahkan untuk data yang tidak relevan.
- Algoritma untuk JST beroperasi secara langsung dengan angka sehingga data yang tidak numerik harus diubah menjadi data numerik.
- JST tidak diprogram untuk menghasilkan keluaran tertentu. Semua keluaran atau kesimpulan yang ditarik oleh jaringan didasarkan pada pengalamannya selama mengikuti proses pembelajaran. Pada proses pembelajaran, ke dalam JST dimasukkan pola-pola input (dan output) lalu jaringan akan diajari untuk memberikan jawaban yang bisa diterima.
- Pada dasarnya karakteristik JST ditentukan oleh :
 1. Pola hubungan antar neuron (disebut arsitektur jaringan)
 2. Metode penentuan bobot-bobot sambungan (disebut dengan pelatihan atau proses belajar jaringan)
 3. Fungsi aktivasi

ARSITEKTUR JST

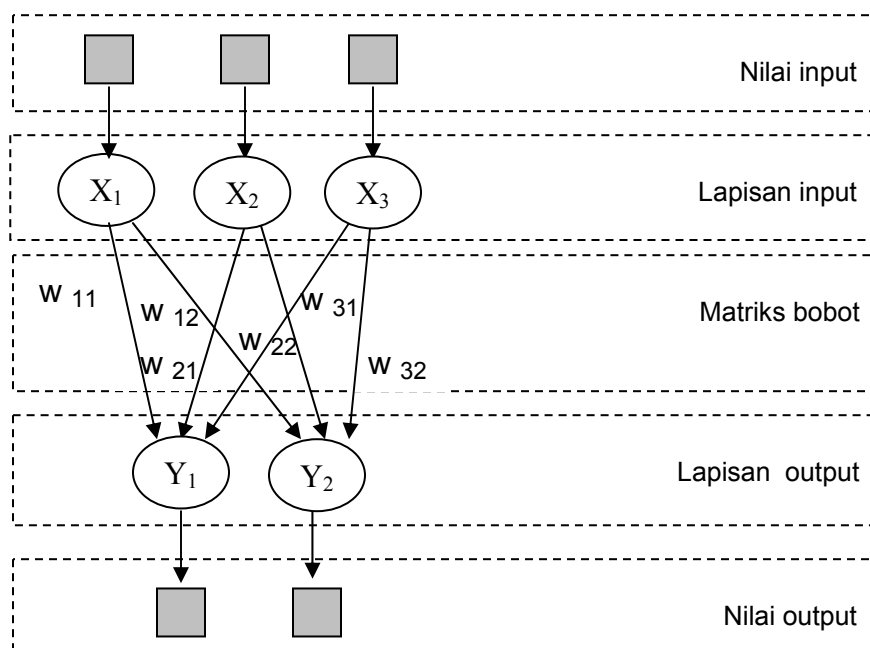
- Pada JST, neuron-neuron akan dikumpulkan dalam lapisan-lapisan (layer) yang disebut dengan lapisan neuron (neuron layers).
- Neuron-neuron pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya.
- Informasi yang diberikan pada jaringan syaraf akan dirambatkan lapisan ke lapisan, mulai dari lapisan input sampai ke lapisan output melalui lapisan tersembunyi (hidden layer).
- Gambar berikut ini jaringan syaraf dengan 3 lapisan dan bukanlah struktur umum jaringan syaraf karena beberapa jaringan syaraf ada yang tidak memiliki lapisan tersembunyi.



- Faktor terpenting dalam menentukan kelakuan suatu neuron adalah fungsi aktivasi dan pola bobotnya.
- Umumnya neuron-neuron yang terletak pada lapisan yang sama akan memiliki keadaan yang sama sehingga pada setiap lapisan yang sama neuron-neuron memiliki fungsi aktivasi yang sama.
- Bila neuron-neuron pada suatu lapisan (misal lapisan tersembunyi) akan dihubungkan dengan neuron-neuron pada lapisan lain (misal lapisan output) maka setiap neuron pada lapisan tersebut (lapisan tersembunyi) juga harus dihubungkan dengan setiap neuron pada lapisan lainnya (lapisan output)
- Macam arsitektur JST ada 3 :

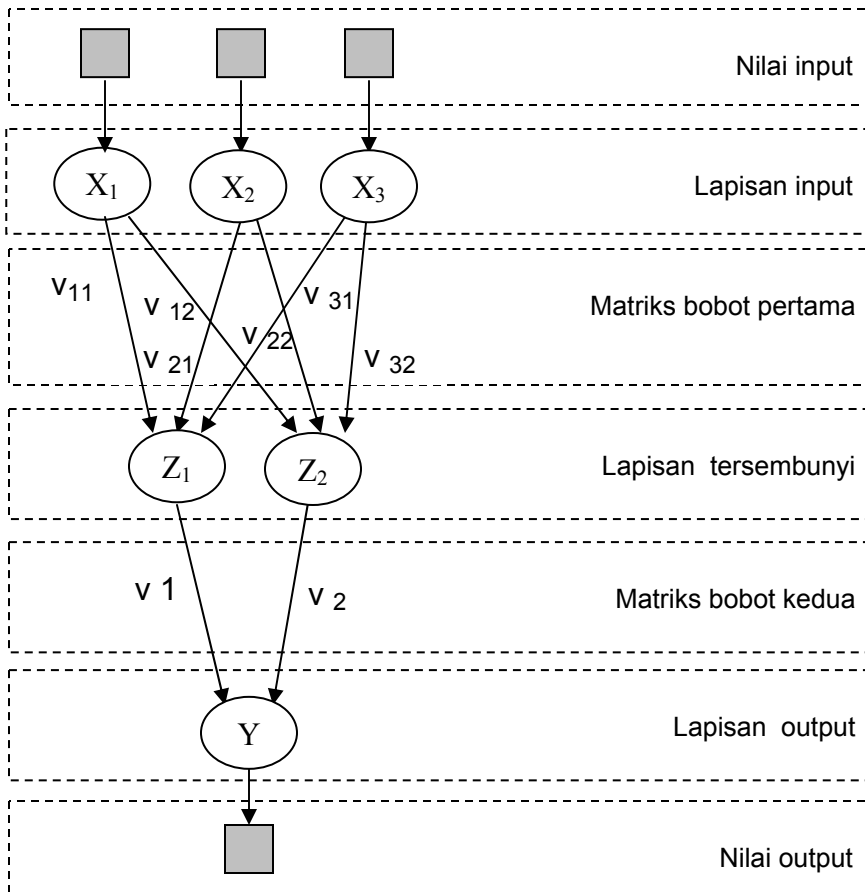
1. Jaringan dengan lapisan tunggal (single layer net)

Hanya memiliki 1 lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima input kemudian secara langsung akan mengolahnya menjadi output tanpa harus melalui lapisan tersembunyi. Pada gambar berikut neuron-neuron pada kedua lapisan saling berhubungan. Seberapa besar hubungan antara 2 neuron ditentukan oleh bobot yang bersesuaian. Semua unit input akan dihubungkan dengan setiap unit output.



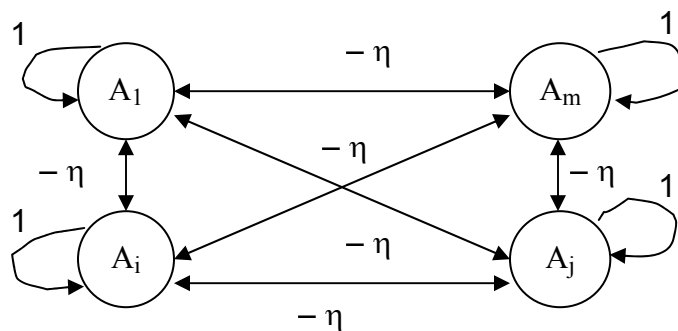
2. Jaringan dengan banyak lapisan (multilayer net)

Memiliki 1 atau lebih lapisan yang terletak diantara lapisan input dan lapisan output. Umumnya ada lapisan bobot-bobot yang terletak antara 2 lapisan yang bersebelahan. Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit daripada lapisan tunggal, tentu saja dengan pembelajaran yang lebih rumit. Pada banyak kasus, pembelajaran pada jaringan dengan banyak lapisan ini lebih sukses dalam menyelesaikan masalah.



3. Jaringan dengan lapisan kompetitif (competitive layer net)

Pada jaringan ini sekumpulan neuron bersaing untuk mendapatkan hak menjadi aktif. Umumnya hubungan antar neuron pada lapisan kompetitif ini tidak diperlihatkan pada diagram arsitektur. Gambar berikut menunjukkan salah satu contoh arsitektur jaringan dengan lapisan kompetitif yang memiliki bobot $-\eta$



PROSES PEMBELAJARAN JARINGAN

- Cara belajar JST :**

Ke dalam JST diinputkan informasi yang sebelumnya telah diketahui hasil keluarannya. Penginputan informasi ini dilakukan lewat node-node atau unit-unit input. Bobot-bobot antarkoneksi dalam suatu arsitektur diberi nilai awal dan kemudian JST dijalankan. Bobot-bobot ini bagi jaringan digunakan untuk belajar dan mengingat suatu informasi. Pengaturan bobot dilakukan secara terus-menerus dan dengan menggunakan kriteria tertentu sampai diperoleh keluaran yang diharapkan.

- Hal yang ingin dicapai dengan melatih/mengajari JST adalah untuk mencapai keseimbangan antara kemampuan memorisasi dan generalisasi.
- Kemampuan memorisasi = kemampuan JST untuk memanggil kembali secara sempurna sebuah pola yang telah dipelajari.
- Kemampuan generalisasi = adalah kemampuan JST untuk menghasilkan respon yang bisa diterima terhadap pola-pola input yang serupa (namun tidak identik) dengan pola-pola yang sebelumnya telah dipelajari.

Hal ini sangat bermanfaat bila pada suatu saat ke dalam JST diinputkan informasi baru yang belum pernah dipelajari, maka JST masih akan tetap dapat memberikan tanggapan yang baik, memberikan keluaran yang paling mendekati.

- Paradigma/metode pembelajaran/pelatihan JST :
 1. **Pembelajaran terawasi (supervised learning)**
 Pada pembelajaran ini kumpulan input yang digunakan, output-outputnya telah diketahui. Perbedaan antara output-output aktual dengan output-output yang diinginkan digunakan untuk mengoreksi bobot JST agar JST dapat menghasilkan jawaban sedekat (semirip) mungkin dengan jawaban yang benar yang telah diketahui oleh JST.
 2. **Pembelajaran tak terawasi (unsupervised learning) / pembelajaran tanpa guru**
 Pada pembelajaran ini, JST mengorganisasi dirinya sendiri untuk membentuk vektor-vektor input yang serupa, tanpa menggunakan data atau contoh-contoh pelatihan. Struktur menggunakan dasar data atau korelasi antara pola-pola data yang dieksplorasi. Paradigma pembelajaran ini mengorganisasi pola-pola ke dalam kategori-kategori berdasarkan korelasi yang ada.
 3. **Gabungan pembelajaran terawasi dan tak terawasi (hybrid)**
 Merupakan kombinasi dari kedua pembelajaran tersebut. Sebagian dari bobot-bobotnya ditentukan melalui pembelajaran terawasi dan sebagian lainnya melalui pembelajaran tak terawasi.

FUNGSI AKTIVASI

- Dipakai ntuk menentukan keluaran suatu neuron
- Merupakan fungsi yang menggambarkan hubungan antara tingkat aktivasi internal (summation function) yang mungkin berbentuk linier atau nonlinear. Beberapa fungsi aktivasi JST diantaranya hard limit, purelin, dan sigmoid. Yang populer digunakan adalah fungsi sigmoid yang memiliki beberapa varian : sigmoid logaritma, sigmoid biner, sigmoid bipolar, sigmoid tangen.
- Hard limit memberikan batasan tegas 0 atau 1, purelin memisahkan secara linier, sigmoid berupa fungsi smooth bernilai antara 0 sampai dengan 1 (bila biner) atau antara -1 sampai 1 (bila bipolar)

SUMMATION FUNCTION

- Fungsi yang digunakan untuk mencari rata-rata bobot dari semua elemen input.
- Bentuk sederhananya adalah dengan mengalikan setiap nilai input (X_j) dengan bobotnya (W_{ij}) dan menjumlahkannya (disebut penjumlahan berbobot atau S_i)

$$S_i = \sum_{j=1}^N W_{ij} * X_j$$

- Diibaratkan dengan sebuah neuron yang memonitor sinyal yang datang dari neuron-neuron lain. Neuron ini menghitung penjumlahan berbobotnya dan kemudian menentukan sinyal untuk dikirim ke neuron-neuron lain.

SUM SQUARE ERROR dan ROOT MEAN SQUARE ERROR

- Perhitungan kesalahan merupakan pengukuran bagaimana jaringan dapat belajar dengan baik sehingga jika dibandingkan dengan pola yang baru akan dengan mudah dikenali.
- Kesalahan pada keluaran jaringan merupakan selisih antara keluaran sebenarnya (current output) dan keluaran yang diinginkan (desired output)
- Selisih yang dihasilkan antara keduanya biasanya ditentukan dengan cara dihitung menggunakan suatu persamaan.

- Sum Square Error (SSE) :
 1. Hitung keluaran jaringan syaraf untuk masukan pertama
 2. Hitung selisih antara nilai keluaran jaringan syaraf dan nilai target/yang diinginkan untuk setiap keluaran
 3. Kuadratkan setiap keluaran kemudian hitung seluruhnya

$$SSE = \sum_p \sum_j (T_{jp} - X_{jp})^2$$

T_{jp} : nilai keluaran jaringan syaraf

X_{jp} : nilai target/yang diinginkan untuk setiap keluaran

- Root Mean Square Error (RMS Error) :
 1. Hitung SSE
 2. Hasilnya dibagi dengan perkalian antara banyaknya data pada pelatihan dan banyaknya keluaran, kemudian diakarkan.

$$RMSError = \sqrt{\frac{\sum_p \sum_j (T_{jp} - X_{jp})^2}{n_p n_o}}$$

T_{jp} : nilai keluaran jaringan syaraf

X_{jp} : nilai target/yang diinginkan untuk setiap keluaran

n_p : jumlah seluruh pola

n_o : jumlah keluaran

Keberhasilan suatu proses belajar JST ditunjukkan dengan besarnya error yang minimum. Pada kondisi inilah JST tersebut dapat digunakan. Ketika ada hal baru yang harus diketahui oleh JST maka proses belajar harus diulang kembali dengan menggunakan informasi-informasi yang lama ditambah dengan informasi-informasi baru.

APLIKASI JARINGAN SYARAF TIRUAN

- Aerospace
autopilot pesawat terbang, simulasi jalur penerbangan, sistem kendali pesawat, perbaikan autopilot, simulasi komponen pesawat
- Otomotif : sistem kendali otomatis mobil
- Keuangan dan perbankan
pendeteksian uang palsu, evaluator aplikasi kredit, pengidentifikasian pola-pola data pasar saham
- Militer
Pengendali senjata, pendeteksi bom, penelusuran target, pembedaan objek, pengendali sensor, sonar, radar, dan pengolahan sinyal citra yang meliputi kompresi data, ekstraksi bagian istimewa, dan penghilangan derau, pengenalan sinyal atau citra.
- Elektronik
Pembuatan perangkat keras yang bisa mengimplementasikan JST secara efisien, machine vision, pengontrol gerakan dan penglihatan robot, sintesis suara
- Broadcast : pencarian klip berita melalui pengenalan wajah
- Keamanan : JST digunakan untuk mengenali mobil dan mengenali wajah oknum
- Medis : analisis sel kanker
- Pengenalan suara : pengenalan percakapan, klasifikasi suara
- Pengenalan tulisan : pengenalan tulisan tangan, penerjemahan tulisan ke dalam tulisan latin
- Matematika : alat pemodelan masalah dimana bentuk eksplisit dari hubungan antara variabel-variabel tertentu tidak diketahui
- Pengenalan benda bergerak
selain pola dari citra diam, JST juga bisa digunakan untuk mendeteksi citra bergerak dari video seperti citra orang yang bergerak, dll.
- JST digunakan sebagai detektor virus komputer, penginderaan bau, dll