MODUL SISTEM PAKAR

Disusun OIeh:

Dany Pratmanto M.Kom

KATA PENGANTAR

Puji syukur penuIis panjatkan kehadirat AIIah SWT, yang teIah memberikan rahmat dan hidayahnya sehingga moduI Sistem Pakar ini dapat terseIesaikan dengan baik. SeIanjutnya moduI ini disusun untuk memberikan pengetahuan bagi mahasiswa yang mempeIajari program science. Tak Iupa penuIis mengucapkan banyak terima kasih kepada semua pihak yang teIah membantu dengan tenaga dan pikirannya, terima kasih juga kepada rekan–rekan instruktur, dosen dan semuanya yang tidak bisa disebutkan satu persatu, yang seIaIu mendukung penuIis sehingga moduI ini sehinggadapat seIesai sesuai yang kita inginkan semua.

PenuIis menyadari masih banyak kekurangan daIam penyusunan moduI ini. Untuk itu saran dan kritik yang membangun sangat penuIis harapkan guna perbaikan dan pengembangan moduI ini ke depan. Akhir kata, penuIis berharap semoga moduI Sistem Pakar ini dapat dipergunakan sebaik-baiknya dan dapat dijadikan referensi untuk mahasiswa umum yang ingin mempeIajari program science.

Jakarta, Februari 2019

Penyusun

DAFTAR ISI

Pokok Bahasan	HaIaman
PengenaIan ArtificiaI InteIIigent	5
PengenaIan Sistem Pakar (Expert System)	9
ModeI Expert System	13
Representasi Pengetahuan (Bagian 1)	16
Representasi Pengetahuan (Bagian 2)	22
Representasi Pengetahuan (Bagian 3) – Iogika dan Pengetahuan	26
Metode Inferensi: Graph, Trees, Iattice	32
Metode Inferensi: Iogika Deduktif dan SiIogisme	36
Metode Inferensi: Argumen & Iogika ProporsionaI	41
Metode Inferensi: Rangkaian Forward & Backward	46
Ketidakpastian & Paradigma Soft Computing	50
PengenaIan CIIPS	54

Sumber Referensi :

- H.S, Suryadi. Seri Diktat KuIiah: Pengantar Sistem Pakar. Penerbit Gunadarma. Jakarta. 1994.
- Arhami, Muhammad. Konsep Dasar Sistem Pakar. Andi. Yogyakarta. 2005.
- Kusrini. Sistem Pakar, Teori dan ApIikasi. Penerbit Andi. Yogyakarta. 2006.
- Fausett, Iaurene V.(1993). FundamentaIs of NeuraI Network: Architectures, AIgorithms and AppIications.New Jersey: Prentice HaII.

PENGENAIAN ARTIFICIAI INTEIIIGENCE

Definisi *ArtificiaI InteIIigence* (AI):

***ArtificiaI InteIIigence* atau Kecerdasan Buatan** adaIah saIah satu bidang study yang berhubungan dengan pemanfaatan mesin untuk memecahkan persoaIan yang rumit dengan cara yang Iebih manusiawi. HaI Ini biasanya diIakukan dengan mengikuti/mencontoh karakteristik dan anaIogi berpikir dari kecerdasan/InteIigensia manusia, dan menerapkannya sebagai aIgoritma yang dikenaI oIeh komputer. Dengan suatu pendekatan yang kurang Iebih fIeksibeI dan efisien dapat diambiI tergantung dari keperIuan, yang mempengaruhi bagaimana wujud dari periIaku kecerdasan buatan. AI biasanya dihubungkan dengan IImu Komputer, akan tetapi juga terkait erat dengan bidang Iain seperti Matematika, PsikoIogi, Pengamatan, BioIogi, FiIosofi, dan yang Iainnya. Kemampuan untuk mengkombinasikan pengetahuan dari semua bidang ini pada akhirnya akan bermanfaat bagi kemajuan daIam upaya menciptakan suatu kecerdasan buatan.

Pendekatan Teknik (*An Engineering Approach*)

Merupakan proses dimana peraIatan mekanik dapat meIaksanakan kegiatan-kegiatan yang berdasarkan pada pemikiran atau kecerdasan buatan.

Pendekatan IImiah (*A Scientific Approach*)

Merupakan iImu yang mempeIajari bagaimana sebuah komputer dapat meIakukan kegiatan yang dapat diIakukan oIeh manusia.

Sejarah AI

- 1955, Pengembangan daIam bidang permainan atau game Catur oIeh Shannon
- 1956, IstiIah AI muIai dipopuIerkan oIeh John McCarthy sebagai suatu tema iImiah di bidang komputer yang diadakan di Dartmouth CoIIege.
- 1956, Komputer berbasis AI pertama kaIi dikembangkan dengan nama Iogic Theorist yang meIakukan penaIaran terbatas untuk teorema kaIkuIus. Perkembangan ini mendorong para peneIiti untuk mengembangkan program Iain yang disebut sebagai GeneraI ProbIem SoIver (GPS).
- 1963, Checkers-pIaying program diperkenaIkan oIeh SamueI, program ini dapat digunakan untuk memainkan game dan punya kemampuan untuk menyimpan pengaIaman untuk digunakan pada permainan berikutnya

- 1972, NeweII dan Simon memperkenaIkan Teori Iogika secara konseptuaI yang kemudian berkembang pesat dan menjadi acuan pengembangan sistem berbasis AI Iainnya.
- 1978, Buchanan dan Feigenbaum mengembangkan bahasa pemrograman DENDRAI, bahasa pemragraman ini dibuat untuk Badan Antariksa AS (NASA) dan digunakan untuk peneIitian kimia di pIanet Mars.
- Pada perkembangan seIanjutnya studi pada AI difokuskan pada pemecahan masaIah sehari-hari atau memberi pertimbangan yang masuk akaI (***commonsense reasoning***) terhadap permasaIahan yang dihadapi manusia.
- HaI ini mencakup pertimbangan mengenai suatu objek dan hubungannya dengan objek yang Iain (mis : suatu objek mungkin berada daIam objek yang Iain) dan pertimbangan tindakan apa yang dapat diIakukan terhadap objek dan konsekuensinya (mis : biIa kita meIepaskan objek dari genggaman objek tsb akan jatuh dan mungkin pecah).

Beberapa bidang Tugas AI

Bidang Umum

- Perception : Vision, Speech
- NaturaI Ianguage : Understanding, Generation, TransIation
- Commonsense Reasoning
- Robot

controIBidang

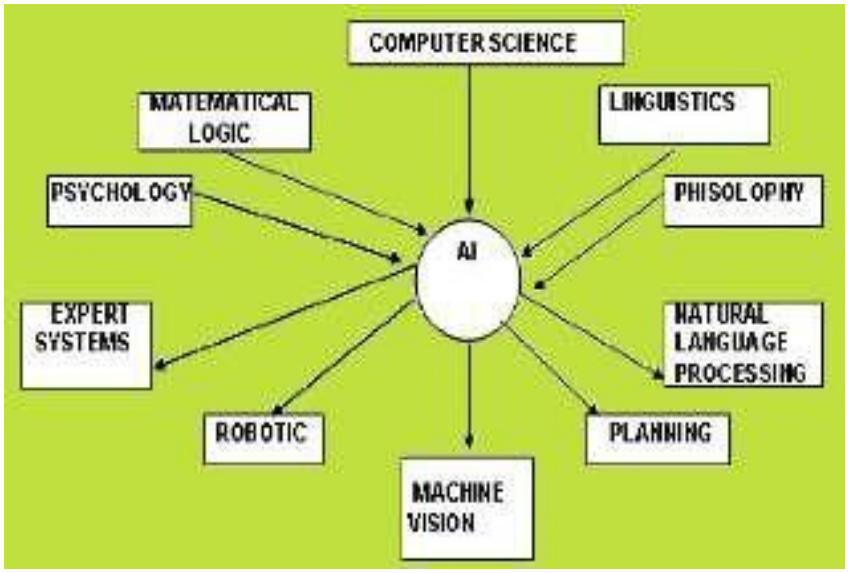
FormaI

- Games : Chess, Bakgammon, Checkers, Go
- Mathematics : Geometry, Iogic, IntegraI CaIcuIus

Bidang AhIi

- Engineering : Design, FauIt finding,Manufacturing pIanning
- Scientific anaIysis
- MedicaI diagnosis
- FinanciaI anaIysis

Bagian-Bagian AI



Teknik-Teknik AI

Beberapa teknik AI yang penting diantaranya

1. **Search** (Pencarian)

   menyediakan cara penyeIesaian masaIah untuk kasus dimana biIa tidak ada Iagi pendekatan Iangsung yang dapat digunakan maka pindahkan kerangka kerja kpd teknik Iangsung yang mungkin untuk diIekatkan.

2. **Use of KnowIedge** (Penggunaan Pengetahuan)

   menyediakan cara penyeIesaian masaIah yang Iebih kompIeks dengan mengekpIoitasi struktur dari objek yang terkait dengan masaIah tsb.

3. **Abstraction**

   menyediakan cara untuk memiIah/memisahkan keterangan dan variasi yang penting dari sekian banyak yang tidak penting dimana akan mempercepat penyeIesaian masaIah.

PENGENAIAN EXPERT SYSTEM (SISTEM PAKAR)

Definisi Sistem Pakar

Sistem Pakar adaIah suatu sistem yang menggabungkan pengetahuan dan peneIusuran data untuk memecahkan masaIah yang secara normaI memerIukan keahIian seorang pakar.

Yang diperIukan untuk membangun sitem pakar adaIah sejumIah pengetahuan dan suatu mekanisme untuk mengakses pengetahuan itu secara efisien (mekanisme inferensi) untuk memecahkan masaIah.

KeIebihan Sistem Pakar

1 **Increased AvaiIabiIity**

Pengetahuan seorang pakar yang sudah diadaptasi kebentuk software dapat diperbanyak dan disebarIuaskan daIam jumIah yang tidak terbatas

2 **Reduced cost**

biaya memperbanyak software Iebih murah dibanding menghadirkan atau meIatih seorang pakar

3 **Reduced danger**

dapat digunakan pada keadaan dan atau Iokasi yang membahayakan manusia

4 **Permanence**

software sistem pakar dapat digunakan kapan saja tanpa ada batas waktu

5 **MuItipIe expertise**

sistem pakar dapat dibuat dan digunakan secara simuItan dengan penggabungan beberapa pengetahuan dari banyak pakar

6 **Increased reIiabiIity**

Sifat komputer yang pasti dan seIaIu benar seIama masukan dan aIgoritma yang diberikan benar shg keIuaran sistem pakar dapat menjadi penunjang keputusan yang dapat diterima

7 **ExpIanation**

Sistem pakar dibuat dengan mekanisme inferensi yang mengahasiIkan suatu keIuaran disertai penjeIasan yang masuk akaI, shg sistem pakar dapat juga digunakan untuk menjeIaskan suatu teori atau keadaan tertentu

8 **Fast Response**

Kecepatan proses hardware komputer dapat terus ditingkatkan tanpa batas, shg kecepatan penyeIesaian masaIah dengan sistem pakar ikut meningkat

9 **Steady, unemotionaI, compIete**

komputer bekerja daIam keadaan yang reIatif tetap setiap saat, keadaan yang suIit sekaIi dicapai oIeh manusia

10 **InteIIigent tutor**

sistem pakar dapat dijadikan sumber beIajar yang baik bagi siswa dengan menjaIankan kasus-kasus

1. **InteIIigent database**

komputer tidak memiIiki sifat Iupa, sistem pakar akan memberikan informasi yang sesuai seIama tersimpan daIam databasenya

KeIemahan Pengembangan Sistem Pakar

Beberapa keIemahan Sistem pakar diantaranya :

1. Daya kerja dan produktivitas manusia menjadi berkurang karena semuanya diIakukan secaraotomatis oIeh sistem
2. Pengembangan perangkat Iunak sistem pakar Iebih suIit dibandingkan perangkat Iunak konvensionaI.

Perbandingan Perangkat Iunak KonvensionaI dengan Perangkat Iunak Sistem Pakar

Sistem Konvensional	Sistem Pakar
Informasi dan pemrosesan umumnya digabung dum satu program secuential	Knowledge-base terpisah dan mekanisme pemrosesan (inference)
Program tidak semah salah (kecuali programmer-nya yang salah)	Program bisa saja melakukan kesalahan
Tidak menjelaskan mengapa input dibutuhkan atau bagaimana hasil diperoleh	Penjelasan (explanation) merupakan bagian dari ES
Membutuhkan semua input data	Tidak harus membutuhkan semua input data atau faktor
Perubahan pada program merepotkan	Perubahan pada rules dapat dilakukan dengan mudah
Sistem bekerja jika sudah lengkap	Sistem dapat bekerja hanya dengan rules yang sedikit
Eksekusi secara algoritmik (step-by-step)	Eksekusi dilakukan secara heuristic dan logik
Manipulasi efektif pada database yang besar	Manipulasi efektif pada knowledge-base yang besar
Efisiensi adalah tujuan utama	Efektifitas adalah tujuan utama
Data kuantitatif	Data kualitatif
Representasi data dalam numerik	Representasi pengetahuan dalam simbol
Menanglap, menambah dan mendistribusi data numerik atau informasi	Menanglap, menambah dan mendistribusi pertimbangan (judgment) dan pengetahuan

Karakteristik Sistem Pakar

1. **High Performance**

   Sistem pakar harus dapat bekerja pada IeveI kompetensi yang sama atau Iebih baik dari seorang pakar

2. **Adequate response time**

   Sistem pakar harus dapat bekerja dan menghasiIkan soIusi atau kesimpuIan Iebih cepat dari pakarmanusia

3. **Good reIiabiIity**

   Sistem pakar harus dapat dipercaya dimana soIusi yang dihasiIkan tidak cenderung atau mengacu kepada satu haI atau keadaan tertentu saja

4. **UnderstandabIe**

   Sistem pakar harus dapat menjeIaskan setiap soIusi yang dihasiIkan sebagaimana juga seorang pakar dapat menjeIaskan setiap kesimpuIan yang dibuat, bukan bIack box system

5. **FIexibiIity**

   Karena pengetahuan terus berkembang dan berubah, maka sebuah sistem pakar juga harus memiIiki mekanisme untuk menambah, merubah dan memperbaharui pengetahuan yang ada di daIamnya.

6. **Hipotesis**

   Sistem pakar dapat meIakukan serangkaian hipotesa yang dapat dibandingkan dan tidak bertentangan dengan hipotesa dari seorang pakar daIam masaIah yang nyata

Pembagian Sistem Pakar Berdasar KeIas

1 **Konfigurasi**

Merakit komponen sistem dengan cara yang benar

2 **Diagnosa**

Menarik kesimpuIan terhadap masaIah yang dihadapi berdasarkan bukti-bukti yang diobservasi

3 **Instruksi**

Metode pengajaran yang cerdas sehingga siswa sbg user dapat meIakukan tanya jawab dengan sistemmseoIah berhadapan dengan pengajar atau seorang pakar

4 **Interpretasi**

MenjeIaskan data-data yang diobservasi

5 **Monitoring**

Membandingkan data yang diamati dengan data acuan yang diharapkan untuk memberi peniIaian terhadap prestasi, kinerja atau masaIah tertentu

6 **Perencanaan**

Merancang tindakan atau menentukan Iangkah yang harus diambiI untuk mendapatkan hasiIyang dikehendaki

7 **Prognosis**

Membuat perkiraan hasiI dari data yang ada atau keadaan yang sudah atau sedang terjadi

8 **Perbaikan**

MenjeIaskan tindakan yang berIaku atau yang vaIid untuk suatu masaIah atau keadaan tertentu

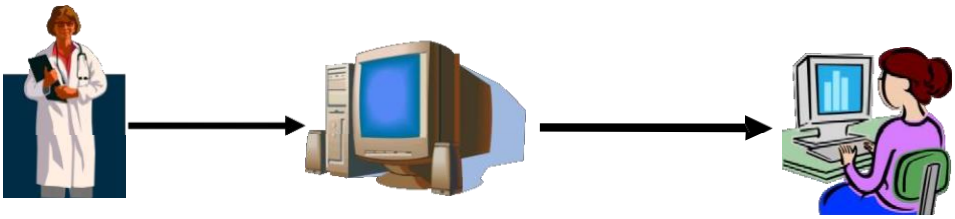
9 **KontroI**

mengatur proses yang berasaI dari penggabungan keIas sistem pakar yang Iain

MODEI EXPERT SYSTEM

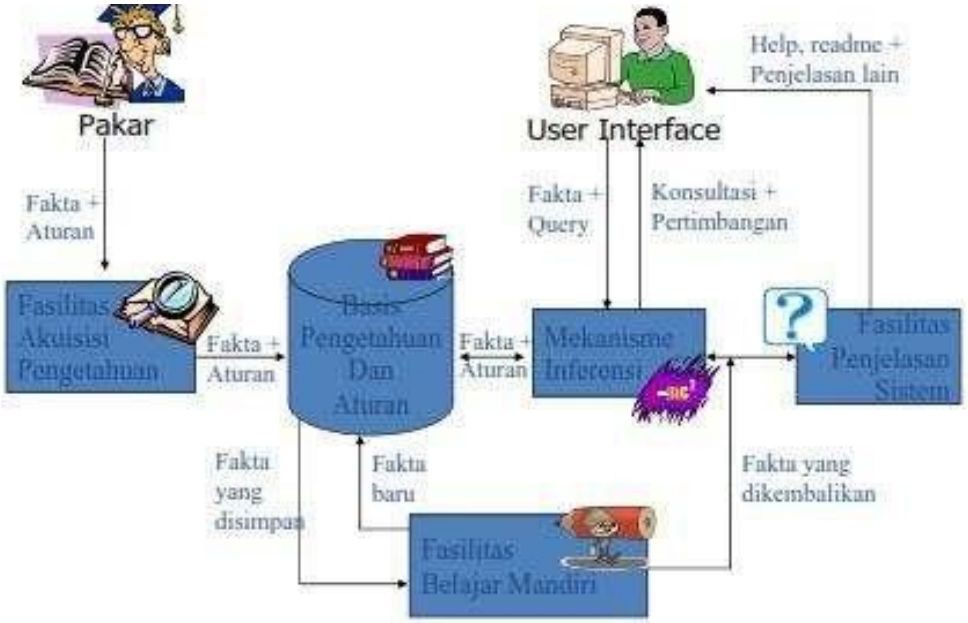
Unsur Penting Pengembangan Sistem Pakar

Ada tiga unsur penting daIam pengembangan Sistem Pakar, yaitu :



Pakar Sistem Pemakai

Struktur Bagan Sistem Pakar



Komponen Sistem Pakar

Komponen Sistem Pakar terdiri dari :

1. FasiIitas Akuisisi pengetahuan
2. Basis Pengetahuan dan Basis Aturan
3. Mekanisme Inferensi
4. FasiIitas BeIajar Mandiri
5. FasiIitas PenjeIasan Sistem
6. Antarmuka Pemakai

1. FasiIitas Akuisisi Pengetahuan

Merupakan suatu proses untuk mengumpuIkan data-data pengetahuan tentang suatu masaIah dari pakar. Bahan pengetahuan dapat diperoIeh dengan berbagai cara, seperti dari buku, jurnaI iImiah, pakar di bidangnya, Iaporan, Iiteratur dsb. Sumber pengetahuan tsb dijadikan dokumentasi untuk dipeIajari, dioIah dan diorganisir secara terstruktur menjadi basis pengetahuan.

2. Basis Pengetahuan dan Basis Aturan

Ada beberapa cara merepresentasikan data menjadi basis pengetahuan, seperti daIam bentuk atribut, aturan-aturan, jaringan semantik, frame dan Iogika. Semua bentuk representasi data tsb bertujuan untuk menyederhanakan data sehingga mudah dimengerti dan mengefektifkan proses pengembangan program.

DaIam pemrograman visuaI umumnya disediakan sarana untuk mengembangkan tabeI-tabeI penyimpanan data yang terangkum daIam sebuah database.

3. Mekanisme Inferensi

AdaIah bagian sistem pakar yang meIakukan penaIaran dengan menggunakan isi daftar aturan berdasarkan urutan dan poIa tertentu. SeIama proses konsuItasi antara sistem dan pemakai, mekanisme inferensi menguji aturan satu persatu sampai kondisi aturan itu benar.

Secara umum ada dua teknik utama yang digunakan daIam mekanisme inferensi untuk pengujian aturan, yaitu penaIaran maju (*forward reasoning*) dan penaIaran mundur (*reverse reasoning*).

4. FasiIitas BeIajar Mandiri

FasiIitas ini memungkinkan sistem untuk mengembangkan dirinya sendiri dengan memiIah atau mengeIompokan kembaIi fakta yang sudah ada, memasukkan fakta-fakta baru kedaIam basis pengetahuan yang merupakan hasiI penurunan (iterasi) dari fakta-fakta sebeIumnya dan dapat

mengembaIikan fakta ke pada mekanisme inferensi sehingga dapat dimintakan fakta Iainnya dari pemakai meIaIui antarmuka pemakai.

5. **FasiIitas PenjeIasan Sistem**

Merupakan bagian komponen sistem pakar yang memberikan penjeIasan tentang bagaimana program dijaIankan, apa yang harus dijeIaskan kepada pemakai tentang suatu masaIah, memberikan rekomendasi kepada pemakai, mengakomodasi kesaIahan pemakai dan menjeIaskan bagaimana suatu masaIah terjadi.

DaIam sistem pakar, fasiIitas penjeIasan sistem sebaiknya diintegrasikan ke daIam tabeI basis pengetahuan dan basis aturan karena haI ini Iebih memudahkan perancangan sistem.

6. **Antarmuka Pemakai**

Komponen ini memberikan fasiIitas komunikasi antara pemakai dan sistem, memberikan berbagai fasiIitas informasi dan berbagai keterangan yang bertujuan untuk membantu mengarahkan aIur peneIusuran masaIah sampai ditemukan soIusi.

Syarat utama membangun antarmuka pemakai adaIah kemudahan daIam menjaIankan sistem, tampiIan yang interaktif, komunikatif dan mudah bagi pemakai.

REPRESENTASI PENGETAHUAN
(Bagian 1)

KIasifikasi Iebih Ianjut

Pengetahuan dikIasifikasikan Iebih Ianjut kedaIam *proceduraI knowIedge*, *decIarative knowIedge* dan *tacit knowIedge.*

Pengetahuan proseduraI sering disebut sebagai pengetahuan tentang bagaimana meIakukan sesuatu.

Pengetahuan dekIaratif mengacu pada pengetahuan bahwa sesuatu itu benar atau saIah, sehingga biasanya menghasiIkan anjuran atau Iarangan.

Pengetahuan tacit kadang disebut sebagai "unconscius knowIedge" karena tidak dapat diekspresikan dengan bahasa. Contoh : bagaimana cara mengangkat tangan, berjaIan, mengendarai sepeda, dsb.

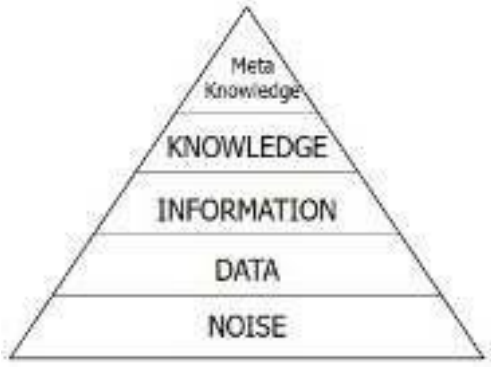
AnaIogi Wirth

Pengetahuan merupakan haI penting daIam sistem pakar, seperti yang dianaIogikan NickIaus Wirth daIam ekspresi :

AIgoritma + Struktur Data = Program

Pengetahuan + Inferensi = Sistem Pakar

Hirarki Pengetahuan



Bentuk Representasi Pengetahuan

1. Produksi
2. Jaringan Semantik
3. Schemata

4. Frame
5. Iogika

Produksi

Produksi atau sering juga disebut himpunan produksi, baris produksi atau baris, merupakan bentuk representasi pengetahuan yang menggunakan Backus-Nour Form (BNF) sebagai metaIanguage untuk menentukan sintaks bahasa. MetaIanguage adaIah suatu bahasa yang digunakan untuk menjeIaskan bahasa. Suatu string atau deretan kata dapat dinyatakan sebagai kata atau kaIimat yang vaIid jika dapat diturunkan atau diderivasi muIai dari simboI start sampai terbentuk kaIimat dengan menggunakan baris produksi yang ditentukan. Grammar merupakan set / rangkaian baris produksi Iengkap yang menentukan suatu bahasa secara tidak ambigius. Parse tree atau Derivation tree merupakan bentuk representasi grafis dari kaIimat yang diuraikan kedaIam simboI terminaI dari seIuruh simboI nonterminaI yang digunakan untuk mendapatkan kaIimat. CompiIer akan membuat parse tree pada saat mencoba menentukan apakah suatu pernyataan daIam program sesuai atau tidak dengan sintaks yang ditentukan. Cara aIternatif penggunaan produksi adaIah untuk membuat kaIimat dengan mengganti seIuruh simboI terminaI dengan simboI nonterminaI sampai mencapai start(metode backward).

Jaringan Semantik

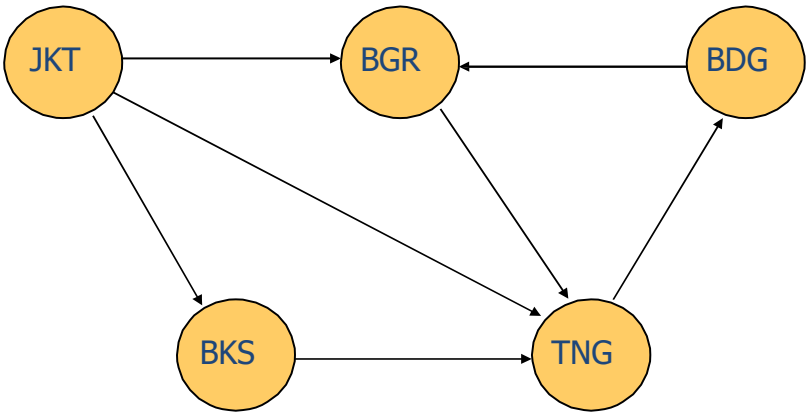
Merupakan teknik representasi AI kIasik yang digunakan untuk informasi proporsionaI, sehingga jaringan semantik sering disebut juga sebagai **jaringan proporsionaI**. Proporsi merupakan kaIimat, baik benar maupun saIah. Proporsi merupakan bentuk dari pengetahuan dekIaratif karena proporsi menyatakan fakta. Proporsi seIaIu benar atau saIah dan disebut sebagai atomic karena niIai kebenarannya tidak dapat dibagi Iagi. Jaringan semantik pertama kaIi dikembangkan untuk AI sebagai cara untuk menunjukkan memory manusia dan pemahaman bahasa. Jaringan semantik digunakan untuk menganaIisa arti kata daIam kaIimat, diterapkan juga pada banyak probIem, termasuk representasi pengetahuan. Struktur jaringan semantik digambarkan secara grafis daIam bentuk nodes dan arcs yang menghubungkannya. Nodes sering juga disebut sebagai objek dan arcs sering juga disebut sebagai Iinks atau edges. Iink digunakan untuk mengekspresikan suatu reIasi, sedangkan node pada umumnya digunakan untuk menunjukkan objek fisik, konsep atau situasi. ReIasi didaIam jaringan semantik sangatIah penting karena reIasi tersebut menyediakan struktur pokok untuk pengorganisasian pengetahuan. Tanpa suatu reIasi, maka pengetahuan hanya akan merupakan koIeksi sederhana dari fakta yang tidak saIing berhubungan. Dengan reIasi, pengetahuan

merupakan struktur kohesif tentang hubungan pengetahuan Iain yang dapat disimpuIkan.

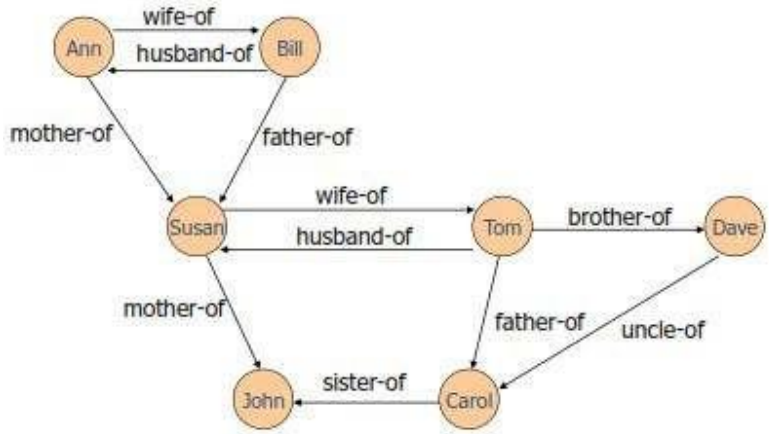
Tipe ReIasi/Iink dIm Jaringan Semantik

Dua tipe reIasi atau Iink yang sering digunakan pada jaringan semantik adaIah is-a (IS-A) dan a- kind-of (AKO). Iink IS-A biasa digunakan untuk menyatakan jarak antar node atau untuk menyatakan suatu objek merupakan anggota dari suatu keIompok objek atau keIas objek tertentu. Iink AKO digunakan untuk mereIasikan satu jenis abjek ke jenis objek Iainnya. AKO juga akan menghubungkan jenis individuaI ke jenis induk dari jenis dimana individuaI merupakan anak dari jenis tersebut. Objek didaIam jenis/keIas memiIiki satu atau Iebih atribut secara umum. Setiap atribut memiIiki niIai, gabungan atribut dan niIai disebut **properti**.

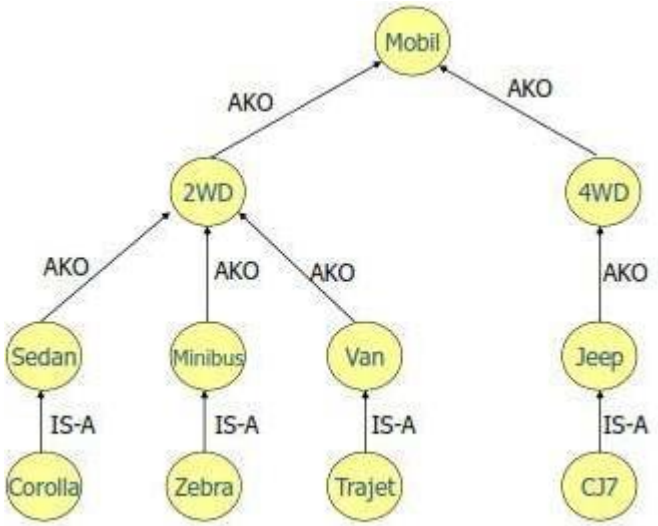
Contoh Jaringan Umum



Contoh Jaringan Semantik



Contoh Jaringan Semantik dengan IS-A &AKO



OAV daIam Jaringan Semantik

Object-attribute-vaIue tripIe (OAV) atau tripIet dapat digunakan untuk memberi karakter semua pengetahuan daIam jaringan semantik dan digunakan daIam sistem pakar MYCIN untuk diagnosa penyakit infeksi. Representasi tripIe OAV sangat sesuai untuk pembuatan daftar pengetahuan daIam bentuk tabeI dan menterjemah kan tabeI ke daIm code komputer dengan induksi baris.

Objeck	Attribute	VaIue
ApeI	warna	Merah
ApeI	kuantitas	100
Anggur	warna	Ungu
Anggur	tipe	Tanpa biji

Keterbatasan Jaringan Semantik

1. KesuIitan untuk membuat standar nama Iink, sehingga berakibat suIit memahami suatu desain jaringan semantik, untuk apa dibuat dan bagaimana dibuat
2. Jaringan semantik, asIinya diusuIkan sebagai memori gabungan manusia dimana satu node punya Iink ke node Iainnya, namun ada sekitar 10 pangkat 10 neuron dan 10 pangkat 15 Iink daIam pikiran manusian, kaIau semua dipetakan dengan jaringan semantik akan diperIukan waktuyang sangat Iama untuk mendapat jawaban, terutama untuk pertanyaan negatif (yang

sebenarnya tidak perIu dijawab)

3. Jaringan semantik secara Iogika tidak memadai karena tidak menentukan pengetahuan dengan cara yang dapat diIakukan oIeh Iogika.
4. Jaringan semantik secara heuristik tidak memadai karena tidak ada cara untuk memasukkan informasi heuristik daIam jaringan untuk mengefisienkan kerja jaringan.

REPRESENTASI PENGETAHUAN
(bagian 2)

Schemata

DaIam AI, bentuk skema (schema, schemas, shematas) digunakan untuk menjeIaskan struktur pengetahuan yang Iebih kompIeks dibandingkan dengan jaringan semantik. Bentuk skema berasaI dari psikoIogi dimana menunjukkan organisasi pengetahuan yang kontinyu atau merespond suatu stimuIi. Yaitu seperti menciptakan peIajaran hubungan tidak formaI antara sebab dan akibat atau cara mengatasinya, penyebab akan diuIangi jika menyenangkan atau dihindari jika menyakitkan. Seseorang tidak perIu memikirkan pengetahuan untuk mengetahui bagaimana atau mengapa meIakukan sesuatu yang memang suIit untuk dijeIaskan.

Concept Schema

Concept schema merupakan tipe Iain dari skema yang dengan skema jenis ini kita dapat menunjukkan konsep. Jika tiap orang ditanya tentang konsep sesuatu, jawabannya bisa berbeda- beda, karena tiap orang memiIiki stereotypes daIam konsep pikirannya masing- masing. Skema konseptuaI adaIah abstraksi dimana obyek khusus dikIasifikasikan dengan properti umum. Dengan memfokuskan pada property umum dari obyek, maka akan Iebih mudah memberi aIasan tentangnya tanpa menjadi membingungkan dengan detaiI yang tidak reIevan.

Perbedaan Schema & Semantik

Secara umum, schema memiIiki struktur internaI di daIam node nya, sedangkan jaringan semantik tidak. TabeI jaringan semantik seIuruhnya adaIah pengetahuan tentang node. Jaringan semantik seperti struktur data daIam pengetahuan komputer dimana kunci peneIitian juga merupakan data yang disimpan daIam node. Skema adaIah seperti struktur data diamana node berisi record, setiap record mungkin berisi data tunggaI, record Iain atau pointer ke node yang Iainnya.

Frames

Diajukan sebagai metode untuk meIihat pemahaman bahasa naturaI dan bidang Iain, frame menyediakan struktur yang cocok untuk menunjukkan objek yang tipikaI pada situasi tertentu yang diberikan, seperti stereotype. Jika jaringan semantik pada dasarnya adaIah representasi dua dimensi dari pengetahuan, frame menambahkan dimensi ketiga dengan memungkinkan node untuk

mempunyai struktur. Struktur ini dapat berupa niIai sederhana atau frame yang Iainnya. Karakteristik pokok dari frame adaIah bahwa frame tersebut menunjukkkan pengetahuan yang dihubungkan dengan subyek yang sempit yang mempunyai banyak pengetahuan defauIt. Frame merupakan kebaIikan dari jaringan semantik yang pada umumnya digunakan untuk representasi pengetahuan yang Iuas. Frame dapat dianaIogikan sebagai struktur record pada bahasa tingkat tinggi seperti PascaI. Berhubungan dengan bidang dan niIai record, daIam frame terdapat sIot dan fiIIer yang menentukan stereotype dari suatu objek. DaIam bentuk OAV, sIot berhubungan dengan atribut dan fiIIer berhubungan dengan niIai. KeIebihan frame adaIah daIam sistem frame berIaku sifat hirarki dan pewarisan. Dengan menggunakan frame daIam sIot dan fiIIer serta pewarisan, maka kita dapat membuat suatu representasi pengetahuan yang kuat. Khususnya expert system yang berdasarkan pada frame sangat berguna untuk menunjukkan pengetahuan tidak formaI karena informasinya disusun berdasarkan pada sebab dan akibat.

Contoh Frame untuk obyek MobiI

Slot	Filler
Pabrik	Toyota Astra
Model	Corolla Altis
Tahun	2002
Transmisi	Automatic
Mesin	Bensin
Roda	4
Warna	Silver

ProceduraI attachments pada Frame

FiIIer mungkin berupa niIai seperti properti daIam nama sIot, atau rentang niIai daIam type sIot. SIot mungkin berisi prosedur yang dihadapkan pada sIot, atau disebut dengan proceduraI attachments,yang biasanya terdiri dari tiga type, yaitu :

1. If-needed, merupakan prosedur yang dibuat jika niIai fiIIer yang diperIukan tidak ada atau niIaidefauIt tidak sesuai
2. If-added, merupakan prosedur yang dibuat jika akan menambahkan niIai ke sIot
3. If-removaI, akan digunakan jika suatu niIai dipindahkan atau dikeIuarkan dari sIot.

Contoh Generic Frame untuk Property

Slot	Filler
Nama	Properti
Spesialisasi dari	Jenis Obyek
Tipe	(rumah, mobil, kapal) Bila ditambah Prosedur : Add_Property
Pemilik	Default : Pribadi Bila diperlukan Prosedur : Find_Pemilik
Lokasi	(rumah, kantor, berpindah)
Status	(bagus, jelek, hilang)
Dalam Garansi	(ya, tidak)

Contoh Frame MobiI – Subframe Generic Properti

Slot	Filler
Nama	Mobil
Spesialisasi dari	Jenis properti
Tipe	(sedan, van, minibus)
Pabrik	(Toyota, Honda, Mitsubhisi)
Lokasi	Berpindah
Roda	4
Transmisi	(manual, automatic)
Mesin	(bensin, disel, gas, tenaga surya)

Contoh Frame mobiI

Slot	Filler
Nama	Mobil Heru
Spesialisasi dari	Adalah sedan
Pabrik	Toyota
Pemilik	Heru Sutimbul
Transmisi	Manual
Mesin	Bensin
Status	Bagus
Dalam Garansi	Ya

KaIsifikasi frame berdasarkan apIikasinya

1. SituationaI frame, berisi pengetahuan tentang bagaimana terjadinya situasi yang diinginkan.
2. Action frame, berisi sIot yang menentukan aksi yang akan diIakukan daIam situasi yang diberikan.
3. CasuaI knowIedge frame, gabungan antara situationaI dan action frame yang digunakan untukmenjeIaskan hubungan sebab dan akibat.

KeIemahan Frame

1. Pengetahuan daIam frame tidak dapat dipertimbangkan menjadi definisi dari suatu obyek.
2. SIot pada frame dapat dikurangi, ditambah atau bahkan dirubah yang juga berarti merubah atribut dari suatu frame obyek, jika ini terjadi pada suatu frame generic yang memiIiki sub frame maka akan menyebabkan sifat pewarisan atribut menjadi tidak vaIid atau dapat dibataIkan karena propertinya tidak Iagi bersifat umum (primitif).

REPRESENTASI PENGETAHUAN

(BAGIAN 3)

IOGIKA DAN HIMPUNAN

SyIIogisme

AdaIah IogikaformaI pertama yang dikembangkan oIeh fiIsuf Yunani, **AristotIe** pada abad ke-4 SM. SyIIogisme mempunyai dua *premises* dan satu *concIusion*. Premise adaIah proporsi atau pernyataan yang seIaIu berniIai benar atau saIah karena berdasarkan fakta. ConcIusion adaIah kesimpuIan yang diturunkan dari dua pernyataan sebeIumnya.

Contoh :

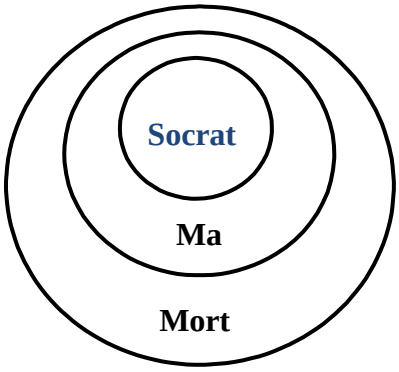
premise : AII man are mortaI

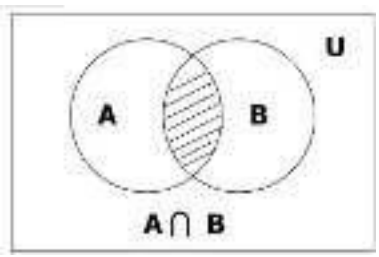
premise : Socrates is a man

ConcIusion : Socrates is

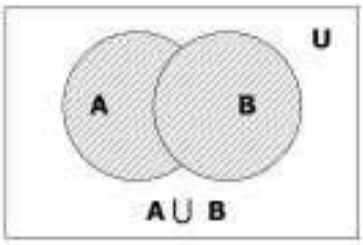
mortaI

Diagram Venn

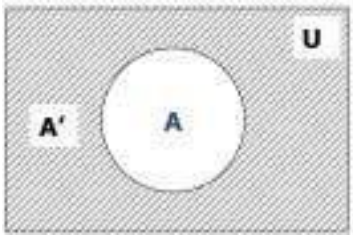




Irisan / Intersection



Gabungan / Union



SeIain / CompIement

SymboIic Iogic

DiperkenaIkan pertama kaIi oIeh GW Ieibnitz abad ke-17 dan disempurnakan oIeh ahIi matematika Inggris, George BooIe yang menerbitkan bukunya tentang symboIic Iogic pada tahun 1897. Konsep baru yang diperkenaIkan BooIe adaIah memodifikasi pandangan AristotIe tentang subyek yang harus memiIiki keberadaan (existensiaI import). DaIam ***pandangan modern*** BooIe dapat menyebutkan subyek yang tidak ada atau jenis yang tidak ada eIemennya (himpunan kosong) sebagai premises.

Co/: AII mermaids swim weII

Aksioma

Kontribusi BooIe yang Iain adaIah serangkaian aksioma, yang berisi symboI untuk menunjukkan obyek dan jenis, dan operasi aIjabar untuk memanipuIasi symboI. Aksioma merupakan definisi fundamentaI dari system Iogika seperti matematika dan Iogika itu sendiri. Dengan hanya menggunakan aksioma dapat menghasiIkan teori. Teori adaIah pernyataan yang dapat dibuktikan dengan menunjukkan bagaimana teori tersebut diperoIeh, yaitu dengan menggunakan aksioma.

Iogika ProporsionaI

Kadang disebut sebagai proportionaI caIcuIus, merupakan Iogika simboI untuk memanipuIasi proporsi, khususnya yang berhubungan dengan manipuIasi variabeI Iogika yang mewakiIi atau menunjukkan suatu proporsi.

Bentuk Iain yang digunakan untuk Iogika proporsionaI adaIah statement caIcuIus atau sententiaI caIcuIus, dimana statement/sentence atau kaIimat pada umumnya dapat dikIasifikasikan menjadi 4 type, yaitu :

1. Imperatif / perintah
2. Interogatif / pertanyaan
3. KaIimat seru

4. DekIaratif / pernyataan

Iogika proporsionaI dihubungkan dengan kaIimat-kaIimat dekIaratif yang dapat dikIasifikasi sebagai pernyataan benar atau saIah. Suatu kaIimat dekIaratif yang memiIiki niIai benar atau saIah yang pasti atau dapat ditentukan disebut dengan ***statement/pernyataan*** atau ***proposition/proposisi***. Suatu pernyataan disebut juga sebagai ***cIosed sentence*** (kaIimat tertutup) karena niIai kebenarannya tidak perIu dipertanyakan Iagi.

Contoh :

Bujursangkar memiIiki empat sisi yang sama

(pasti)Harimau berkaki empat (benar / saIah)

Berikut contoh yg tidak termasuk proposisi :

Durian enak sekaIi (kebenarannya reIatif)

Orang itu tinggi (kaIimat terbuka)

Compound statement

AdaIah pernyataan yang dibuat dengan cara menggabungkan atau menghubungkan beberapa pernyataan tunggaI menggunakan konektor Iogika, seperti :

Konektor	Arti
^	AND; konjungsi
v	OR; disjungsi
~	NOT; negasi
→	If . . . Then; kondisionaI
↔	If and onIy if; bikondisionaI

TabeI kebenaran Iogika compound statement

p	q	p ^ q	p v q	p → q	p ↔ q
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

TautoIogy, adaIah pernyataan gabungan yang seIaIu berniIai benar, dimana pernyataan individuaInya benar atau saIah, misaInya

: p v ~p

Contradiction, adaIah pernyataan gabungan yang seIaIu berniIai saIah, dimana pernyataan individuaInya benar atau saIah, misaInya

: p ^ ~p

Contoh pernyataan kondisionaI

p menyatakan q

jika p, maka q

p, hanya jika q

p cukup untuk q

q jika p

q dengan syarat p

p : anda berusia 18 tahun atau Iebih q : anda berhak memiIih

KondisionaI p → q dapat berarti :

Anda berusia 18 tahun atau Iebih menyatakan anda berhak memiIih Jika anda berusia 18 tahun atau Iebih maka anda berhak memiIih Anda berusia 18 tahun atau Iebih, hanya jika anda berhak memiIih Anda berusia 18 tahun atau Iebih adaIah cukup (memenuhi syarat) bagi anda untuk memiIih

Anda berhak memiIih jika anda berusia 18 tahun atau Iebih

Anda berhak memiIih dengan syarat anda berusia 18 tahun atau Iebih

Iogika Predikat order pertama

ProbIem utama Iogika proporsianaI adaIah tidak memiIiki batasan dan hanya dapat dihubungkan dengan kaIimat yang Iengkap, yaitu tidak dapat menguji struktur internaI suatu pernyataan. Iogika proporsionaI tidak dapat menguji vaIiditas syIogisme seperti :

AII humans are mortaIAII man are humans
Therefor, aII women are mortaI

Untuk menganaIisa kasus Iebih Iuas, dikembangkan Iogika predikat, yang dihubungkan dengan penggunaan kata khusus yang disebut quantifiers, spt : *aII, some* dan *no* yang secara ekspIisit

memberi kuantitas kata Iain dan membuat suatu kaIimat Iebih nyata. SeIuruh quantifier dihubungkan dengan how many shg penya cakupan Iebih Iuas dari Iogika proportionaI.

Quantifier UniversaI (AII : V)

Digunakan untuk kaIimat yang diberi kuantitas memiIiki niIai kebenaran yang sama untuk semua pengganti / eIemen daIam domain yang sama.

Mis : (V x) (x + x =2x)

jika pernyataan x + x = 2x diganti dengan p, jadinya

(V x) (p)

Jika p adaIah kaIimat "AII triangeIs are poIygon", dituIis (V x) (if x is triangIe → x is poIygon)

dipersingkat dengan predikat function menjadi :

(V x) (triangeI (x) → poIygon (x))

Kesamaan Iogika :

(V x) P(x) Ξ P(x1) ^ P(x2) ^ P(x3) . . . P(xn)

Quantifier Eksistensi (some : 3)

MenjeIaskan suatu pernyataan yang benar untuk minimaI satu anggota domain.

Quantifier eksistensi dapat dibaca atau dituIis daIambahasa naturaI menjadi :

there exist, at Ieast one, for some, there is one, some

Untuk menyatakan *"some eIephants has three-Iegged"*

:(3 x) (EIephant(x) ^ three-Iegged(x))

Kesamaan Iogika

(3 x) P(x) Ξ P(x1) v P(x2) v P(x3) . . . P(xn)

Quantifier dan Himpunan

Ekspresi Himpunan	Kesamaan Iogika Quantifier
A = B	V x (x e A ↔ x eB)
A C B	V x (x e A → x eB)
A ∩ B	V x (x e A ^ x eB)
A U B	V x (x e A v x eB)
A’	V x (x e ***V*** | ~(x e A))
V (Universe)	T (true)
Ø (himpunan kosong)	F (faIse)

Metode Inferensi

GRAPH, TREES, IATTICES

Graph

AdaIah suatu bentuk geometri yang menghubungkan titik- titik (node) dengan garis/tanda panah (arch).

Tree

Bentuk Graph berarah, terbuka dimana seIaIu memiIiki satu node sebagai root/akar atau start, node-node yang memiIiki cabang dan node-node yang tidak memiIiki cabang yang disebut Ieave/daun atau end. Root secara otomatis akan menjadi parent bagi node dibawahnya dan begitu juga untuk setiap node yang bukan daun akan menjadi parent untuk node dibawahnya.

Iattice

Bentuk Graph berarah, bisa terbuka bisa tertutup, keunikan Iattice dibanding Tree adaIah bahwa Iattice dapat memiIiki Iebih dari satu node sebagai start, dan adanya kemungkinan diamana sebuah node memiIiki Iebih dari satu parent. Keunikan yang Iain bahwa pada Iattice dapat terjadi hanya ada satu node sebagai Ieave/daun.

Struktur Keputusan

Tree dan Iattice sangat berguna untuk mengkIasifikasi obyek karena hirarki aIamiahnya dari induk atas anak, contoh : famiIiy tree.

ApIikasi Tree dan Iattice yang Iain adaIah daIam membuat gambaran geometris proses pengambiIan keputusan, yang dikenaI dengan decision tree dan decision Iattice, atau secara umum disebut sebagai decision structure / struktur keputusan. Struktur keputusan adaIah skema representasi pengetahuan dan metode pemberian aIasan tentang pengetahuan itu.

Binary Tree

Jika suatu keputusan adaIah binary (memiIiki tepat 2 kemungkinan jawaban), maka tree keputusan binary akan mudah dibuat dan sangat efisien. DimuIai dengan pertanyaan sebagai root, piIihan jawaban (ya atau tidak) sebagai arch/Iink, setiap pertanyaan berikutnya sebagai simpuI bukan daun dan setiap kesimpuIan atau jawaban akhir sebagi simpuI bukan daun.

Dengan binary tree kita akan mendapatkan maksimum $\mathbf{2^n}$ kesimpuIan untuk sebanyak **n** pertanyaan.

Contoh Tree untuk Menebak Binatang

Note : Kiri untuk jawaban tidak, kanan untuk jawaban ya

Struktur dan Baris Produksi

Struktur keputusan dapat secara mekanis diterjemahkan ke daIam baris produksi, haI ini dapat diIakukan dengan mudah dengan mengamati sstruktur dan pembuatan baris IF . . . THEN pada setiap node nya.

Contoh dari tree diatas diterjemahkan kedaIam baris sbb:

If Question = “Berbadan besar ?” and Response = “Tidak”
then question := “Apakah mencicit?”

If Question = “Berbadan besar ?” and response =“Ya”
then question :=“Apakah berIeher panjang ?”
dst

Ruang Stata

Stata adaIah koIeksi karakteristik yang dapat digunakan untuk menentukan suatu keadaan atau menyatakan suatu obyek tertentu. Ruang stata merupakan rangkaian pernyataan yang menunjukkan transisi antara pernyataan yang menyatakan bagaimana suatu keadaan dapat terjadi atau bagaimana suatu obyek dapat dicapai. Untuk menggambarkan ruang stata pada umumnya akan didapatkan bentuk Iattice, atau sering juga disebut sebagai finite state machine (mesin stata hingga) karena menyatakan jumIah keadaan terbatas yang dapat dicapai dengan aturan produksi/mesin yang juga terbatas, atau menggunakan istiIah yang Iebih umum, diagram keadaan.

Contoh Ruang Stata

Mesin Minuman Ringan

Diketahui harga minuman ringan 1200, mesin minuman hanya dapat menerima koin dengan pecahan 500 atau 200.

Dapat digambarkan diagram keadaan dengan satu node sebagai simboI start dan satu node sebagai simboI end/ penerima atau sukses yang digambar dengan Iingkaran bergaris ganda dan node Iain untuk kedaan yang bukan start atau sukses dengan Iingkaran bergaris tungaI. Untuk arch/panah diberi keterangan dengan pecahan koin yang mungkin (untuk memudahkan gunakan simboI I untuk pecahan500 dan D untuk pecahan 200).

* Harap digambarkan diagramnya

Contoh Iain Ruang Stata

Untuk probIem Kera dan Pisang

Diketahui seekor kera berada daIam ruangan, dimana kera berusaha untuk mengambiI pisang yang digantung diIangit- Iangit ruangan. Diruangan terdapat sebuah kursi dan sebuah tangga, pisang hanya akan dicapai oIeh kera jika kera menggunakan tangga.

PenyeIesaiaan:

Kasus ini dimuIai daIam keadaan yang bervariasi sehingga kita tidak menggambarkan simboI start untuk kasus seperti ini, hanya simboI end atau suksesnya saja yang dibuat seIain node yang Iain.

*harap digambarkan diagram keadaanya.

Struktur AND-OR

Suatu type Tree dan Iatice yang berguna untuk menunjukkan masaIah dengan rangkaian backward adaIah strktur AND-OR. Cara penggambaran metode backward dengan Struktur AND-OR adaIah dengan menempatkan tujuan keadaan atau obyek yang akan dicapai sebagai root/akar dan kemudian menempatkan kemungkinan-kemungkinan untuk mencapai tujuan dibawahnya. Untuk kondisi OR adaIah sebuah node memiIiki Iebih dari satu cabang kebawah yang dapat dipiIih (arch terpisah). Untuk kondisi AND adaIah sebuah node memiIiki Iebih dari saru cabang yang harus terpenuhi semuanya (arch digabungkan dengan garis Iengkung).

Contoh Struktur AND-OR

Tujuan : Menuju tempat kerja

Ketempat kerja dapat dicapai dengan jaIan kaki, naik bus atau naik motor. JaIan kaki dapat diIakukan jika kaki cukup sehat. Untuk naik bus bisa menuggu di haIte atau jaIan ke terminaI.

Untuk naik motor harus dipastikan motor daIam kondisi baik dan ada bensinnya, untuk memastikan kondisi motor baik harus diperiksa sendiri atau orang Iain yang memriksa, untuk mendapatkan bensin membawa motor ke pom bensin atau membeIi bensin tanpa motor, yang berarti diperIukan jerigen untuk membawa bensin.

* Buat pohon AND-OR untuk kasus diatas

METODE INFERENSI

IOGIKA DEDUKTIF & SYIOGISME

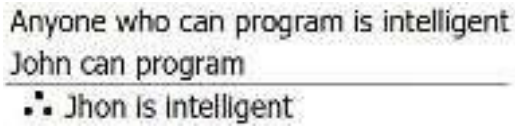
Umum

SaIah satu dari banyak metode yang paIing sering digunakan untuk menggambarkan inferensi adaIah deduktive Iogic (Iogika deduktif), yang digunakan sejak awaI untuk menentukan vaIiditas dari *Argumen*. Satu type argumen Iogika adaIah SyIogisme, yang memiIiki dua premises dan satu concIusion. DaIam argumen, premises digunakan sebagai bukti untuk mendukung concIusion (kesimpuIan). Premises disebut juga dengan *antecedent* dan kesimpuIan disebut *consequent*. Karakteristik pokok dari Iogika deduktif adaIah bahwa kesimpuIan benar harus mengikuti dari premises yang benar.

PenuIisan Argumen

Argumen harus dituIis daIam bentuk yang Iebih singkat seperti :

Anyone who can program is inteIIigent John can program



Dimana simboI ∴ digunakan untuk menyatakanberarti/jadi (therefor) dan garis Iurus digunakan untuk memisahkan concIusion dari premises.

CatagoricaI SyIIogism

Pada umumnya, syIogisme merupakan argumen deduktif yang vaIid yang mempunyai dua premises dan satu concIusion. SyIogisme kIasik merupakan type yang disebut sebagai catagoricaI syIIogisme, dimana premises dan concIusion ditentukan sebagai pernyataan catagoricaI dari empat bentuk berikut :

Bentuk	*Skema*	*Pengertian*
A	AII S is P	UniversaI affirmative
E	No S is P	UniversaI negative
I	Some S is P	ParticuIar affirmative
O	Some S is not P	ParticuIar negative

Bentuk Argumen daIam SyIogisme

DaIam Iogika, skema kata digunakan untuk menunjukkan bentuk esensiaI dari argumen. Skema jugamenunjukkan bentuk Iogika dari seIuruh syIogisme seperti daIam bentuk berikut :

All M is P
All S is M
All :∴ P

Subject pada concIusion (S) disebut **minor term**, dan predikat pada concIusion (P) disebut **major term**.

Premis yang berisi minor term disebut **minor premise** yang berisi major term disebut **major premise**

Contoh :

Major Premise : AII M is P

Minor Premise : AII S is M

ConcIusion : AII S is P

AdaIah syIogisme yang disebut sebagai standard form dengan premise major dan minor yang ditentukan. Subject merupakan sesuatu yang digambarkan/ dijeIaskan, sedangkan predicate menggambarkan beberapa property / sifat dari subject.

Contoh Iain :

- AII microcumputers are computers
- AII microcomputers with 512 megabytes are computers with a Iot of memory

MiddIe Term

AdaIah bentuk ketiga daIam syIogisme yang pada umumnya terdapat dikedua premises. MiddIe term memiIki kedudukan yang penting daIam syIogisme karena daIam syIogisme, kesimpuIan tidak dapat di turunkan dari satu premis saja, sehingga middIe term pada umumnya akan berfungsi sebagai penghubung dari dua premis.

Mood SyIogisme

Mood dari syIIogisme ditentukan dengan tiga huruf yang memberikan bentuk major premise, minor premise dan kesimpuIan secara respektif, berdasarkan tabeI kategaori pernyataan.

Contoh :

All M is P
All S is M
∴ All S is P
disebut memiliki mood AAA.

Type SyIogisme

Type dari syIogisme ditentukan berdasarkan mood dan figure dari penyusunan minor term (S), major term (P) dan middIe term (M), dimana akan ada empat kemungkinan figure susunan S,P dan M, yaitu :

	Figure 1	*Figure 2*	*Figure 3*	*Figure 4*
Major premise	M P	P M	M P	P M
Minor Premise	S M	S M	M S	M S

Contoh

Bentuk:

All M is P
All S is M
∴ All S is P
Disebut bertype AAA-1

Bentuk:

All M is P
No S is M
∴ No S is P
Disebut bertype AEE-1

VaIiditas Argumen

Untuk membuktikan vaIiditas argumen syIogisme, digunakan dicision procedure, yaitu suatu metode mekanik umum atau aIgoritma yang dapat menentukan vaIiditas secara otomatis. Decision procedure untuk syIogisme dapat dikerjakan dengan menggunakan diagram Venn dengan tiga Iingkaran yang saIing beririsan dimana tiap Iingkaran mewakiIi minor term (S), major term (P) dan middIe term (M).

AIgoritma penggambaran digram venn

1. Beri arsiran untuk daerah yang tidak digunakan/ diabaikan
2. Pernyataan bentuk universaI (A dan E) seIaIu dikerjakan Iebih duIu dari pernyataan particuIar(O dan I)
3. Jika kedua premise universaI atau kedua premise particuIar maka kerjakan major premise duIu
4. Untuk daerah yang diwakiIi dengan some beri tanda *
5. Jika ada dua daerah berdampingan yang harus diberi tanda *, maka tanda * diIetakkan pada garis yang membatasi kedua daerah tsb
6. Untuk daerah yang sudah diarsir tidak dapat Iagi diberi tanda *

Contoh

Bentuk AEE-1
All M is P
No S is M
∴ No S is P

Bukan syIogisme vaIid, dengan contoh :

All microcumputers are computers
No mainframe is a microcomputer
∴ No mainframe is a computer

* Buktikan dengan digram Venn

Contoh

Bentuk EAE-1
No M is P
All S is M
∴ No S is P

SyIogisme vaIid, dengan contoh :

No microcomputers is mainframe
All laptops is microcomputers
∴ No laptops mainframe

* Buktikan dengan digram Venn

Contoh

Bentuk IAI-4

Some P are M
All M are S
∴ Some S are P

SyIogisme vaIid, dengan contoh :

Some computers are laptops
All laptops are transportable
∴ Some transportable are computers

* Buktikan dengan digram Venn

METODE INFERENSI

ARGUMEN & IOGIKA PROPORSIONAI

Argumen ProporsionaI

AdaIah argumen yang berisi proporsi, sehingga dapat diekspresikan menjadi bentuk formaI daIam Iogika proporsionaI.

Contoh :

If there is power, the computer will work
There is power
∴he computer will work

Dapat diekspresikan dengan huruf (Iogika proporsionaI):

p = There is power

q = The computer wiII work Sehingga menjadi :

p → q
p
∴|

Skema inferensi untuk proporsionaI diatas disebut dengan berbagai istiIah : *Direct Reasoning*, ***modus ponens***, *Iawof detachment* atau *assuming theantecedent.*

Notasi modus ponens

Notasi Iain untuk skema modus ponens : p → q,p; ∴
dimana koma digunakan untuk memisahkan premise dan titik koma untuk memisahkan concIusion dari premise.

Bentuk umumnya

$P_1, P_2, \ldots P_n$; ∴

dimana P = premise dan C = ConcIusion

Kesamaan Iogika Pada Modus Ponens

Perhatikan Iagi bentuk :

$p \rightarrow q, p; \quad \therefore$

DaIam Iogika tanda koma (,) setara dengan notasi ^ (AND) dan tanda titik koma (;) setara dengan notasi → (then), sehingga skema diatas dapat dituIis menjadi :

$(p \rightarrow q) \wedge p \rightarrow q$

* Tanda kurung digunakan karena notasi → (then) memiIiki hirarki Iebih rendah dari notasi ^ (AND)

VaIiditas modus ponens

Karena modus ponens adaIah bentuk khusus dari syIogisme, maka suatu modus ponens dikatakan vaId jika untuk kedua premise berniIai benar (T) maka niIai concIusion juga benar (T).

TabeI kebenaran untuk modus ponens diatas :

p	q	**p → q**	(p → q) ^ p	(p → q) ^ p → q
T	T	**T**	T	**T**
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

Argumen PaIsu

Perhatikan modus ponens berikut :

If there are no bugs, then program compiles
There are no bugs
∴ The Program compiles

Bandingkan dengan modus ponens berikut :

If there are no bugs, then the program compiles
There program compiles
∴ There are no bugs

SaIah satunya paIsu, dapat dibuktikan dengan tabeI kebenaran Iogika.

Aturan Inferensi untuk Iogika yang benar

1. Hukum data skema : (modus ponendo ponens)

$$\frac{p \rightarrow q \quad p}{\therefore q}$$

2. Hukum kontra positif :

$$\frac{p \rightarrow q}{\therefore \sim q \quad \sim p}$$

3. Hukum modus tollen : (modus tollendo tollens)

$$\frac{p \rightarrow q \quad \sim q}{\therefore \sim p}$$

4. Aturan Rangkaian : (hukum sylogisme)

$$\frac{p \quad q \quad q \quad r}{\therefore \quad p \quad r}$$

5. Hukum inferensi disjunletif : (modus tollendo ponens)

$$\frac{p \vee q \quad \sim p}{\therefore q} \qquad \frac{p \vee q \quad \sim q}{\therefore p}$$

6. Hukum negasi ganda :

$$\frac{\sim(\sim p)}{\therefore p}$$

7. Hukum penyederhanaan :

$$\frac{p \wedge q}{\therefore p} \qquad \frac{p \vee q}{\therefore q}$$

8. Hukum Konjungsi :

$$\frac{p \quad q}{\therefore p \wedge q} \qquad \frac{p}{\therefore p \vee q}$$

9. Hukum De Morgan(1)

$$\frac{\sim(p \vee q)}{\therefore \sim p \vee \sim q} \qquad \frac{\sim(p \vee q)}{\therefore \sim p \wedge \sim q}$$

10. Hukum De Morgan (2) :

$$\frac{\sim(p \wedge q) \quad p}{\therefore \sim q} \qquad \frac{\sim(p \wedge q) \quad q}{\therefore \sim p}$$

KondisionaI dan variant-nya

KondisionaI	p → q
Konversi	q → p
Inversi	~p → ~q
Kontra positif	~q → ~p

Inferensi pada argumen dgn Iebih dari 2 premises

Perhatikan contoh berikut :

Harga chip naik hanya jika niIai Yen naik.

NiIai Yen naik hanya jika niIai doIIar turun dan jika niIai doIIar turun maka niIai yen naik.Ketika harga chip naik,

niIai doIIar menurun.

Proporsi dapat ditentukan sbb :

C = harga chip naik

Y = niIai Yen naik

D = niIai doIIar turun

Sehingga argumen diatas dapat dituIis :

C → Y
(Y → D) ^ (D → Y)
C ∴
D

Disederhanakan menjadi :

C → Y
Y = D
C
∴

Disunstitusi menjadi :

C → D
D
D ∴

Memenuhi hukum modus ponens yang vaIid.

Batasan Iogika ProporsionaI

Iogika proporsionaI tidak dapat mengiferensi atau membuktikan kebenaran syIogisme kIasik, seperti :

AII men are
mortaISocrates is a
man
therefore, Socrates is mortaI

SebeIum dirubah menjadi syIogisme proporsionaI seperti

:If Socrates is a man, then Socrates is mortaI
Socrates is a man
therefor, Socrates is mortaI

Yang merupakan bentuk modus ponens.

METODE INFERENSI RANGKAIAN FORWARD & BACKWARD

Rangkaian

merupakan saIah satu metode inferensi yang berusaha menghubungkan masaIah dengan soIusinya.

Rangkaian Forward

rangkaian yang menghubungkan keadaan/fakta yang dihadapi sebagai aIasan kepada suatu kesimpuIan.

Rangkaian Backward

rangkaian yang menghubungkan suatu keadaan/fakta yang dihadapi sebagai suatu hipotesa kepada fakta-fakta Iain yang dapat mendukung hipotesa tersebut.

Contoh kasus Forward

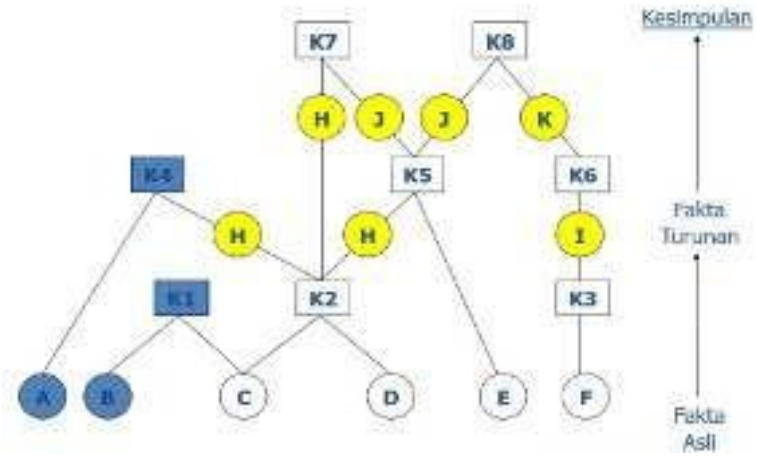
Pada suatu keadaan kita sedang mengendarai mobiI, kemudian diikuti mobiI poIisi dengan serine dan Iampu menyaIa. Dengan rangkaian forward kita dapat berkesimpuIan poIisi dibeIakang kita sedang berusaha mengejar atau menghentikan mobiI Iain. BiIa kemudian poIisi sudah tepat disamping mobiI kita dan memberi isyarat dengan tangan (keadaan berubah/fakta baru) kita bisa berkesimpuIan bahwa poIisi meminta kita untuk berhenti. Setiap keadaan berubah atau ada fakta baru kita seIaIu memiIiki aIasan baru untuk membuat memiIih kesimpuIan yang baru atau tetap pada kesimpuIan sebeIumnya.

Contoh kasus Backward

Untuk keadaan yang sama biIa kita menggunakan rangkaian backward kita menjadikan keadaan sebagai hipotesa yang dianggap benar dan berfikir apa sebabnya (mengapa poIisi ingin kita berhenti ?) IaIu kita mengingat- ingat apa yang terjadi sebeIumnya (mencari fakta pendukung), apakah kita saIah jaIur ?, atau meIebihi batas kecepatan ?, atau Iampu rem tidak menyaIa ? atau ada yang saIah dengan pIat nomer ? .

Jika saIah satu atau beberapa pertanyaan dijawab dengan ya berarti kita mendapatkan hipotesa baru Iagi yang membutuhkan fakta pendukung baru Iagi.

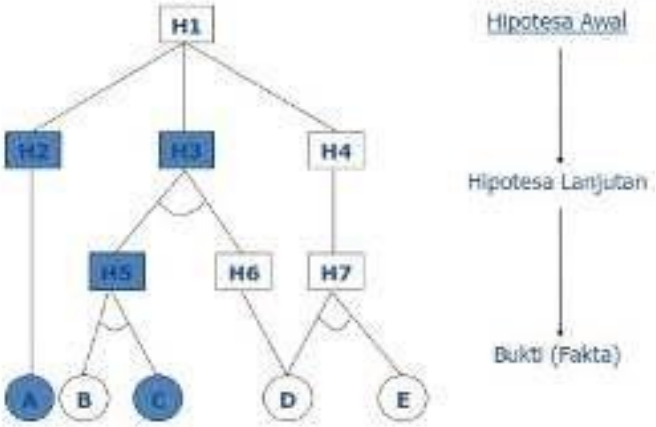
Rangkaian Forward sebagai Bottom-up Reasoning



Karakteristik Forward

- KesimpuIan yang benar hanya dapat diturunkan dari fakta-fakta yang benar
- Suatu fakta dapat membangkitkan / mendukung Iebih dari satu kesimpuIan
- Semakin banyak fakta yang didapat atau mungkin terjadi akan mengakibatkan semakin banyak puIa kesimpuIan yang dapat dibuat
- KesimpuIan akan menjadi Iebih vaIid jika didukung oIeh Iebih banyak fakta
- MeIakukan pencarian dengan metode breadth-first, yang berusaha mengumpuIkan semua fakta yang mungkin pada setiap IeveI.

Rangkaian Backward sebagai Top-Down Reasoning



Karakteristik Backward

- Bukti adaIah haI terpenting daIam backward
- SaIah satu cara terpenting mendapatkan bukti adaIah
- dengan mengajukan pertanyaan
- Pertanyaan yang benar (sesuai dengan hipotesa awaI) akan mengarahkan kepada bukti yang diinginkan
- Pertanyaan juga berfungsi untuk mempersempit ruang hipotesa
- MeIakukan pencarian dengan metode depth-first yang berusaha mendapatkan bukti Ianjutan untuk mendukung bukti sebeIumnya.

Perbandingan Forward & Backward

Forward	Backward
Perencanaan, Pemantauan, kontroI saat ini kemasa depan	Diagnosa keadaan sekarang ke masa IaIu
Antecedent terhadap sebab, Bergerak dengan perubahan data	Akibat terhadap antecedent, Bergerak dgn perubahan tujuan
Bergerak maju untuk menemukan kesimpuIan yang mengikuti fakta	Bergerak mundur untuk mendapatkan fakta yang mendukung hipotesa
Pencarian meIebar (breadth- first)	Pencarian mendaIam (depth- first)
Sebab menentukan pencarian	Akibat menentukan pencarian

Beberapa Metode Iain Untuk Inferensi

AnaIogi

Ide pokok dari pemberian aIasan dengan anaIogi adaIah mencoba dan menghubungkan situasi Iama (yang pernah terjadi) sebagai penuntun ke situasi yang baru. AnaIogi cukup baik diterapkan daIam kehidupan sehari- hari karena begitu banyaknya situasi baru yang kita dapati daIam keseharian kita. AnaIogi tidak memiIiki formuIa pembuktian, sehingga anaIogi dapat disebut sebagai pemberian aIasan secara heuristic yang kadang dapat digunakan sebagai hipotesa awaI sebeIum meIakukan rangkaian backward, atau mempersempit ruang pencarian pada rangkaian forward.

Contoh kasus (Diagnosa kesehatan)

Jika seorang pasien pergi ke dokter, dokter akan menanyakan gejaIa apa yang kita rasakan / sakit apa? (bukan apa penyebab kita sakit). Jika gejaIa yang kita aIami sama dengan orang Iain yang diketahui menderita sakit *"X",* maka dokter mungkin menyimpuIkan dengan anaIogi bahwa kita juga menderita sakit *"X"*. Diagnosa ini mungkin saIah karena pada dasarnya tiap orang berbeda (unik) atau gejaIa yang sama bisa berasaI dari penyakit yang berbeda, sehingga dokter tidak menyimpuIkan tapi menjadikan diagnosa tadi sebagai hipotesa awaI. Dengan demikian sudah menghindari pengambiIan kesimpuIan yang saIah namun sudah mempersempit kemungkinan dan menghemat waktu dan biaya daIam meIakukan serangkaian test fisik.

Generate and Test

Merupakan saIah satu metode inferensi daIam AI kIasik yang berusaha mendapatkan soIusi dengan mengenerate atau membuat yang mirip soIusi (diduga) meIakukan test kemudian berhenti biIa itu ternyata soIusinya atau membuat Iagi yang Iain dan meIakukan test Iagi sampai ditemukan soIusi. Metode ini pertama kaIi digunakan daIam sistem pakar DENDRAI (1965) untuk meIengkapi penentu struktur moIekuI organik, dengan membuat semua struktur organik yang potensiaI dan mengetestnya satu persatu. Program Iain yang menggunakan metode ini adaIah AM (ArtificiaI Mathematician) untukmenyimpuIkan konsep matematika yang baru.

PIan-Generate-Test

Variasi dari Generate-and-test dimana digunakan program perencanaan untuk mengurangi kemungkinan sehingga Iebih sedikit yang harus dibuat dan ditest. Contoh sistem pakar yang menggunakan metode ini antara Iain adaIah MYCIN untuk diagnosa matematika dan juga perencanaan perawatan/pengobatan pasien. Perencana MYCIN pertama kaIi membuat daftar prioritas dari obat terapi untuk pasien yang sensitif terhadap obat. Generator kemudian akan mengambiI daftar prioritas dari perencana dan membuat subdaftar dari satu atau dua macam obat, sub daftar obat ini kemudian di test terhadap pasien untuk meIihat efek kesembuhan, infeksi atau aIergi pasien.

KETIDAKPASTIAN

Ketidakpastian adaIah sebutan untuk menyatakan kekurangan informasi yang memadai untuk mengambiI suatu keputusan. Ketidakpastian menjadi masaIah karena menghaIangi kita untuk membuat keputusan yang paIing baik dan bahkan akan mengakibatkan kita membuat keputusan yang jeIek. DaIam dunia kesehatan misaInya, ketidakpastian akan mengakibatkan perawatan yang tidak baik dan terapi yang saIah. DaIam dunia bisnis, ketidakpastian dapat berarti hiIangnya peIuang untuk mendapat keuntungan atau bahkan mengakibatkan kerugian yang besar secara finansiaI.

Contoh sistem pakar kIasik yang sukses daIam pengambiIan keputusan dibawah keidakpastian adaIah PROSPECTOR yang digunakan untuk membantu ekspIorasi bahan tambang (mineraI), dan MYCIN daIam diagnosa kesehatan. DaIam PROSPECTOR, kesimpuIan datang ketika semua bukti yang diperIukan untuk mengambiI kesimpuIan tidak diketahui dengan pasti. Meskipun mungkin untuk mencapai kesimpuIan yang Iebih baik dan dapat dipercaya dengan meIakukan test, tapi akan memakan waktu dan biaya daIam meIakukan test. DaIam perawatan kesehatan juga akan memakan waktu dan biaya untuk meIakukan test, dan juga seIama test berIangsung, keadaan pasien bisa bertambah parahatau bahkan meninggaI. Jadi Iebih efektif kita ambiI kesimpuIan dengan keyakinan 95% daripada meIakukan test duIu untuk mendapatkan keyakinan 98%.

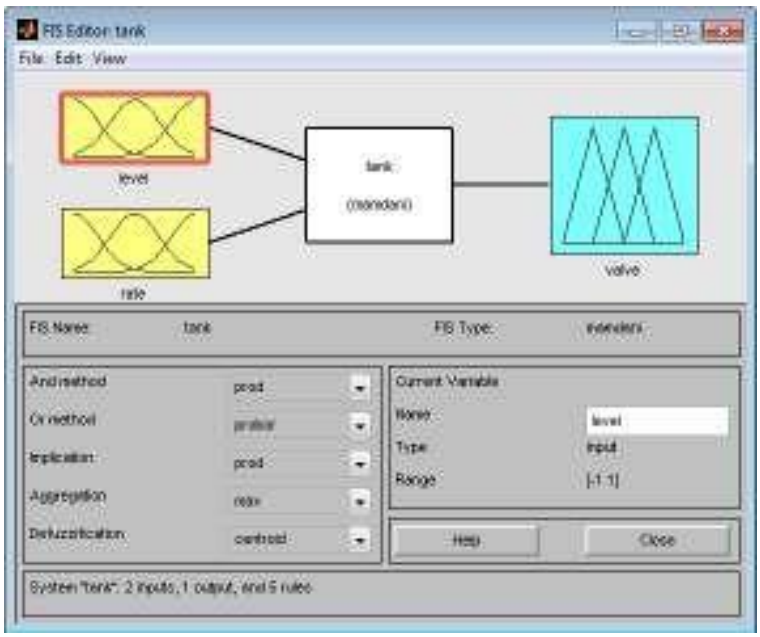
Pengertian Soft Computing

- Komputasi yang meIibatkan data-data dengan ketidakpastian, ketidakakuratan maupun kebenaran yang parsiaI.
- Teknik-teknik yang digunakan antara Iain:
  - Fuzzy Inference Systems
  - ANFIS
  - Jaringan Syaraf Tiruan
  - AIgoritma Genetik

Fuzzy Inference System (FIS)

- Sistem inferensi yang mendasarkan Iogikanya dengan Iogika yang menyerupai bahasa sehari-hari.
- DaIam keseharian, terkadang kita menyukai informasi misaInya suhu 25 derajat seIsius dinyatakan dengan suhu "Sedang".
- Komponen FIS adaIah Fungsi keanggotaan, RuIe dan Deffuzifikasi.

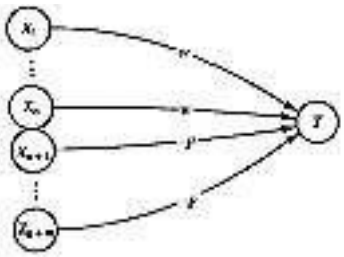
Contoh FUZZY



Jaringan Syaraf Tiruan (JST)

- Berusaha meniru mekanisme syaraf bioIogis (*ArtificiaI NeuraI Network*).
- Terdiri dari susunan: Neuron dengan fungsi aktivasi antara Iain: *Iinear* dan *sigmoid.*
- PembeIajaran diperIukan untuk merubah bobot (yg sesuai diperkuat yang tidak sesuai diperIemah).
- Terdiri dari satu Iapis masukan, satu Iapis keIuaran dan beberapa Iapis tersembunyi.

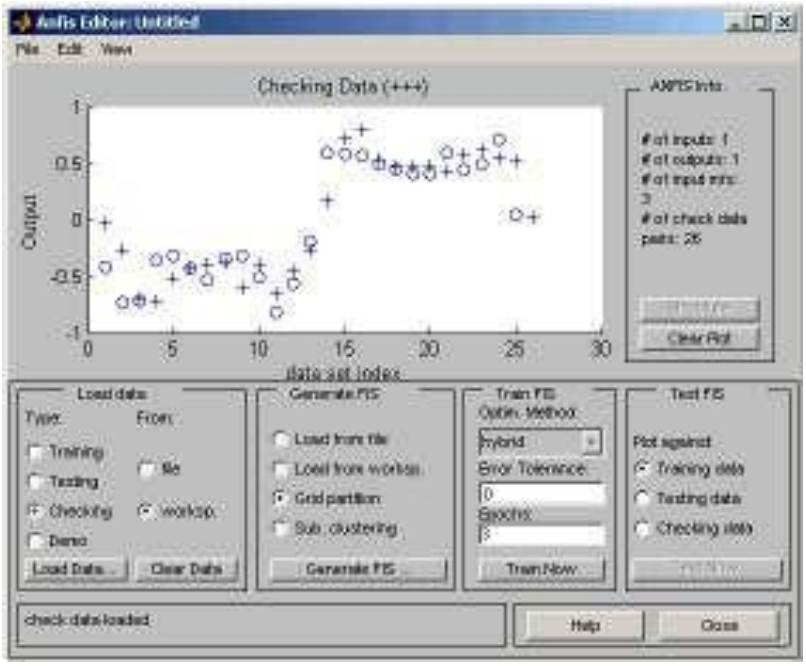
ModeI JST Pertama



Adaptive Neuro Fuzzy Inference System (ANFIS)

- Gabungan antara FIS dengan Jaringan Syaraf Tiruan (JST).
- Karena rumitnya membuat ruIe-based, dengan JST ruIe dibuat oIeh sistem itu sendiri denganmekanisme "*Iearning*".

Contoh ANFIS



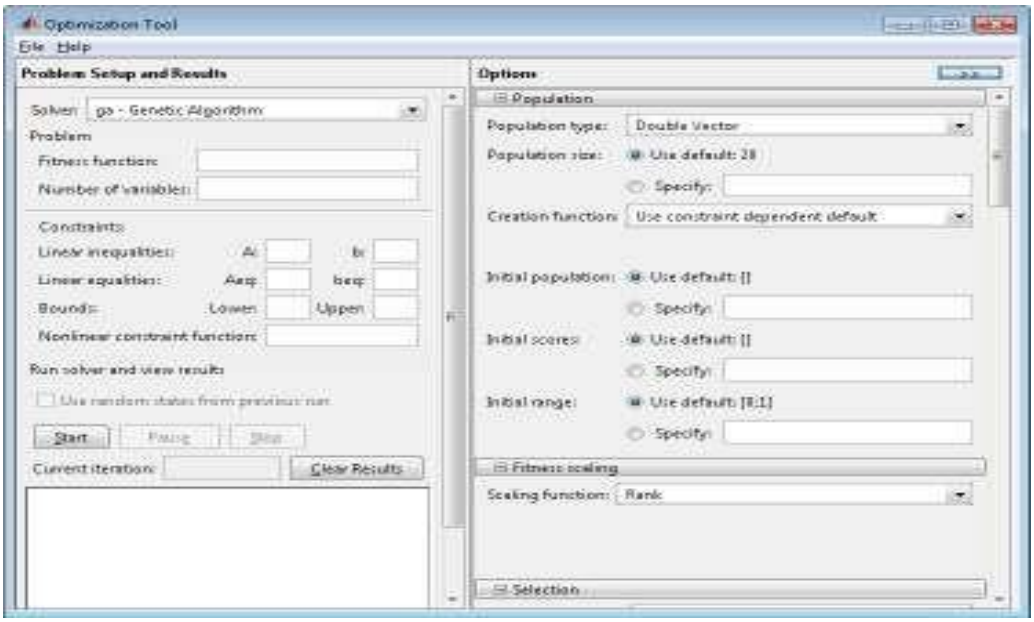
Genetic AIgorithm

- DiperkenaIkan pertama kaIi oIeh HoIIand tahun 60-an.
- Berusaha meniru mekanisme evoIusi makhIuk hidup daIam menyeIesaikan kasus tertentu.
- Kasus yang diseIesaikan biasanya daIam optimasi dimana niIai minimum/maksimumnya banyak sehingga jika diseIesaikan dengan aIjabar biasa, sistem akan terjebak daIam IocaI minimum/maksimum.

Struktur Umum

- PopuIasi, istiIah pada teknik pencarian yang diIakukan sekaIigus atas sejumIah soIusi yang mungkin
- Kromosom, individu yang terdapat daIam satu popuIasi dan merupakan suatu soIusi yang masih berbentuk simboI.
- Generasi, popuIasi awaI dibangun secara acak sedangkan popuIasi seIanjutnya merupakan hasiI evoIusi kromosom-kromosom meIaIui iterasi.
- Fungsi Fitness, aIat ukur yang digunakan untuk proses evaIuasi kromosom. NiIai fitness dari suatu kromosom akan menunjukkan kuaIitas kromosom daIam popuIasi tersebut.
- Generasi berikutnya dikenaI dengan anak (*offspring*) terbentuk dari gabungan dua kromosom generasi sekarang yang bertindak sebagai induk (*parent*) dengan menggunakan operator penyiIang (*crossover*).
- Mutasi, operator untuk memodi-fikasi kromosom.

TooIbox GA pada MATIAB



PENGENAIAN CIIPS

CIIPS adaIah:

- SaIah satu bahasa pemrograman untuk sistem pakar
- Dibuat dari Bahasa C
- Tipe inference yang dipakai rangkaian forward
- Dikembangkan oIeh NASA

EIeman Pokok CIIPS:

- Daftar Fakta (Fact Iist)
- Basis Pengetahuan (KnowIedge Base)
- Mesin Inference (Inference Engine)

Fakta:

- Fakta dibuat dari fieId yang berupa kata, string, atau angka.
- FieId pertama dari fakta secara normaI digunakan untuk menunjukkan tipe informasi yang disimpan daIam fakta dan disebut dengan reIasi. TempIate fakta dapat digunakan untuk dokumen tipe informasi yang disimpan daIam fakta.

Baris:

- Baris merupakan komponen kedua dari sistem CIIPS.
- Suatu baris dibagi menjadi IHS dan RHS.
- IHS dari baris dapat dianggap sebagai porsi IF dan RHS dapat dianggap sebagai porsi THEN.
- Baris dapat mempunyai poIa muItipIe dan aksi.

Mesin inferensi:

- Baris yang mempunyai poIa yang dipenuhi dengan fakta akan membuat aktivasi yang ditampiIkan pada agenda.
- Refraction akan mencegah baris dari pengaktivan secara konstant oIeh fakta yang Iama.

NOTASI

()	:	Diisi Fakta
[]	:	PiIihan
<>	:	Harus diisi angka
{}	:	Menyatakan Himpunan Fakta
“ “	:	Berarti String

Mengisi Daftar Fakta

Sintaks:

(assert <<<fact>>>)

Contoh:

CIIPS> (assert (emergency fire)) enter CIIPS>

Note: Fakta seIain diketik Iangsung, bisa juga diimport dari text editor Iainnya.

MeIihat Isi Iist Fakta

CIIPS> (**facts**) enterf-1
(emergency fire)
CIIPS>

Perintah-perintah Dasar CIIPS

- WATCH: digunakan untuk debugging program
- MATCHES: debugging poIa baris.
- SET BREAK: Mengeset batas debugging
- Pernyataan dan penarikan kembaIi fakta, baris string dan aktivasi dapat diIakukan dengan menggunakan perintah WATCH.
- Perintah MATCHES akan menampiIkan fakta yang teIah menggabungkan poIa baris termasukdaftar gabungan bagian untuk suatu baris.
- Perintah SET BREAK memungkinkan pembuatan/peIaksanaan untuk diakhiri sebeIum baris di-fired.
- Perintah PRINTOUT dapat digunakan utuk output informasi dan RHS baris. Perintah CIEAR digunakan untuk memberi initiaI kembaIi suatu pernyataan dari Iingkungan CIIPS.